



ADMINISTRATOR GUIDE

6.3.1 | February 2019 | 3725-85477-002A

Polycom® VVX® Business IP Phones, OBi Edition

VVX 150/VVX 250/VVX 350/VVX 450



Copyright© 2019, Polycom, Inc. All rights reserved. No part of this document may be reproduced, translated into another language or format, or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Polycom, Inc.

6001 America Center Drive
San Jose, CA 95002
USA

Trademarks Polycom®, the Polycom logo and the names and marks associated with Polycom products are trademarks and/or service marks of Polycom, Inc., and are registered and/or common law marks in the United States and various other countries.



All other trademarks are property of their respective owners. No portion hereof may be reproduced or transmitted in any form or by any means, for any purpose other than the recipient's personal use, without the express written permission of Polycom.

Disclaimer While Polycom uses reasonable efforts to include accurate and up-to-date information in this document, Polycom makes no warranties or representations as to its accuracy. Polycom assumes no liability or responsibility for any typographical or other errors or omissions in the content of this document.

Limitation of Liability Polycom and/or its respective suppliers make no representations about the suitability of the information contained in this document for any purpose. Information is provided "as is" without warranty of any kind and is subject to change without notice. The entire risk arising out of its use remains with the recipient. In no event shall Polycom and/or its respective suppliers be liable for any direct, consequential, incidental, special, punitive or other damages whatsoever (including without limitation, damages for loss of business profits, business interruption, or loss of business information), even if Polycom has been advised of the possibility of such damages.

End User License Agreement By installing, copying, or otherwise using this product, you acknowledge that you have read, understand and agree to be bound by the terms and conditions of the End User License Agreement for this product. The EULA for this product is available on the Polycom Support page for the product.

Patent Information The accompanying product may be protected by one or more U.S. and foreign patents and/or pending patent applications held by Polycom, Inc.

Open Source Software Used in this Product This product may contain open source software. You may receive the open source software from Polycom up to three (3) years after the distribution date of the applicable product or software at a charge not greater than the cost to Polycom of shipping or distributing the software to you. To receive software information, as well as the open source software code used in this product, contact Polycom by email at OpenSourceVideo@polycom.com.

Customer Feedback We are striving to improve our documentation quality and we appreciate your feedback. Email your opinions and comments to DocumentationFeedback@polycom.com.

Polycom Support Visit the [Polycom Support Center](#) for End User License Agreements, software downloads, product documents, product licenses, troubleshooting tips, service requests, and more.

Contents

- Before You Begin 11**
 - Audience, Purpose, and Required Skills 11
 - Get Help 11
 - Polycom and Partner Resources 11
 - The Polycom Community 11
 - Documentation Feedback 12
 - Notational Conventions 12
 - Canonical Fashion 12
 - Literal Fashion 12
 - Boolean Values 13
 - Multiple Choice Values 13
 - Parameter Values 13

- Getting Started 14**
 - Product Overview 14
 - Supported Phones and Accessories 14
 - Working with OBi Edition VVX Phones 15
 - Input Methods 15
 - Phone Keypad 15
 - Logging In to the Phone 15
 - Phone Native Web Portal 16
 - Find the Phone IP Address 16
 - Local Device Configuration 16
 - Access the Device Management Web Portal 16
 - Navigating the Device Management Web Portal 16
 - OBiTALK Portals 18
 - User Portal 18
 - Device Management Platform Portal – for ITSPs 19
 - Record Version Information 19

- Supported Network Configurations 20**
 - Configuring the External Ethernet Ports 20
 - Configuring the WAN Interface 20

Configuring the Wi-Fi Interface	21
DHCP Options	22
DNS Lookup	22
Lookup Order	22
Locally Configured DNS Lookup Table	22
NTP Servers and Local Time	23
Manually Configuring Phones	24
Telephone-IVR-Based Local Configuration	24
Main Menu	24
Additional Integrated Voice Response Operations Options	26
Phone UI	32
Settings	32
Preferences	33
Provisioning Phones	34
Remote Provisioning	34
3CX Plug-and-Play Provisioning	36
Configuring Security Options	37
802.1X Authentication	37
Upgrading the Software	38
Updating the Firmware	38
Update Firmware with IVR	38
Update Firmware from Phone User Interface	38
Update Firmware from the Native Web Portal	38
Automated Firmware Update	38
Background Firmware Update	39
Customization Data Auto Update	40
Customization Data Package	40
Auto Update Operation	41
Auto Update Configuration Parameters	41
Diagnostics and Status	43
Factory Reset	43
Phone Display and Appearances	44
LED Pattern Customization	44
LED Settings Parameters	44
UI Menu Customization	51

Main Menu Item IDs	52
Net Services Menu Item IDs	53
Preferences Menu Item IDs	54
Settings Menu Item IDs	56
Product Info Menu Item IDs	56
Cache Control of Downloaded (Temporary) Data	57
Phone Customization Data	57
Background Pictures	58
Text Fonts Customization	58
Language Customization with Dictionary Files	62
Phone Book Pictures	72
Ring Tones	72
Phone Customization Data Package	72
Uploading Customization Data Package to the Phone	73
Internal Data Storage Paths for User Preferences Settings	74
Capture Your Device's Current Screen	76
System Logs	77
Syslog	77
Hardware and Accessories	78
Powering OBi Edition VVX Phones with an Ethernet Switch Connection	78
Available Accessories	78
Audio Features	79
Tone Patterns	79
Tone Profile Features of Your Phone	79
Field-1 Composition	79
Field-2 Composition	79
Field-3 to Field-6 Composition	80
Tone Examples	81
Ringtones and Ring Patterns	82
Field-1 Composition	82
Field-2 to Field-5 Composition	83
Voice Services	84
ITSP Profiles	84
Audio Codec Profiles	85
Specification of Target Phone Numbers	85
Overview of Common Trunk Configuration	85
Trunk Capacity	85

Service Enable	85
Service Account Credentials	85
Servers	86
SIP/SP Service	87
OBiTALK Service	87
Directories and Contacts	89
LDAP Service Setup	89
Client Authentication	93
LDAP Directory Search Application	93
Invoke LDAP by Main Menu — Directories Option	93
Invoke LDAP by Main Menu – Network Directory Option	93
Invoke LDP by Softkey — LDAP	94
Search Fields	94
Result Fields	94
Sorting of Results	95
Replace the Built-In Phone Book with LDAP	95
Call Controls	96
Phone Level and Line Level Feature	96
Call States	96
Core Call Features	98
Line Capacity	98
Complex Operations Between Multiple, Diverse Voice Services	98
Making Outgoing Calls	98
Digit Map	98
Audio Path and On-Hook or Off-Hook States	98
Off-Hook Dialing	99
On-Hook Dialing	99
Outbound Call Routes	99
Primary Line	99
Explicitly Selecting a Line to Make a Call	100
Dialing “Speed Dials 99” Numbers	100
Called Party Caller ID Display	100
Handling Incoming Calls	100
Inbound Call Routes	100
Rejecting Incoming Calls	101
Caller ID Display	101
Ending Calls	101
Holding Calls	101

Resuming Calls	101
“Foregrounding” a Call	102
Call Waiting	102
Call Transfer	102
Transfer Signaling	104
Limitations of Transfer by Internal Bridging	104
Conference Calls	104
Local Mixing/Bridging	104
External Conference Bridge	105
Auto Attendant	105
AA Callback Service	106
Expanded Call Features	107
Auto Answer and Intercom	107
Auto-Answer Incoming Call Based on Inbound Call Routing Rules	108
Auto-Answer when Caller Requests to Barge-In (with OBiTALK Service)	108
Barge-In as a Coach	108
Push To Talk	109
Speed Dial	109
Speed Dial Feature Key	109
Speed Dial Numbers	110
Block Caller ID	110
Block Anonymous Call	111
Calling Line ID Display	111
Call Forwarding	112
Call Forward Numbers	112
Call Forward ALL	112
Call Forward on Busy	112
Call Forward on No Answer	112
Call Forward Signaling	113
Limitations of Call Forward by Internal Bridging	113
Do Not Disturb	114
Do Not Ring	114
Message Waiting Indication – Visual and Tone Based	114
Multicast Page Groups	115
Music On Hold (MOH)	117
Premium Call Features	117
Busy Lamp Field (BLF)	117
Single Versus Multiple BLF Event Notification	118
BLF with Call Park Status	119
What Happens When BLF Key is Pressed	119

BLF Operation: Speed Dial	119
BLF Operation: Directed Call Pickup	119
BLF Operation: Barge In	119
BLF Operation: Call Pickup	120
BLF Operation: Resume	120
BLF Configuration	120
Floating BLF Key Assignment	122
SIP for BLF	123
BLF With OBiTALK Service	124
Call Park and Call Pickup	125
Call Park Methods	125
Call Park Monitor and Call Pickup Methods	126
Call Park Ring	127
Shared Line and Shared Call Appearances (SCA)	128
Line Seize	130
What Happens When a Call Appearance Key is Pressed	130
Buddy List	130
Expanded Buddy List and Groups	133
Buddy List Management	133
Presence Monitor	134
Call Recording Controls	134
Hold and Talk Event Package	135
Advice of Charges (AOC)	135
Emergency Calls	135
Call Diversion History	135
Digit Map Configuration	136
Digit Map Elements	136
Digit Map Rule Examples	141
Matching Against Multiple Rules in Digit Map	142
Forcing Interdigit Timeout With The Hash/Pound (#) Key	143
Invoke Second Dial Tone in Digit Map	143
Change Interdigit Long Timer Dynamically After Partial Match	144
User-defined Digit Maps	144
A User-defined Digit Map For IPv4 Dialing	144
Controlling Calls Using Star Codes	145
Star Code Script Variables (VAR)	145
Star Code Script Actions (ACT)	146
Star Code Script Format	147
Star Code Script Examples	147

Customizing Devices	148
Applications	148
Built-In Phone Applications	148
OBiPhoneXML Applications	149
Softkey Set Customization	149
Softkey Set Parameter Syntax	149
Softkey Specification	150
Assignable Softkeys	150
Softkey Set Parameters	161
Feature Keys	163
Preassigned Feature Keys (PFKs)	172
Feature Key Configuration Parameters	173
Highlights of Feature Key Functions	174
Call Keys	174
Line Monitor Keys	174
Speed Dial Keys	175
Busy Lamp Field Keys	175
Call Park Monitor Keys	175
Presence Monitor	175
Group Page Keys	175
Line Keys and Virtual Line Keys	176
Line Key Window Customization	176
Example	180
Line Key Tabs	181
Calls App Behavior	181
IP Phone Settings	181
Phone Settings	181
DigitMap and OutboundCallRoute	181
Primary Line	182
Network Directory	182
Buddy List	183
User Preferences Settings	183
Page Groups 1 and 2	183
Line Keys	183
Network	185
SIP Registration	185
Third-Party Registration	186
Registration Period	186
REGISTER Final Non-2xx Response Handling	187

SIP Outbound Proxy Server	188
DNS Lookup of SIP Servers	188
NAT Traversal Considerations	189
Keep Alive Messages	189
SIP Proxy Server Redundancy and Dual REGISTRATION	191
DNS NAPTR Support	192
SIP Privacy	192
STUN and ICE	193
ITSP Driven Distinctive Ringing	194
RTP Statistics – the X-RTP-Stat Header	194
RTCP	195
Media Loopback Service	195
A SIP/SP Configuration Example	196
Call Routing	197
Basic Incoming Call Handling	197
Inbound Call Route Configuration	198
Basic Outgoing Call Handling	200
Outbound Call Route Configuration	200
Voice Gateways and Trunk Groups	202
Voice Gateways	202
Trunk Groups	203
Trunks, Endpoints, and Terminals	203
Third-Party Servers	205
BroadSoft	205
Built-In BroadSoft Phone Applications	205
Net Services App	205
Net Dir App	206
Call Recording Controls	206
BroadSoft Call Center Features	206
Disposition Code	207
Customer Originated Call Trace	207
Escalation	207
Call Center Information	207
BroadSoft Guest Login/Logout (Hoteling)	208
BroadSoft AS-Feature-Event Features	208
Call Forward All	209
Call Forward Busy	209
Call Forward No Answer	210

Do Not Disturb	210
ACD Agent State	210
Security Classification	211
Executive Call Filter	211
Executive Assistant	211
Call Recording Settings	212
BroadSoft XSI Features	212
Network Directories	213
BroadSoft Hosted PBX Platform	213
Network Directory Softkey Options	213
Other PBX Platforms	213
Replace the Built-In Phone Book with a Network Directory	215
Network Call Logs	215
BroadWorks Anywhere	215
Remote Office	215
Simultaneous Ring	216
Call Forward Always	216
Call Forward Busy	216
Call Forward No Answer	217
Anonymous Call	217
Do Not Disturb	217
Device Web Page and Configuration Parameter Reference	218
Status Parameters	218
System Status Web Page	218
Reboot Reason Codes	220
Call Status Web Page	221
Call History Web Page	222
SP Services Stats Web Page	223
OBiWiFi Configuration Parameters	223
WiFi Settings Web Page	224
WiFi Scan Web Page	225
System Management Parameters	225
WAN Settings Web Page	226
Auto Provisioning Web Page	231
\$MACRO Expansion Supported by the Phone	237
Device Admin Web Page	238
Device Update Web Page	242
Firmware Update	242
Screenshot	243

Extracting PCAP Capture Result	243
Backup Configuration	243
Restore Configuration	244
Reset Configuration	244
Service Providers Parameters	245
ITSP Profile X (X = A, B, C, D, E, F)	245
General Web Page	245
SIP Web Page	246
RTP Web Page	254
Voice Services	255
SPn Service Web Page (n = 1, 2, 3, 4, 5, 6)	256
OBiTALK Service Web Page	269
Auto Attendant Web Page	273
Gateways and Trunk Groups Web Page	275
IP Phone Settings	276
Phone Settings Web Page	276
Left Line Keys Web Page	289
Right Line Keys Web Page (VVX 450 only)	292
Programmable Keys Web Page	293
LED Settings Web Page	295
Soft Keys Web Page	299
LDAP Search Web Page	300
Line Key Customization Web Page	302
Screen Item Customization Web Page	303
Codec Profiles	305
Codec Profile X Web Page (X = A, B)	305
Tone Settings	308
Tone Profile X Web Page (X = A, B)	308
Ring Settings	311
Ring Profile X Web Page (X = A, B)	311
Star Code Profiles	313
Star Code Profile X Web Page (X = A, B)	313
User Settings	315
User Preferences Web Page	315
Speed Dials Web Page	321
User Defined Digit Maps Web Page	321

Before You Begin

This guide describes how to administer, configure, and provision Polycom VVX Business IP phones with OBi Edition software.

Audience, Purpose, and Required Skills

This guide is for a technical audience. You must be familiar with the following concepts before beginning:

- Current telecommunications practices, protocols, and principles
- Telecommunication basics, video teleconferencing, and voice or data equipment
- OpenSIP networks and VoIP endpoint environments

Get Help

For more information about installing, configuring, and administering Polycom products, see **Documents & Software** at [Polycom Support](#).

Polycom and Partner Resources

In addition to this guide, the following documents and other resources provide details about Polycom UC Software:

- For Polycom UC Software releases and documentation, see [Polycom Voice Support](#).
- For user guides for Polycom voice products, refer to the product support page for your phone at [Polycom Voice Support](#).
- For help or technical support for your phones, you can search for Polycom documentation at the [Polycom Unified Communications \(UC\) Software Resource Center](#).
- You can find Request for Comments (RFC) documents by entering the RFC number at <http://www.ietf.org/rfc.html>.
- For information on IP PBX and softswitch vendors, see Polycom [Desktop Phone Compatibility](#). If you're using the Polycom Trio solution, see [Polycom Trio and SoundStation IP Platform Compatibility](#).

To find all Polycom partner solutions, see [Strategic Global Partner Solutions](#).

The Polycom Community

The [Polycom Community](#) gives you access to the latest developer and support information. Participate in discussion forums to share ideas and solve problems with your colleagues. To register with the Polycom Community, simply create a Polycom Online account. When logged in, you can access Polycom support

personnel and participate in developer and support forums to find the latest information on hardware, software, and partner solutions topics.

Documentation Feedback

We welcome your feedback to improve the quality of Polycom documentation.

You can email [Documentation Feedback](#) for any important queries or suggestions related to this documentation.

Notational Conventions

This guide provides device configuration parameters and their values in the following formats:

- Canonical fashion
- Literal fashion

Both notational conventions point to the same parameters, but their appearances are different.

The canonical fashion simplifies locating parameters on the phone's native web portal or on OBITALK.com.

The literal fashion is required when provisioning or writing OBIPhoneXML apps.

Canonical Fashion

This example shows the format of the canonical fashion.

- **Parameter Group Name::ParameterName** = Parameter Value {replace-with-actual-value}

The **Parameter Group Name** is the heading of the parameter group on the left side panel of the device local configuration or OBITALK Configuration web page. This string may contain spaces. When a group heading has more than one level, each level is separated with a –, such as:

- **Services Providers - ITSP Profile A – SIP:**

The **ParameterName** is the name of the parameter as shown on the web page and **MUST NOT CONTAIN ANY SPACES**. **Parameter Group Name** and **ParameterName** are separated by two colons (::), as shown in the first example above.

The `Parameter Value` is the literal value to assign to the named parameter and may contain spaces. You can omit **Parameter Group Name** or its top-level headings when the context is clear. For example:

- **SP1 Service::AuthUserName** = 4082224312
- **ITSP Profile A - SIP::ProxyServer** = sip.myprovider.com
- **ProxyServerPort** = 5082

Literal Fashion

These examples show the format of the literal fashion. The literal fashion is used when provisioning or writing OBIPhoneXML apps.

- **ParameterGroupName.ParameterName**.Parameter Value {replace-with-actual-value}
- **Parameter.Group.Name.ParameterGroupName.ParameterName**.Parameter Value

The ***ParameterGroupName*** is the name of the first parameter group in literal fashion. This string MUST NOT CONTAIN ANY SPACES, and always is terminated with a period, as shown. More than one ***ParameterGroupName*** may be used. The ***ParameterGroupName*** is case-sensitive.

The ***ParameterName*** is the name of the parameter, and always is terminated with a period, as shown. This string MUST NOT CONTAIN ANY SPACES. The ***ParameterName*** is case-sensitive.

The ***Parameter Value*** is the literal value to assign to the named parameter and may contain spaces. The ***Parameter Value*** is not case-sensitive, but it MUST EXACTLY MATCH the value when one or more choices are available.

When using the literal fashion in your XML, you need to exactly match the text string for ***ParameterGroupName.ParameterName.Parameter Value***, but text formatting such as bold face is not required and is removed when your script or app is processed.

Boolean Values

You can identify parameters that take a Boolean value on your phone's configuration web pages by a check box next to the parameter name. Throughout the document, we refer to a Boolean value as "enable or disable" or "yes or no", but the only valid Boolean parameter values to use in a phone configuration file is either `true/false` or `True/False` (case-sensitive). This is equivalent to selecting or clearing the check box on the configuration web pages.

Multiple Choice Values

You must provision parameters that take one of several valid options from a drop-down list on the device message with string values that match exactly one of those choices. Otherwise, the device uses the default choice. Matching the provisioned value against valid strings is case-sensitive and doesn't allow extra spaces.

Parameter Values

When entering a parameter value from the web page or via provisioning, avoid adding extra white spaces before or after the parameter value. If the value is a comma-separated list of strings or contains attributes after a comma or semicolon, avoid adding extra white space before and after the delimiter.

For example: ***CertainParameter*** = 1,2,3,4;a;b;c

If a parameter value can include white spaces, such as ***X_DisplayLabel***, use just a single space and no extra space before and after the value.

For example: ***X_DisplayLabel*** = My New Service

Getting Started

The Polycom VVX OBi Edition business IP phones support Polycom HD Voice technology. You can manage the phone user interface (UI) and the network interaction on OBi Edition VVX phones directly from your phone or through an associated web portal.

Product Overview

VVX OBi Edition business IP phones implement the following features and functionalities:

- Six SIP accounts with universal inter- and intra-service two-way call bridging among the six accounts and the OBiTALK service.
- High-quality voice encoding using G.711, G.729, G.722, iLBC, and OPUS algorithms.
Note: The VVX 150 doesn't support the OPUS codec
- Recursive digit maps and associated call routing (outbound and inbound).
- Fully programmable line keys and softkeys.
- Programmable feature keys with preassigned functions and labels.
- Support for all standard SIP-based IP PBX and ITSPs/VSPs.
- Support for 3CX PBX and uaCSTA interoperability.
- Suited for all service provider and enterprise deployment environments, regardless of size.
- Ideal for self-service installations—home users, small business owners, or corporate IT departments
- Integration with softswitch architectures.
- Cloud management enabled via OBiTALK.com with both a user portal and an ITSP partner portal with an optional REST API.

Supported Phones and Accessories

The following table lists the product names and features for Polycom VVX OBi Edition business IP phones.

Polycom VVX OBi Edition Products and Accessories

Product Name	Features
VVX 150 business IP phone	Compact monochrome display
VVX 250 business IP phone VVX 350 business IP phone VVX 450 business IP phone	High-resolution color active-matrix TFT LCD display with a customizable UI
5 VDC power adapter	Powers your phone when Power over Ethernet (PoE) isn't available

Polycom VVX OBi Edition Products and Accessories

Product Name	Features
Polycom® WiFi wireless network adapter	Connects your phone to a Wi-Fi network (VVX 250, VVX 350, and VVX 450 only)
USB headset	(VVX 250, VVX 350, and VVX 450 only)

Working with OBi Edition VVX Phones

You have several ways to configure and manage OBi Edition VVX business IP phones. Use the methods that best suit your deployment scenario.

Input Methods

You can input information directly into your phone using different options in your phone's keypad.

Phone Keypad

You can input digits or text directly from your phone's keypad. There are three input modes:

- **Numeric (123)**: Each key press inputs the corresponding digit 0–9, *, #
- **Alphanumeric (abc)**: Each key press invokes a choice of ASCII characters where you can select the character to input
- **Capitalized Alphanumeric (Abc)**: Same as alphanumeric mode except the ASCII character selection has default set to a capital letter
- **IPv4 Address (IPv4)**: Same as numeric, but * maps to a dot (.) and # maps to a colon (:) for entering an IPv4 address

You can change between input modes by pressing the `mode` softkey (by default, label is **Switch Mode**).

Logging In to the Phone

You can access the OBi Edition VVX business IP phones and web portals using default account credentials:

- For user access, the default user name and password are `user` and `user`.
- For admin access, the default user name and password are `admin` and `admin`.

If you don't have access to the phone or web portals using the default credentials, contact your service provider for the user name and password. Make sure that you have access to the correct admin or user password before you attempt to log in to the OBi Device Management web page.



The **Voice Services** and **Device Administration** options in the UI are protected by the same admin password used for accessing the phone's local configuration web pages.

Phone Native Web Portal

Each phone has its own built-in portal site at `http://{ip-address}` where `{ip-address}` is the IP address of the phone. The native web portal offers configurable options and status information organized into a number of web pages. The default page is the **System Status** page.

Default user name and passwords are `user` and `user` for the user level, and `admin` and `admin` for the admin level.

Admin level has full access to all configuration parameters. The administrator can decide which parameters are hidden, read-only, or read-writable at the user level.

Find the Phone IP Address

Each phone must have a valid IPv4 address to connect to the network and communicate with other devices or cloud-based services. A valid IP address is a value other than `0.0.0.0`.

To find your phone's IP address:

- » Do one of the following:
 - Launch the **Product Info App** from the **Main** menu.
 - Go to **Settings > Network**.
 - Dial `***` and then select option `1` to enter the **Voice Admin** menu.

Local Device Configuration

OBi Edition VVX business IP phones have an integrated device management web portal that you can access from any standard web browser. The device management web page is organized into sections to allow for a manageable and compartmentalized approach to configuring parameters available on your phone.

Access the Device Management Web Portal

You can access and sign into the device management web page as a user or admin.

To access the Device Management web page:

- 1 Connect your phone to the LAN.
- 2 From the phone **Main** menu, select **Settings**.
- 3 Under **Settings**, the first item, **Network**, shows the IP address of the phone.
- 4 Enter the phone **IP Address** as the URL of the web site you want to visit in your web browser.
- 5 When prompted, enter your user name and password.

Navigating the Device Management Web Portal

There are many configurable parameters available on the phone, organized into a number of device configuration web pages. By browsing through the web pages, you can view all the parameters that you can configure and read or set their values.

Each web page is divided into three frames:

- A customizable top frame
- A left frame that lists the links to available pages
- A main frame that shows the parameters of the currently selected page

Use the expandable or collapsible menu tree on the left side of the page to navigate the various configuration parameter sections of your phone.



You must submit changes made on each configuration page before moving to another page. Otherwise, you lose any changes you made when you navigate to another page.

Most changes require a restart of the phone to take effect. However, you can restart the phone just once after you have made and submitted all the necessary changes on all the pages.

The following table lists the available device configuration web pages:

Device Configuration Web Pages

Configuration Page	Available Settings
Status	System Status Call Status SP Services Stats
OBiWiFi Configuration	Wi-Fi Settings Wi-Fi Scan
System Management	WAN Settings Auto Provisioning Device Admin Device Update
Service Providers	ITSP Profile A (repeated for ITSP Profile B, C, D, E, and F) <ul style="list-style-type: none"> • General • SIP • RTP
Voice Services	SP1 Service SP2 Service SP3 Service SP4 Service SP5 Service SP6 Service OBiTALK Service Auto Attendant Gateways and Trunk Groups

(continued) Device Configuration Web Pages

Configuration Page	Available Settings
IP Phone	Phone Settings Line Keys — on right side of phone screen (available only on VVX 450, OBi Edition) Left Line Keys Programmable Keys Softkeys LED Settings LDAP Feature Key Customization
Codec Profiles	Codec Profile A Codec Profile B
Tone Settings	Tone Profile A Tone Profile B
Ring Settings	Ring Profile A Ring Profile B
Star Codes	Star Code Profile A Star Code Profile B
User Settings	User Preferences Speed Dials User-defined Digit Maps

OBiTALK Portals

OBiTALK.com is a device management portal website that serves Polycom customers and service providers deploying OBi Edition devices. OBiTALK.com uses remote provisioning to manage OBi Edition devices. It stores or dynamically generates on demand a configuration file for each managed device and periodically checks in with the OBiTALK server for configuration updates.

There are two levels of management portals at OBiTALK.com: User and ITSP.

User Portal

Users can add and manage one or more OBi Edition devices from an OBiTALK account. The portal has setup wizards that help users configure voice services on any of their devices. Users can also see the detailed status and current parameter values of their devices and change settings on multiple devices from within the portal. The user portal has an upper limit of 20 devices per portal instance.

The user portal also enables members to add people and associated OBi Edition VVX phones to *circles of trust*. These circles of trust enable authorized users to share additional functionality.

Device Management Platform Portal – for ITSPs

Service providers, system integrators, and value-added resellers deploying OBi Universal Adapters and IP Phones can use the Device Management Platform portal to manage all their OBi endpoints remotely, enabling fast and easy device deployment and maintenance.

Service providers and system integrators can add OBi Edition devices to the OBiTALK portal for secure and remote provisioning, real-time call status reporting and troubleshooting, device UI management, and more.

Record Version Information

Contact partner@obihai.com to learn more about the OBiTALK DMP, and how to sign up to receive access.

In case you need to contact Polycom technical support, you should record the following information for future reference:

- Phone models
- Software version
- Partner Platform

Supported Network Configurations

OBi Edition VVX phones offer two interfaces for networking: Ethernet (described as “WAN” in the configuration) and Wi-Fi. You can use both interfaces at the same time, but Ethernet takes precedence.

Configuring the External Ethernet Ports

You can configure the two external Ethernet ports from the phone’s local configuration web pages.

To configure the external Ethernet ports:

- 1 Access the local phone web portal and log in with admin credentials.
- 2 Go to **System Management > WAN Settings**.
- 3 Configure the following parameters:

External Ethernet Port Configuration

Parameter	Values
<i>WAN Settings – Switch Port::Speed</i> <i>WAN Settings – PC Port::Speed</i>	<ul style="list-style-type: none">• Auto• 100 Full• 100 Half• 10 Full• 10 Half• 1000 Full• Disabled <p>Polycom recommends using only the <code>Auto</code> or <code>Disabled</code> settings.</p>

Configuring the WAN Interface

The WAN interface on the phone refers to the internal Ethernet switch port connected directly to the phone processor. You can configure the following setting groups on the phone.

WAN Interface Setting Groups

Setting Group	Description
VLAN	The phones support VLAN tagging in compliance with 802.1p/q. If you enable WAN Settings – Internet Settings::VLANEnable , the phone tags outbound traffic according to the VLANID and VLANPriority parameters. The phones ignore inbound traffic that doesn't belong to the same VLAN.
LLDP	The phones support LLDP-MED to automatically discover Network Policy (VLAN and DSCP) settings and perform other related handshake functions. Enable this feature using the WAN Settings – Internet Settings::LLDP-MED parameter.
IP Address Assignment	The phones support different methods of acquiring an IP address assigned to its WAN interface. Configure the method using the WAN Settings – Internet Settings::AddressingType parameter, which can have one of the following values: <ul style="list-style-type: none"> • DHCP: Request address assignment from a DHCP server. • Static: Use the statically assigned IP address, subnet mask, and default gateway from the WAN Settings – Internet Settings::IPAddress, SubnetMask and DefaultGateway parameters, respectively.
DNS Servers	You can specify as many as two DNS servers to use with the WAN interface using WAN Settings – Internet Settings::DNSServer1 and DNSServer2 . Note that if the DHCP offer includes DNS Servers, the phone takes as many as 16 servers from the list and uses them together with the explicitly configured servers.

Configuring the Wi-Fi Interface

The VVX 250, VVX 350, and VVX 450 business IP phones support Wi-Fi via the Polycom Wi-Fi wireless network adapter. You can connect the adapter to one of the USB ports on the phone.

VLAN and LLDP features aren't available on Wi-Fi.

Wi-Fi Interface Setting Groups

Setting Group	Description
IP Address Assignment	The phones support different methods of acquiring an IP address assigned to its Wi-Fi interface. Configure the method using the WiFi Settings – Basic Settings::AddressingType parameter, which can have one of the following values: <ul style="list-style-type: none"> • DHCP: Request address assignment from a DHCP server. • Static: Use the statically assigned IP address, subnet mask, and default gateway from the WiFi Settings – Internet Settings::IPAddress, SubnetMask, and DefaultGateway parameters, respectively.
DNS Servers	You can specify as many as two DNS servers to use with the Wi-Fi interface using WiFi Settings – Internet Settings::DNSServer1 and DNSServer2 . Note that if the DHCP offer includes DNS Servers, the phone takes as many as 16 servers from the list and uses them together with the explicitly configured servers.

DHCP Options

The phones support the following DHCP options for both networking interfaces:

- 42
- 66
- 150
- 159
- 160
- 161

The options that the phone tries to extract from DHCP offer is a comma-separated list of option numbers specified in the **WAN Settings – DHCP Client Settings::ExtraOptions** parameter. Note that the phone won't recognize any option numbers other than the supported ones listed above. You can use the macros `$DHCPOPT66`, `$DHCPOPT150`, `$DHCPOPT159`, `$DHCPOPT160`, and `$DHCPOPT161` to refer to the values of these options in any of the configuration parameters. For example, the default value of **Auto Provisioning – ITSP Provisioning::ConfigURL** is `tftp://$DHCPOPT66/$DM.xml`.

DNS Lookup

The DNS behavior described below applies to both network interfaces.

Lookup Order

In cases where there are multiple DNS servers available, the phone attempts to resolve a domain name quickly by querying as many DNS servers as necessary. You can insert a short delay between trying each DNS server sequentially such that the querying stops as soon as any of the servers receive a positive response. You can configure this desired short delay in seconds in **WAN Settings – DNS Control::DNSQueryDelay**. When you set the delay to 0, all the DNS servers are queried at the same time. In cases where there are DNS servers obtained from DHCP and from statically configured values, you can control the order of querying the two groups of servers through the **WAN Settings – DNS Control::DNSQueryOrder** parameter. Essentially you can choose to query the statically configured DNS servers first, or the DHCP supplied DNS servers first.

Locally Configured DNS Lookup Table

You can define as many as 30 local DNS records in the phone configuration to enable the phone to search through these 30 records before hitting the external DNS services when attempting to resolve a domain name. These records can be A or SRV records. This feature is particularly useful when you want to enable proxy redundancy without using any DNS servers. Note that the only way to provide a list of redundant servers to the phone is through the use DNS A or DNS SRV records.

NTP Servers and Local Time

The phone keeps track of current time by querying NTP servers (using SNTP). You can configure as many as two NTP servers using the **NTPServer1** and **NTPServer2** parameters.

By default, the phone discovers the SNTP server using DHCP option 42, and discovered servers take precedence. To disable SNTP discovery in DHCP and use the configured SNTP servers, remove 42 from the DHCP extra option list.

The local time is determined by the local time zone set in the **LocalTimeZone** parameter. The phone queries the NTP servers once per hour to update the current time, which is interpolated by the phone using its own local clock in-between NTP refreshes.

You can enable Daylight Saving Time (DST) by enabling the **DaylightSavingTimeEnable** parameter. DST is automatically adjusted based on the start and end rules specified in **DaylightSavingTimeStart** and **DaylightSavingTimeEnd** parameters. You can configure the amount of time to adjust when daylight saving time is in effect using the **DaylightSavingTimeDiff** parameter.

You can also enable the phone to use SIP Signaling to tell time. When the phone renews registration with a SIP proxy server, the server may include a Date header in the response to the phone that indicates the current GMT time. The phone processes this time value the same way it does with the result from the NTP servers, if **ITSP Profile X – SIP::X_ProcessDateHeader** is enabled.

Manually Configuring Phones

There are several ways to configure and manage your phone. Use the methods that best suit your deployment scenario.

Telephone-IVR-Based Local Configuration

The phone has a built-in interactive voice response (IVR) for checking and setting a small but essential subset of configuration parameters. Configuration via the IVR is a legacy configuration method inherited from older OBi products that don't have a display. It's included here nevertheless for additional convenience and also so that customers who are already familiar with using the OBi IVR can perform basic configuration tasks without learning new specifics about the phone first.

The IVR is, in essence, an instance of an automated attendant (AA). The OBi Edition VVX phone offers two instances of AA; referred to as AA1 and AA2. The IVR for configuration purposes is AA2, which we'll just refer to as the IVR to avoid confusion with AA1, which is the AA used to handle phone calls.

To invoke the IVR, pick up the phone, dial * * *, and follow the announced instructions. In order for the * * * number to work, make sure the digit map pattern * * * is included in the **Phone Settings::DigitMap** parameter, and the rule, {***:aa2} is included in the **Phone Settings::OutboundCallRoute** parameter. The standard (non-customized) default values of these parameters are, respectively:

```
([1-9]x?* (Mpli) | [1-9]S9 | [1-9] [0-9]S9 | *** | **0 | **1 (Msp1) | **2 (Msp2) | **3 (Msp3) | **4 (Msp4) | *
*9 (Mpp) | (Mpli))
and
{ ([1-9]x?* (Mpli)) :pp}, {**0:aa}, {***:aa2}, { (<**1:> (Msp1)) :sp1},
{ (<**2:> (Msp2)) :sp2}, { (<**3:> (Msp3)) :sp3}, { (<**4:> (Msp4)) :sp4},
{ (<**9:> (Mpp)) :pp}, { (Mpli) :pli}
```

For the meaning of these values, please see the [User-defined Digit Maps](#) section and the [Call Routing](#) chapter. Some parameter changes require a reboot to take effect. Changes made from the OBiTALK Configuration web page trigger the phone to reboot automatically (after any current calls end).

Key Ahead: By pressing the appropriate button sequence on the telephone, you can barge into the next menu of the IVR or invoke a command without waiting for the previous announcement to end.

Main Menu

The Main menu after starting the IVR is a list of operations that users can select by entering the corresponding 1-digit option number (as listed below):

Integrated Voice Response Operations

Selection	Announcement	What Can You Do?
1	Basic Network Status Reads the IP address and DHCP status.	Press 0 to repeat the information.
2	Advanced Network Status Reads the primary and back-up DNS server, primary and back-up NTP server.	Press 0 to repeat the information.
3	DHCP Current Value Reads the current value and you have the option to change the value.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information.
4	IP Address Current Value Reads the current value and you have the option to change the value. If you elect to enter a new value (static IP address), DHCP is disabled.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information.
5	Password Current Value Reads the current IVR password value and you have the option to change the value.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information.
6	Please Wait (while OBi is checking for software update)... One of the following messages plays: <ul style="list-style-type: none"> • Software Update Available. Press 1 to update software • Software Update Not Available. 	If an update is available, press 1 to proceed with the update. The software update process starts as soon as you hang up the phone. Warning: Once the software upgrade process starts, the device's power LED blinks rapidly. Please make sure the power and network cable stay connected to the unit until the process is complete.
8	Restore Factory Default	Press 1 to confirm device restore to factory default settings. Press # to return to device configuration menu. Press ## to exit IVR.
9	Reboot OBi Device	Press 1 to confirm device reboot. Press # to return to device configuration menu. Press ## or hang up to exit IVR.
0	Additional Options Access other configuration options of your phone.	Enter option followed by the # key.

Additional Integrated Voice Response Operations Options

There are many additional options beyond the top-level Integrated Voice Response Operations options 1 – 9. Unlike the top-level options, however, the list of available additional options in Menu 0 isn't announced.

To access the additional IVR options, do the following:

- 1 Enter ***** 0 #**
- 2 Enter your desired option number, followed by **#**
- 3 Follow any additional prompts, as desired.

You must enter the corresponding option number (followed by a **#** key) to select the particular option. The following tables list the available additional options (grouped by function):

Syslog Operation Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
1	Firmware Version Reads the current value of the firmware version.	Press 0 to repeat the information. Press # to enter another configuration selection.
2	IVR Password Reads the current value of the IVR password.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
3	Debug Level Reads the current value of the debug level.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
4	Syslog Server IP Address Reads the current IP address of the syslog server.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
5	Syslog Server Port Reads the current value of the syslog server port.	Press 1 to enter a new value. Press 2 to set the default value of 514. Press 0 to repeat the information. Press # to enter another configuration selection.

WAN Settings Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
20	DHCP Configuration Reads the current value of the DHCP configuration.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
21	IP Address Reads the current value of the IP address.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
22	Default Gateway Reads the current value of the default Internet gateway.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
23	Subnet Mask Reads the current value of the subnet mask.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
24	DNS Server (Primary) Reads the current value of the primary DNS server.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
26	NTP Server (Primary) Reads the current value of the primary NTP server.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

SP1 Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
100	Enable Service Provider One (SP1) Reads the current value.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
101	Registration State of SP1 Reads the current value.	Press 0 to repeat the information. Press # to enter another configuration selection.
102	SP1 User ID Reads the current value.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
167	SP1 Block Caller ID Enable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
168	SP1 Block Anonymous Call Enable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
172	SP1 Call Forward ALL – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
173	SP1 Call Forward ALL Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
174	SP1 Call Forward on Busy – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
175	SP1 Call Forward on Busy Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

SP1 Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
176	SP1 Call Forward on No Answer – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
177	SP1 Call Forward on No Answer Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

SP2 Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
200	Enable Service Provider Two (SP2) Reads the current value.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
201	Registration State of SP2 Reads the current value.	Press 0 to repeat the information. Press # to enter another configuration selection.
202	SP2 User ID Reads the current value.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
267	SP2 Block Caller ID Enable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
268	SP2 Block Anonymous Call Enable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

SP2 Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
272	SP2 Call Forward ALL – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
273	SP2 Call Forward ALL Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
274	SP2 Call Forward on Busy – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
275	SP2 Call Forward on Busy Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
276	SP2 Call Forward on No Answer – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
277	SP2 Call Forward on No Answer Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

OBiTALK Service Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
900	Enable OBiTALK Service Reads the current value.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
901	Registration State of OBiTALK Reads the current value.	Press 0 to repeat the information. Press # to enter another configuration selection.
967	OBiTALK Block Caller ID Enable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
968	OBiTALK Block Anonymous Call Enable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
972	OBiTALK Call Forward ALL – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
973	OBiTALK Call Forward ALL Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
974	OBiTALK Call Forward on Busy – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
975	OBiTALK Call Forward on Busy Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

OBiTALK Service Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
976	OBiTALK Call Forward on No Answer – Enable / Disable	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.
977	OBiTALK Call Forward on No Answer Number	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

Auto Attendant Options

Selection (Always Press “#” After Entering Selection)	Announcement	What Can You Do?
80	Enable / Disable Auto Attendant.	Press 1 to enter a new value. Press 2 to set the default value. Press 0 to repeat the information. Press # to enter another configuration selection.

Phone UI

You can configure a limited number of phone configuration settings directly from the phone UI. The most essential ones pertain to getting the phone connected the network, such as IP address settings or Wi-Fi settings.

Settings

You can configure the contents of numerous parameters from the phone UI under the **Settings** option of the main menu. For a complete list, see the [Polycom VVX Business IP Phones, OBi Edition User Guide](#), or access the phone native web portal or OBiTALK.com.

Settings groups include:

- Network
- Wi-Fi

- Programmable Keys
- Line Keys — on right side of phone screen (available only on VVX 450, OBi Edition)
- Left Line Keys
- Voice Services
- Speed Dials 99
- Device Administration

Preferences

You can configure the contents of these preference parameters:

- Language
- Skin
- Background Picture
- Ringtone
- Screen Saver
- Screen Saver Delay in Seconds
- Screen Save Type
- Screen Brightness
- Preferred Audio Device
- Preferred Headset Device
- Do Not Disturb
- Do Not Ring
- Call Forward
- Call Waiting
- Block Anonymous Call
- Anonymous Call
- Auto Answer Page
- Join Page Group 1
- Joint page Group 2
- Ringer Volume
- Speakerphone Volume
- Speakerphone Mic Gain
- Handset Volume
- Handset Mic Gain
- RJ9 Headset Volume
- RJ9 Headset Mic Gain
- Equalizer
- Acoustic Echo Cancellation

Provisioning Phones

The OBiTALK Device Management Platform portal provides access to the configurations of the multiple endpoints in batch mode by defining XML base profiles that are applied to multiple units. When a base profile changes, all units using this base profile are updated automatically with the changes.

Remote Provisioning

The phone can download a configuration file from a server located in the cloud or in the same enterprise. The configuration file may contain all the necessary parameter values for the phone to function normally. The file may also tell the device to download an additional configuration file from a different URL or to download a different firmware to replace the current one. The configuration file format and parameter naming conventions are proprietary to Polycom but are common across all OBi Edition products.

There are currently two configuration file formats supported: full XML format and short XML format with the XML tags substituted with single-letter abbreviations. The XML structure and parameter naming convention closely follows TR-104. For a full description of the configuration file and parameter names, refer to the [Polycom® OBi Edition Deployment Guide](#).

Similar to parameter groupings in device configuration web portal, parameters in the configuration file are grouped into a number of configuration objects for remote provisioning. In fact, you'll find a near one-to-one correspondence between these objects and their location within the configuration web pages.

To illustrate this, consider the web page SP1 Service, the SIP Credentials section:

SIP credentials

SIP Credentials

Parameter Name	Value	Default	
AuthUserName	<input type="text" value="join.j.smith@gmail.com"/>	<input type="checkbox"/>	?
AuthPassword	<input type="password" value="*****"/>	<input type="checkbox"/>	?
URI	<input type="text"/>	<input checked="" type="checkbox"/>	?
X_MyExtension	<input type="text" value="16188"/>	<input type="checkbox"/>	?
X_XsiUserName	<input type="text"/>	<input checked="" type="checkbox"/>	?
X_XsiPassword	<input type="password"/>	<input checked="" type="checkbox"/>	?
X_XmppDomain	<input type="text"/>	<input checked="" type="checkbox"/>	?
X_XmppUserName	<input type="text"/>	<input checked="" type="checkbox"/>	?
X_XmppPassword	<input type="password"/>	<input checked="" type="checkbox"/>	?
X_ContactUserID	<input type="text"/>	<input checked="" type="checkbox"/>	?
X_EnforceRequestUserID	<input type="checkbox"/>	<input checked="" type="checkbox"/>	?

The corresponding configuration object in a phone configuration XML file is **VoiceService.1.VoiceProfile.1.Line.1.SIP**, as shown below:

```
<Object>
  <Name>VoiceService.1.VoiceProfile.1.Line.1.SIP.</Name>
  <ParameterValueStruct>
    <Name>AuthUserName</Name>
    <Value>john.j.smith@gmail.com</Value>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>AuthPassword</Name>
    <Value>zYz123#$12</Value>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>URI</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_MyExtension</Name>
    <Value>16188</Value>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XsiUserName</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XsiPassword</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XmppDomain</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_XmppUserName</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_ContactUserID</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
  <ParameterValueStruct>
    <Name>X_EnforceRequestUserID</Name>
    <Value X_UseDefault="Yes"/>
  </ParameterValueStruct>
</Object>
```

Note that the dot (.) at the end of the object name is part of the name that cannot be omitted in the XML file. You must use the correct object name to create a valid configuration file for the phone.

3CX Plug-and-Play Provisioning

3CX plug-and-play provisioning lets the phone broadcast a SIP Subscribe to the Event: ua-profile on each reboot. With 3CX enabled, the phone also listens to SIP Notify of the same Event to extract from it the provisioning URL to request configuration, but only if the current **ITSP Provisioning – ConfigURL** parameter is still set to the default value. To enable 3CX plug-and-play provisioning, change the X_Enable3CXPNP parameter to false on the device web page.

Configuring Security Options

You can download and add custom certificates to the phones. All certificates must be in DER or PEM format. In case of (EAP) TLS, the client certificate file must also include the private key (PEM) file appended to the client certificate. For more information on supported certificates, refer to the Certificate Updates for Polycom VVX Business IP Phones, OBi Edition web page.

802.1X Authentication

The phones support the following 802.1X authentication modes:

- Disable
- MD5
- TLS
- TTLS/MSCHAPv2
- PEAP-MSCHAPv2 (optional for all parameters)

You can set the authentication mode using the **WAN Settings – Internet Settings::802_1XMode** parameter. Depending on the selected mode, you have to configure the additional authentication parameters listed in the following table.

802.1X Authentication

Parameter	Description	(EAP) MD5	(EAP) TLS (1.0)	TTLS/MSCHAPv2
802_1XIdentity	A user name. If the value isn't needed, set the value as an empty string.	Required	Required	Required
802_1XPassword	A password or passphrase. If a password or passphrase isn't needed, set the value as an empty string.	Required	Required	Required
802_1XAnonymousID	When empty, anonymous identity isn't used in authentication.		Required	Required
802_1XTLSecurityProfile	Security profile for the 802.1x authentication.		Required	

Upgrading the Software

This section summarizes the administrative features of your phone.

Updating the Firmware

You may be required to occasionally update the firmware on your phone to receive new features, enhancements, and bug fixes. The VVX Business IP phones are designed to make firmware updates a simple task.

Update Firmware with IVR

The easiest way to update the phone's firmware is by dialing *** then pressing option 6. If a firmware update is available, you're prompted to press 1 to continue and then hang up.

After hanging up, the download and update process completes in a couple of minutes over a fast internet connection. This can take much longer over legacy connectivity such as DSL, T1 and so on. The headset button starts to flash once the firmware download is complete and the phone prepares to reboot.

Update Firmware from Phone User Interface

Launch the **Settings** App from the Main menu and press the **F/W Update** softkey. A message pops up on the screen to describe if new firmware is available or not. If available, press the **Update Now** softkey. Do not power off the phone until the update completes and the phone automatically reboots immediately afterward.

Update Firmware from the Native Web Portal

Download the firmware file from OBiTALK.COM and store it on your computer. Then visit the phone's native web page from a web browser on the same computer and click the **Firmware Update** link on the left side panel of the landing page. On the firmware update page, enter the path to the firmware file (or click **Browse** to locate the firmware file on the computer).

Automated Firmware Update

Set up rules in the configuration to automatically check and download new phone firmware from a server.

Background Firmware Update

Background firmware update is supported in VVX 250, VVX 350, and VVX 450. Once enabled, the phone handles the firmware update without going through the polyupdater, and the update process occurs in the background during normal operation. The phone can still be used normally. After the new firmware is copied to flash memory, the phone auto-reboots to switch to use the new version as soon as the phone is not in a call. With this feature enabled, you also can install new firmware directly from the phone web page by selecting the firmware file stored on a local disk drive.



This feature is not available on the VVX150.

Two Firmware File Formats: *sip.ld* and *.fw*

The current VVX firmware file format is the `sip.ld` format, which is used by both UCS and OBi Edition. To allow firmware update in the background, the new firmware to update to must use a different firmware format which is referred to as the `.fw` format. The `.fw` format is supported only when the background firmware update feature has been enabled. When this update feature has been enabled, the `sip.ld` format is no longer supported.

Like the `sip.ld` format firmware, Polycom generates the `.fw` format for each build, signed with a Polycom Release Key. For example, the following firmware file may be generated:

- `Skyline-6-3-1-25001-release-2018-12-13.fw` (signed with Release Key)

But unlike the `sip.ld` format firmware, which requires a separate build for each phone model, the same `.fw` format firmware file can be applied to all three phone models: VVX250, VVX350, and VVX450. The filename of a `.fw` image can be anything and must be explicitly specified in provisioning rules.

Enabling Background Firmware Update



If you purchase a phone with software version 6.3.1 or greater, background firmware update is enabled by default.

Enabling background firmware update requires a one-time format transition process. This process takes about 4 to 5 minutes to complete and the phone is not usable during this period. It is highly recommended to not to disconnect the power during the process. You can refer to the **BackgroundUpdate** parameter status on the Status/System Status web page (Production Information section) to check whether background firmware update feature has been enabled on the phone.

You can start this transition process by appending the `,bgu-conv=enable` keyword to the **FirmwareURL** parameter, such as:

```
http://192.168.1.1/backgroundupdate/,bgu-conv=enable
```

As soon as the device verifies that the firmware in the `http://192.168.1.1/backgroundupdate/` path is current, it starts the process automatically with a notification message displayed on the phone screen until the process is done.

Once the transition process is completed, the phone auto reboots and factory resets. All existing configuration, call logs, and phone book are lost. The **BackgroundUpdate** parameter is also changed to `Enabled`. In case the procedure is interrupted or failed, you will have to use Polyupdater to recover the device. It is sometimes necessary to reinstall a `sip.ld` formatted image into the device.

Using the Background Firmware Update Feature

After the feature is enabled, the device only takes a `.fw` formatted image. Also, when using auto firmware update during provisioning, a complete URL including the firmware file name must be given in the **FirmwareURL** parameter. The phone does NOT append `partnumber.sip.ld` to the path when downloading the image. Failure to include the `.fw` firmware filename results in a 404 or `File not found` error. The following is an example of a correct URL:

```
http://192.168.1.1/Skyline-6-3-1-25001-devel-2018-12-13.fw
```

When installing a `.fw` firmware from the device web page, the phone will NOT take a Config XML file as firmware file any more, as is the case of a `sip.ld` file. You must provide the path to the firmware file directly. If the given file is not a valid `.fw` firmware file, the firmware update procedure fails.

Reverting Back to sip.ld

Reverting back to use `sip.ld` firmware is equivalent to disabling background firmware update (to a `.fw` firmware). This requires a one-time format (inverse) transition process on the device that has background firmware update enabled. To do this, specify a complete URL including the firmware filename of a corresponding `sip.ld` formatted image corresponding to the phone model in use in the **FirmwareURL** parameter with the special keyword `bgu-conv=disable` appended. For example:

```
http://192.168.1.1/ucs/3111-48840-001.sip.ld,bgu-conv=disable
```

When the phone finds the special keyword in **FirmwareURL**, it first downloads the firmware file header according to the given URL and makes sure it is a valid `sip.ld` firmware image. Then it saves the firmware update URL into the Polyupdater and marks a flag to let the Polyupdater restore the flash partition on the next boot, and then it auto-reboots. On the next boot up, the Polyupdater proceeds with reverting to the original (UCS style) flash partitioning and subsequently upgrades to the given `sip.ld` firmware. Due to flash repartitioning, the phone configuration is factory reset and all existing configuration, call logs, phone book, and so forth are lost as a result.

Customization Data Auto Update

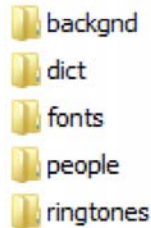
You can customize your phone in many ways. Some customization features require extra data such as background pictures, dictionary, fonts, and so forth. The customization data is categorized into two levels: Service Provider (ITSP) Level and User Level. The data for each level is allocated at dedicated location inside of internal storage in the phone. The auto update customization of data only manages the data at the ITSP level.

Customization Data Package

Your phone can accept data only in the form of a tar file, which is called Phone Customization Data Package. This tar file can be either gzipped or not gzipped. The size of tar file must be smaller than 30 MB.

Inside the tar file, the data should be organized in the following directory hierarchy, as your phone searches certain data from the certain subdirectory. For example, the power-up logo must be named `logo.raw` and is allocated under the `backgnd` subdirectory.

Folders for custom data



Example Linux command to generate a package named `itsp20141001.tar.gz` assuming the above directory hierarchy is under a working directory called `itspdata`:



You must issue this command inside of this working directory (`itspdata`), as your phone can't strip the first level of the tar file.

```
% tar -zcvf ../itsp20141001.tar.gz *
```

Example Linux command to generate the MD5 checksum:

```
% md5sum itsp20141001.tar.gz
```

Auto Update Operation

Similar to the Auto Firmware Update feature, the Customization Data Auto Update feature is configured by a set of parameters. When it is enabled, your phone attempts to download the Customization Data package according to the scripts specified in the **DownloadURL** parameter. Then the data in the package is validated by the given MD5 checksum before it is installed to internal storage. As soon as data installation successfully completes, your phone restarts itself (warm boot) and won't try to download the package again until the MD5 checksum is changed, even if the feature is still enabled.

The Auto Update operation is always performed after firmware update and all configuration provisioning are completed.

Auto Update Configuration Parameters

This feature is configured by the following parameters.

Auto Update Configuration Parameters

Parameter	Description	Default Setting
Method	<p>Current operational method of Provisioning. Choice of:</p> <ul style="list-style-type: none"> • <code>Disabled</code> = Do not download from DownloadURL • <code>System Start</code> = Download from DownloadURL just once on system start • <code>Periodically</code> = Download from DownloadURL on system start, and then periodically at the interval specified in the Interval parameter <p>Note: First download on system start is performed after firmware update and configuration provisioning are complete.</p>	Disabled
Interval	When Method is set to <code>Periodically</code> , this is the number of seconds between download from DownloadURL . If value is 0, device downloads once only on system start (equivalent to setting Method to <code>System Start</code>).	0
DownloadURL	URL of Customization Data package.	
MD5Checksum	Standard MD5 checksum (hexadecimal string) of the Customization Data package.	
Incremental	When enabled, the Customization Data package is installed incrementally without erasing the old data.	false
DnsLookupType	<p>Choice of:</p> <ul style="list-style-type: none"> • A Record Only • SRV Record Only • Try Both 	A Record Only
DnsSrvPrefix	<p>Choice of:</p> <ul style="list-style-type: none"> • No Prefix • With Prefix • Try Both 	No Prefix
Username	Optional Username for authentication if URL scheme is <code>http://</code> or <code>https://</code> .	
Password	Optional Password for authentication if URL scheme is <code>http://</code> or <code>https://</code> .	

Diagnostics and Status

OBi Edition VVX business IP phones include a variety of information screens and tools that can help you monitor the phone and resolve problems.

Factory Reset

Resetting all configuration parameter to factory default values or to the customized default values for phones that are customized with some non-generic parameter values.

You can access the Factory Reset function from the native web portal or from the phone UI under the **Settings** App.

Phone Display and Appearances

The phone has built-in Skins that dictate the look and feel of the phone UI. Users can select a particular skin via the **Preferences** menu on the phone. You can also download a custom skin onto the target phone. You can request an additional document with UI customization details from Polycom.

LED Pattern Customization

Feature Key LED patterns for each well-defined state can be customized using the LED Settings in the configuration. Each customizable pattern is configured in its own parameter as a comma-separated list of {Color} [{Duration}] pairs, where {Color} = R (for red), G (for green), or X (for off). The optional {Duration} part is the number of milliseconds to show the given color. If {Duration} is not specified, the LED stays at the given color indefinitely. The entire pattern is played repeatedly from left to right indefinitely until the state changes. Here are some examples:

X - Steady off

R - Steady red

R500,X500 - 500 ms red followed by 500 ms off

G50,X50,G50,X1000 - 50 ms green, 50 ms off, 50 ms green, 1s off

LED Settings Parameters

LED Settings Parameters

Parameter Group	Parameter	Description
Call State		
For a bound or unbound call key. The LED pattern reflects the local call state.		
<i>IP Phone – LED Settings – Call State</i>	Dialing	
<i>IP Phone – LED Settings – Call State</i>	Trying	
<i>IP Phone – LED Settings – Call State</i>	PeerRinging	
<i>IP Phone – LED Settings – Call State</i>	Connected	

LED Settings Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LED Settings – Call State</i>	Ringing	
<i>IP Phone – LED Settings – Call State</i>	Holding	
<i>IP Phone – LED Settings – Call State</i>	Error	
<i>IP Phone – LED Settings – Call State</i>	CallParked	
<i>IP Phone – LED Settings – Call State</i>	ServiceDown	

SCA State

For a call key that is bound to a shared line. The LED pattern reflects the SCA state when the user of the SCA is on another phone.

<i>IP Phone – LED Settings – SCA State</i>	Seized	
<i>IP Phone – LED Settings – SCA State</i>	Trying	
<i>IP Phone – LED Settings – SCA State</i>	PeerRinging	
<i>IP Phone – LED Settings – SCA State</i>	Connected	
<i>IP Phone – LED Settings – SCA State</i>	Held	
<i>IP Phone – LED Settings – SCA State</i>	PrivateHeld	
<i>IP Phone – LED Settings – SCA State</i>	ServiceDown	

BLF State

For feature key that is assigned the **Busy Lamp Field** function. The LED reflects the state of the monitored entity.

<i>IP Phone – LED Settings – BLF State</i>	Idle	
<i>IP Phone – LED Settings – BLF State</i>	CallParked	
<i>IP Phone – LED Settings – BLF State</i>	Ringing	

LED Settings Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LED Settings – BLF State</i>	Busy	
<i>IP Phone – LED Settings – BLF State</i>	Holding	
<i>IP Phone – LED Settings – BLF State</i>	ServiceDown	

Service State

For feature key that is assigned the **Line Monitor** function. The LED pattern reflects the state of the bounding voice service.

<i>IP Phone – LED Settings – Service State</i>	Idle	
<i>IP Phone – LED Settings – Service State</i>	InUse	
<i>IP Phone – LED Settings – Service State</i>	Ringing	
<i>IP Phone – LED Settings – Service State</i>	Holding	
<i>IP Phone – LED Settings – Service State</i>	ServiceDown	

ACD Agent State

For feature key that is assigned the **ACD Sign On or Off** function. The LED reflects the current ACD Agent state.

<i>IP Phone – LED Settings – ACD Agent State</i>	LoggedOff	
<i>IP Phone – LED Settings – ACD Agent State</i>	Available	
<i>IP Phone – LED Settings – ACD Agent State</i>	Unavailable	
<i>IP Phone – LED Settings – ACD Agent State</i>	WrappingUp	
<i>IP Phone – LED Settings – ACD Agent State</i>	ServiceDown	

Presence State

For feature key that is assigned the **Presence Monitor** function. The LED reflects the presence of the monitored entity.

<i>IP Phone – LED Settings – Presence State</i>	Offline	
-------------------------------------------------	----------------	--

LED Settings Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LED Settings – ACD Agent State</i>	Online	
<i>IP Phone – LED Settings – ACD Agent State</i>	Busy	
<i>IP Phone – LED Settings – ACD Agent State</i>	Away	
<i>IP Phone – LED Settings – ACD Agent State</i>	ExtendedAway	
<i>IP Phone – LED Settings – ACD Agent State</i>	ServiceDown	

Feature Key State

Miscellaneous feature key functions not covered in the above.

<i>IP Phone – LED Settings – Feature Key State</i>	AnonymousCallEnabled	For the feature key assigned to the Block Caller ID function. This is the LED pattern when the feature is enabled.
<i>IP Phone – LED Settings – Feature Key State</i>	AnonymousCallDisabled	For the feature key assigned to the Block Caller ID function. This is the LED pattern when the feature is disabled.
<i>IP Phone – LED Settings – Feature Key State</i>	AnonymousCallServiceDown	For the feature key assigned to the Block Caller ID function. This is the LED pattern when the service that provides the feature is down, if the feature is provided by the ITSP.
<i>IP Phone – LED Settings – Feature Key State</i>	AnonymousCallBlockEnabled	For the feature key assigned to the Block Anonymous Call function. This is the LED pattern when the feature is enabled.
<i>IP Phone – LED Settings – Feature Key State</i>	AnonymousCallBlockDisabled	For the feature key assigned to the Block Anonymous Call function. This is the LED pattern when the feature is disabled.
<i>IP Phone – LED Settings – Feature Key State</i>	AutoAnswerIntercomEnabled	For the feature key assigned to the Auto Answer Intercom function. This is the LED pattern when the feature is disabled.
<i>IP Phone – LED Settings – Feature Key State</i>	AutoAnswerIntercomDisabled	For the feature key assigned to the Auto Answer Intercom function. This is the LED pattern when the feature is disabled.

LED Settings Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LED Settings – Feature Key State</i>	CallForwardEnabled	For the feature key assigned to the Call Forward function. This is the LED pattern when the feature is enabled.
<i>IP Phone – LED Settings – Feature Key State</i>	CallForwardDisabled	For the feature key assigned to the Call Forward function. This is the LED pattern when the feature is disabled.
<i>IP Phone – LED Settings – Feature Key State</i>	CallForwardServiceDown	For the feature key assigned to the Call Forward function. This is the LED pattern when the service providing this feature is down.
<i>IP Phone – LED Settings – Feature Key State</i>	CallParkYes	For the feature key assigned to the Call Park Monitor function. This is the LED pattern when there is a call parked on that park orbit.
<i>IP Phone – LED Settings – Feature Key State</i>	CallParkNo	For the feature key assigned to the Call Park Monitor function. This is the LED pattern when there is no call parked on that park orbit.
<i>IP Phone – LED Settings – Feature Key State</i>	CallParkMonitorServiceDown	For the feature key assigned to the Call Park Monitor function. This is the LED pattern when the underlying call park monitoring service is down
<i>IP Phone – LED Settings – Feature Key State</i>	CallWaitingEnabled	For the feature key assigned to the Call Waiting function. This is the LED pattern when the feature is enabled.
<i>IP Phone – LED Settings – Feature Key State</i>	CallWaitingDisabled	For the feature key assigned to the Call Waiting function. This is the LED pattern when the feature is disabled.
<i>IP Phone – LED Settings – Feature Key State</i>	DoNotDisturbEnabled	For the feature key assigned to the Do Not Disturb function. This is the LED pattern when the feature is enabled.
<i>IP Phone – LED Settings – Feature Key State</i>	DoNotDisturbDisabled	For the feature key assigned to the Do Not Disturb function. This is the LED pattern when the feature is disabled.
<i>IP Phone – LED Settings – Feature Key State</i>	DoNotRingEnabled	For the feature key assigned to the Do Not Ring function. This is the LED pattern when the feature is enabled.
<i>IP Phone – LED Settings – Feature Key State</i>	DoNotRingDisabled	For the feature key assigned to the Do Not Ring function. This is the LED pattern when the feature is disabled.

LED Settings Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LED Settings – Feature Key State</i>	ExecFilterEnabled	For the feature key assigned to the Exec Filter on or Off function. This is the LED pattern when the feature is enabled.
<i>IP Phone – LED Settings – Feature Key State</i>	ExecFilterDisabled	For the feature key assigned to the Exec Filter on or Off function. This is the LED pattern when the feature is disabled.
<i>IP Phone – LED Settings – Feature Key State</i>	ExecFilterServiceDown	For the feature key assigned to the Exec Filter on or Off function. This is the LED pattern when the service providing this feature is down.
<i>IP Phone – LED Settings – Feature Key State</i>	ExecAssistEnabled	For the feature key assigned to the Exec Assistant function. This is the LED pattern when the assistant is not filtering calls for any executives (the executive list for this assistant is empty).
<i>IP Phone – LED Settings – Feature Key State</i>	ExecAssistDisabled	For the feature key assigned to the Exec Assistant function. This is the LED pattern when the assistant is filtering calls for at least one executive.
<i>IP Phone – LED Settings – Feature Key State</i>	ExecAssistDivertOn	For the feature key assigned to the Exec Assistant function. This is the LED pattern when the assistant has turned on the Divert option
<i>IP Phone – LED Settings – Feature Key State</i>	ExecAssistServiceDown	For the feature key assigned to the Exec Assistant function. This is the LED pattern when the service providing the Exec Assistant feature is down.
<i>IP Phone – LED Settings – Feature Key State</i>	HotelingGuestLoggedOn	For the feature key assigned to the Hoteling function. This is the LED pattern when a guest has logged on.
<i>IP Phone – LED Settings – Feature Key State</i>	HotelingGuestLoggedOff	For the feature key assigned to the Hoteling function. This is the LED pattern when no guest has logged on.
<i>IP Phone – LED Settings – Feature Key State</i>	HotelingServiceDown	For the feature key assigned to the Hoteling function. This is the LED pattern when the service providing this feature is down.
<i>IP Phone – LED Settings – Feature Key State</i>	NewMessagesWaitingYes	For the feature key assigned to the Message Status function. This is the LED pattern when there are new messages in that mailbox.

LED Settings Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LED Settings – Feature Key State</i>	NewMessagesWaitingNo	For the feature key assigned to the Message Status function. This is the LED pattern when there are no new messages in that mailbox.
<i>IP Phone – LED Settings – Feature Key State</i>	MWIServiceDown	For the feature key assigned to the Message Status function. This is the LED pattern when the MWI service for that mailbox is down.
<i>IP Phone – LED Settings – Feature Key State</i>	PageGroupJoined	For the feature key assigned to the Page Group 1 or Page Group 2 function. This is the LED pattern when the phone has joined the group.
<i>IP Phone – LED Settings – Feature Key State</i>	PageGroupLeft	For the feature key assigned to the Page Group 1 or Page Group 2 function. This is the LED pattern when the phone has left the group.
<i>IP Phone – LED Settings – Feature Key State</i>	PageGroupMeTalking	For the feature key assigned to the Page Group 1 or Page Group 2 function. This is the LED pattern when you are talking to the group.
<i>IP Phone – LED Settings – Feature Key State</i>	PageGroupThemTalking	For the feature key assigned to the Page Group 1 or Page Group 2 function. This is the LED pattern when you are listening and someone else in the group is talking.
<i>IP Phone – LED Settings – Feature Key State</i>	SecurityClassServiceDown	For the feature key assigned to the Security Class function. This is the LED pattern when the service providing this feature is down.
VMWI Lamp		
<i>IP Phone – LED Settings – VMWI Lamp</i>	NewMessagesWaitingYes	New messages for the phone from all mailboxes.
<i>IP Phone – LED Settings – VMWI Lamp</i>	NewMessagesWaitingNo	No new messages for the phone from all mailboxes.
<i>IP Phone – LED Settings – VMWI Lamp</i>	Disabled	VMWI is disabled.
<i>IP Phone – LED Settings – VMWI Lamp</i>	Ringng	Describes whether the LED should blink with the given pattern when the phone is ringing. Note that if the pattern is blank, ringing won't affect the VMWI Lamp.

UI Menu Customization

The parameters for customizing some UI menus are summarized in the following table. Each menu parameter is a comma-separated list of menu items, where each menu item is specified with an item id followed by an optional semicolon and an item display text separated by a semicolon. For example:

```
menu = item, item, ..., item
item = item-id;item-display-text
```

When multiple instances of the same menu parameters are specified, the items are concatenated internally into a single item list. Items are displayed in the order they are specified. If `display-text` is not specified for an item, the default displayed text is used.

UI Menu Customization

Parameter Group	Parameter	Description
Preferences Menu		
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>PreferencesMenu1</i>	For example: language;Language,timeFormat;Time Format,dateFormat;Date Format
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>PreferencesMenu2</i>	
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>PreferencesMenu3</i>	
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>PreferencesMenu4</i>	
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>PreferencesMenu5</i>	
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>PreferencesMenu6</i>	
Settings Menu		
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>SettingsMenu1</i>	For example: net;Network Settings,wifi;WiFi Settings
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>SettingsMenu2</i>	
Product Information Menu		
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>ProductInfoMenu1</i>	For example: model;Model,obinumber;OBi Number,mac;MAC Address
<i>IP Phone – Phone Settings – GUI Menus</i>	<i>ProductInfoMenu2</i>	
Main Menu		

UI Menu Customization

Parameter Group	Parameter	Description
IP Phone – Phone Settings – GUI Menus	MainMenu1	For example: phone-book;Contacts,calls;Current Calls;call-history;Call History
Net Services Menu		
IP Phone – Phone Settings – GUI Menus	NetServicesMenu1	For example: bci;Anonymous Call,dnd,cfa,cfb,cfna,acd,bwanw,rmoff,simring

Main Menu Item IDs

Main Menu Items

Main Menu Item ID	Default Caption	Description
phone-book	Contacts	Local phone book.
calls	Current Calls	List all the currently ongoing calls.
call-history	Call History	Locally stored call logs (for all calls on all services).
preferences	Preferences	Local phone feature preferences.
settings	Settings	List all the system level settings.
prod-info	Product Info	A list of data related to the product such as model and serial numbers, software and hardware versions.
netsrv	Net Services	Access the settings of many network provided features for each SP n service, $n = 1 - 6$.
netdir	Net Dir	Access the network directory associated with a specific SPI service or the LDAP service.
buddy	Buddy List	Access to the buddy list associated with a specific SP n service.
directories	Directories	A container that lists the following options: <ul style="list-style-type: none"> Local phone-book (same as Contacts) Network directory for each SPn service, when available, $n = 1 - 6$ LDAP Search (if configured) When only the local phone-book is available, this item behaves the same as the phone-book item.
messages	Messages	This option shows a list where each entry is the message or mailbox status of a SP n service, for $n = 1 - 6$, if enabled.

Main Menu Items

Main Menu Item ID	Default Caption	Description
all-preferences	Preferences	A container that lists the following options: <ul style="list-style-type: none"> Phone (same as preferences item) SPn service (or its label), for $n = 1 - 6$, if enabled If none of SP n preferences are enabled, then the item behaves exactly as the legacy preferences item.
call-histories	Call Histories	A container to access the locally stored call histories and the call history associated with each SP n service, $n = 1 - 6$.

Net Services Menu Item IDs**Net Services Menu Items**

Net Services Menu Item ID	Default Caption	Description
acd	ACD Agent Sign On/Off	
bci	Anonymous Call	Turn Anonymous Call feature on or off.
bwanw	BroadWorks Anywhere	
buddy	Buddy List	
ccs	Call Center Status	
cfa	Call Forward Always	
cfb	Call Forward Busy	
cfna	Call Forward No Answer	
rec	Call Recording Mode	
dnd	Do Not Disturb	
dispcode	Enter Call Disposition Code	
exec	Executive Call Filter	
xass	Executive Assistance	
hotel	Hoteling	
clog	Network Call History	
dir	Network Directory	
rmoff	Remote Office	

Net Services Menu Items

Net Services Menu Item ID	Default Caption	Description
secClass	Security Class	
simring	Simultaneous Ring	

Preferences Menu Item IDs**Preferences Menu Items**

Preferences Menu Item ID	Default Caption	Description
language	Language	Select the Language of the displayed text on the phone screen.
timeFormat	Time Format	Select the format of the time display on the phone screen.
dateFormat	Date Format	Select the format of the date display on the phone screen.
hpTime	Auto Home Page	Select the amount of idle time before automatically returning the phone screen to the Home screen.
skin	Skin	Select the built-in UI scheme ("skin") to use.
bgpic	Background Picture	Select an available background picture to use as wall paper.
dring	Default Ringtone	Select an available ringtone as the default ringtone.
dfont	Font	Select an available font as the default font for the displayed text.
appCols	Home App Columns	Select the number of columns to display in the UI Main menu.
packcalls	Pack Calls On Display	Enable or disable packing calls on different services on the same screen of the display.
sdim	Dim Screen	Enable or disable dimming the screen (to the least settable brightness level) after the set interval of no-key-presses-and-no-calls.
sdimDelay	Dim Screen Delay	Interval of no-key-presses-and-no-calls in seconds before dimming the screen, if enabled.
ssvr	Screen Saver	Enable or disable the screen saver feature.
ssvrDelay	Screen Saver Delay	Amount of idle time in seconds before starting the screen saver, if enabled.
ssvrType	Screen Saver Type	Select a screen saver type.

Preferences Menu Items

Preferences Menu Item ID	Default Caption	Description
ssvrLock	Require Passcode On Wake Up	Select whether a passcode is required to exit from the screen saver.
ssvrPass	Wake Up Passcode	Passcode to exit from screen saver.
	Screen Saver Show Custom Contents	
brightness	Screen Brightness	Set the brightness of the LCD screen.
audioDevice	Preferred Audio Device	Select whether to use speakerphone or a headset as the talk device when a headset is connected.
headsetDevice	Preferred Headset Device	Select the preferred headset device when more than one is connected.
ehs	Electronic Hook Switch	
dnd	Do Not Disturb	Enable or disable Do Not Disturb.
dnr	Do Not Ring	Enable or disable Do Not Ring.
cfa.enable	Call Forward	Enable or disable Call Forwarding (all calls, unconditionally).
cwa	Call Waiting	Enable or disable Call Waiting.
bac	Block Anonymous Call	Enable or disable Block Anonymous Caller.
bci	Anonymous Call	Enable or disable Block Caller ID (also known as Anonymous Call).
aans	Auto Answer Page	Enable or disable Auto-Answering incoming page.
pg1	Join Page Group 1	Join Page Group 1.
pg2	Join Page Group 2	Join Page Group 2.
ringerVol	Ringer Volume	Ringer Volume.
speakerVol	Speakerphone Volume	Speakerphone Audio Volume.
micGain	Speakerphone Mic Gain	Speakerphone Mic Input Gain.
handsetVol	Handset volume	Handset Audio Volume.
handsetGain	Handset Mic Gain	Handset Mic Input Gain.
headsetRJ9Vol	RJ9 Headset Volume	Audio Volume for the RJ9 headset device.
headsetRJ9Gain	RJ9 Headset Mic Gain	Mic Input Gain for the RJ9 headset device.
EqEnable	Equalizer	Enable the Equalizer.
AecEnable	Acoustic Echo Cancellation	Enable the Acoustic Echo Canceler.

Preferences Menu Items

Preferences Menu Item ID	Default Caption	Description
edialer	Enhanced Dialer	Enhanced Dialer (show matching list of recently dialed numbers while dialing).
dlkt	Default Line Key Tab	Default Line Key Tab (the Line Key Tab to fallback to after a configurable duration of idle time).

Settings Menu Item IDs**Settings Menu Items**

Settings Menu Item ID	Default Caption	Description
net	Network	Show current IP address and launch Network Settings menu.
wifi	WiFi	Show Wi-Fi connection status and launch Wi-Fi Settings menu.
ringfile	Ringtones	Set up the default ring tone per SP <i>n</i> service.
pkeys	Programmable Keys	Launch Programmable Keys Settings menu.
lkeys	Line Keys	Launch Line Keys Settings menu.
voice	Voice Services	Launch Voice Services Settings menu.
Sd99	Speed Dials 99	Set up 99 speed dial numbers.
clrcache	Clear Data Cache	Clear the downloaded data cache.
admin	Device Administration	Show Administrative Settings menu.
login	Unlock Admin Settings	Log in or out of administrative mode.

Product Info Menu Item IDs**Product Info Menu Items**

Product Info Menu Item ID	Default Caption	Description
model	Model	Model Number
obinumber	OBI Number	OBI Number
mac	MAC Address	(Ethernet) MAC Address
wfmac	WiFi MAC Address	Wi-Fi MAC Address
serial	Serial Number	Serial Number

Product Info Menu Items

Product Info Menu Item ID	Default Caption	Description
swver	Software Version	Software Version
hwver	Hardware Version	Hardware Version
zinfo	Customization Status	ZT Customization Status
uptime	Up Time	UP time since last bootup

Cache Control of Downloaded (Temporary) Data

The following data items can be downloaded and cached by the phone at run time (in DRAM):

- Ring tone file downloaded from the URL specified in **User Preferences::DefaultRingtone**
- Picture file downloaded from the URL specified in **User Preferences::BackgroundPicture**
- Caller ID Picture files downloaded from URL extracted from Call-Info header of SIP Messages
- Picture files downloaded from URL in src attribute of elements in rendered <ScreenItem>
- Var-Tree XML files downloaded from URL in value attribute of <setvar> elements in rendered <ScreenItem>
- Icon, Wave or MP3 files downloaded when rendering OBiPhoneXML Apps

The expiration time of each individually cached data file is taken from the HTTP/Cache-Control header in the 200 response when the file is originally received from the server. If the Cache-Control header is present in the 200 response with the no-cache flag specified, the downloaded data file is not cached. Otherwise, if the Cache-Control header is present with the max-age attribute specified, the data file is cached for the number of seconds as specified in the max-age attribute. Otherwise, the data file is cached indefinitely until one of the following happens that clears ALL the cached data:

- Phone is power cycled, or after a full reboot (such as after a firmware update)
- User selects **Clear Data Cache** on the **Settings** menu from the phone UI

Phone Customization Data

In addition to the configurable aspects of the UI discussed above, some of the data that drives the UI can be customized by the administrator of the phones. These data are collectively called Phone Customization Data which include the following data types:

Phone Customization Data Type

Customization Data Type	Description	Internal Data Storage Path
Background Pictures (Wall Paper)	The pictures that you can choose to show as wall paper during normal phone operation. The data files can be JPEG, PNG, BMP, or GIF.	/backgnd/
Dictionary	The translation to be used for various items shown on the phone screen.	/dict/
Text Fonts	The true-type font files for rendering text to display on the screen.	/fonts/
Ring Tones	Ring tone wave files. It must be in 16-bit mono linear PCM wave format sampled at 16 kHz.	/ringtones/

Background Pictures

Custom background pictures are stored in the internal folder **/backgnd/**. Each background picture should have its pixel dimension JPEG, PNG, BMP, and GIF files are the acceptable file formats. You can choose any of the pictures stored in this folder as background pictures (that is, wallpaper) on the phone screen.

You can browse the **Background Picture** entries under the phone's **Preferences** menu and select one of the available background pictures. Each of the built-in "Skins" has its own default background pictures.

Text Fonts Customization

Fonts are stored in the internal folder **/fonts/**. Custom font files can only be installed on the phone using the data package upload method. Acceptable font file formats are TTF and OTF. In order to choose an installed font, you must also add a font-name-to-font-file for each installed font in the `font.map` file that is also stored in the same folder.

`font.map` maps the font name to the actual font file. The set of font names listed in `font.map` is also presented to you as a list of selections under **Preferences/Font** in the phone UI. On the web page the **DefaultFont** parameter is just a string and should match one of the font names (not file names) listed in `font.map`.

The `font.map` file lists all the factory-installed fonts that come with the phone.

```
#
# font.map
#
# The left column is a font name that a user can select from
# the Preferences menu. The right column is the corresponding
# physical font file installed on the phone.
# A valid font file must either a ttf or otf file.
```

```
#
# You can use any number of white spaces as delimiters between the 2 columns.
#
aleo                Aleo-Light.otf
aleo-bold           Aleo-Bold.otf
aleo-italic         Aleo-LightItalic.otf
aleo-bold-italic   Aleo-BoldItalic.otf
banksia            Banksia-Regular.otf
banksia-bold       Banksia-Bold.otf
droidsans          DroidSans.ttf
droidsans-bold     DroidSans-Bold.ttf
libre-caslon       LibreCaslonText-Regular.ttf
libre-caslon-bold  LibreCaslonText-Bold.ttf
libre-caslon-italic LibreCaslonText-Italic.ttf
opensans           OpenSans-Regular.ttf
opensans-bold      OpenSans-Bold.ttf
opensans-italic    OpenSans-Italic.ttf
opensans-bold-italic OpenSans-BoldItalic.ttf
quattrocento      Quattrocento-Regular.ttf
quattrocento-bold  Quattrocento-Bold.ttf
terminal-dosis     TerminalDosis-Regular.ttf
terminal-dosis-bold TerminalDosis-Bold.ttf
```

In addition, there are four more built-in fonts:

```
ptsans
ptsans-bold
ptsans-italic
ptsans-bold-italic
```

Font Samples

Font	Sample
aleo	abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +*!/=
aleo-bold	abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +*!/=
aleo-bold-italic	<i>abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ</i> <i>1234567890.,; ' " (!?) +*!/=</i>
aleo-italic	<i>abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ</i> <i>1234567890.,; ' " (!?) +*!/=</i>
banksia	abcdefghijklmnopqrstuvwxyz ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +*!/=

Font Samples

Font	Sample
banksia-bold	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
droidsans	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
droidsans-bold	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
libre-caslon	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
libre-caslon-ita lic	<i>abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=</i>
libre-caslon-bo ld	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
opensans	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
opensans-italic	<i>abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=</i>
opensans-bold	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
opensans-bold -italic	<i>abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=</i>
quattrocento	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=
quattrocento-b old	abcdefghijklmnopqrstuvwxy z ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.;; ' " (!?) +-*/=

Font Samples

Font	Sample
terminal-dosis	abcdefghijklmnopqrstuvwxyZ ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +-*/=
terminal-dosis- bold	abcdefghijklmnopqrstuvwxyZ ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +-*/=
ptsans	abcdefghijklmnopqrstuvwxyZ ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +-*/=
ptsans-italic	<i>abcdefghijklmnopqrstuvwxyZ ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +-*/=</i>
ptsans-bold	abcdefghijklmnopqrstuvwxyZ ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +-*/=
ptsans-bold-ita lic	<i>abcdefghijklmnopqrstuvwxyZ ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890.,; ' " (!?) +-*/=</i>

The **DefaultFont** parameter takes only the font family. The style suffix is automatically applied by the phone. Hence, the following built-in font families can be used:

- ptsans
- aleo
- banksia
- droidsans
- libre-caslon
- opensans
- quattroceto
- terminal-dosis



- When using the Tomáš skin, the font is fixed. A configured **DefaultFont** value won't take effect.
- When using a language other than English, make sure to use a font family that includes all the characters in the chosen language.

Language Customization with Dictionary Files

Language customization allows you to change the language of the displayed text shown on the phone UI. The customization is done by installing a dictionary file for each selectable language. Dictionary files are XML files stored in the internal folder `/dict/` and must be UTF-8 encoded. Custom dictionary files may only be installed on the phone using the data package upload method. By default, the phone comes with three built-in dictionary files:

- English-US
- English-UK
- Spanish

You can set select the language under **Preferences** on the UI.

Below is the full Spanish dictionary file that is shipped with the phone. The name of the language is specified in the `lang` attribute of the root element `<dictionary>` and is also the name shown to you in the **Preferences** language selection.

```
<obiphone>
<!-- Each dictionary item has a key/value pair, a mode attribute and a namespace.
      When applying the dictionary, we match the key against the source phrase.
      If mode="i", the comparison is case-insensitive; otherwise the comparison
      is case-sensitive. The namespace ns must be matched also. -->

<dictionary lang="Español">
  <!-- App names -->
  <item key="Contacts" value="Contactos"/>
  <item key="Current Calls" value="Llamadas"/>
  <item key="Call History" value="Historial"/>
  <item key="Preferences" value="Preferencias"/>
  <item key="Settings" value="Ajustes"/>
  <item key="Product Info" value="Producto"/>
  <item key="Buddy List" value="Amistades"/>
  <item key="Net Services" value="Red Servicios"/>

  <!-- Line key function name -->
  <item key="AANS" value="Respuesta Automática" mode="i"/>
  <item key="acd" value="Distr. automático de llamadas" mode="i"/>
  <item key="BXFER" value="Transferencia Ciega" mode="i"/>
  <item key="BAC" value="Blq. Llamadas Anónimas" mode="i"/>
  <item key="BCI" value="Blq. ID de Llamadas" mode="i"/>
  <item key="BLF" value="Monitor de teléfono" mode="i"/>
  <item key="call" value="Llamada" mode="i"/>
  <item key="CFA" value="Desvío de Llamadas" mode="i"/>
  <item key="CWA" value="Llamada en Espera" mode="i"/>
  <item key="CONF" value="Unir a Conferencia" mode="i"/>
  <item key="disp-code" value="Código Disposición" mode="i"/>
  <item key="DND" value="No Molestar" mode="i"/>
  <item key="DNR" value="No Suena" mode="i"/>
  <item key="HOLD" value="En Espera" mode="i"/>
  <item key="hoteling" value="Hoteling" mode="i"/>
  <item key="acd" value="ACD Agent State" mode="i"/>
```



```
<item key="exe" value="Exec Call Filter" mode="i"/>
<item key="xass" value="Exec Assistant" mode="i"/>
<item key="sec" value="Security Class" mode="i"/>
<item key="LINE" value="Monitor Línea" mode="i"/>
<item key="lkgpg" value="Siguiente Tabla" mode="i"/>
<item key="MWI" value="Mensajes" mode="i"/>
<item key="pg1" value="Grupo de Páginas 1"/>
<item key="pg2" value="Grupo de Páginas 2"/>
<item key="SPD" value="Marcado Rápido" mode="i"/>
<item key="XFER" value="Transferencia" mode="i"/>
<item key="pres" value="Presencia" mode="i"/>
<item key="cpm" value="Monitor de Aparcamiento de Llamada" mode="i"/>

<!-- Product Info entries -->
<item key="Model" value="Modelo" mode="i"/>
<item key="OBi Number" value="Número OBi" mode="i"/>
<item key="MAC Address" value="Dirección MAC" mode="i"/>
<item key="WiFi MAC Address" value="Dirección MAC de WiFi" mode="i"/>
<item key="Serial Number" value="Número Serial" mode="i"/>
<item key="Software Version" value="Versión del Software" mode="i"/>
<item key="Hardware Version" value="Versión del Hardware" mode="i"/>
<item key="Customization Status" value="Estado de Personalización" mode="i"/>
<item key="Up Time" value="Tiempo de Actividad" mode="i"/>

<!-- User Preferences entries -->
<item key="Language" value="Lenguaje" mode="i"/>
<item key="Skin" value="Contraste" mode="i"/>
<item key="Background Picture" value="Imagen de Fondo" mode="i"/>
<item key="Default Ringtone" value="Timbre Predeterminado" mode="i"/>
<item key="Default Font" value="Fuente Predeterminada" mode="i"/>
<item key="Screen Saver" value="Protector de Pantalla" mode="i"/>
<item key="Screen Saver Delay" value="Protector de Pantalla de retardo" mode="i"/>
<item key="Screen Saver Type" value="Tipo de Protector de Pantalla" mode="i"/>
<item key="Screen Brightness" value="Brillo de la Pantalla" mode="i"/>
<item key="Preferred Audio Device" value="Dispositivo de Audio Preferido" mode="i"/>
<item key="Preferred Headset Device" value="Auriculares Preferido" mode="i"/>
<item key="Do Not Disturb" value="No Molestar" mode="i"/>
<item key="Do Not Ring" value="No Suena" mode="i"/>
<item key="Call Forward" value="Desvío de Llamadas" mode="i"/>
<item key="Call Waiting" value="Llamada en Espera" mode="i"/>
<item key="Block Anonymous Call" value="Bloq. Llamadas Anónimas" mode="i"/>
<item key="Anonymous Call" value="Llamadas Anónimas" mode="i"/>
<item key="Auto Answer Page" value="Página de Respuesta Automática" mode="i"/>
<item key="Join Page Group" value="Únete Grupo de Páginas" mode="i"/>
<item key="Ringer Volume" value="Volumen del Timbre" mode="i"/>
<item key="Speakerphone Volume" value="Volumen de Altavoz" mode="i"/>
<item key="Speakerphone Mic Gain" value="Ganancia Mic Altavoz" mode="i"/>
<item key="Handset Volume" value="Volumen de Auriculares" mode="i"/>
<item key="Handset Mic Gain" value="Ganancia Mic Auriculares" mode="i"/>
<item key="RJ9 Headset Volume" value="RJ9 Volumen de Auricular" mode="i"/>
```

```
<item key="RJ9 Headset Mic Gain" value="RJ9 Ganancia Mic Auricular" mode="i"/>
<item key="Equalizer" value="Equalizador" mode="i"/>
<item key="AEC" value="Cancelación de Eco Acústico" mode="i"/>
<item key="Home App Columns" value="Columnas de Aplicación Home" mode="i"/>
<!-- Screen Saver Type -->
<item key="Slide Show" value="Pantalla de Visualizacion" mode="i"/>
<item key="Turn Off Display" value="Desactivar la Visualización" mode="i"/>
<!-- Preferred Audio Device -->
<item key="Speaker" value="Altavoz" mode="i"/>
<item key="Handset" value="Auriculares" mode="i"/>
<item key="Headset" value="Auricular" mode="i"/>
<!-- Preferred Headset Device -->
<item key="RJ9 Headset" value="RJ9 Auricular" mode="i"/>

<!-- Call Forward -->
<item key="Destination Number" value="Número de Destino" mode="i"/>
<item key="Voice Service" value="Servicio de Voz" mode="i"/>

<!-- Phone Book -->
<item key="Phone Book" value="Agenda" mode="i"/>
<item key="Fields" value="Campos" mode="i"/>
<item key="Name" value="Nombre" mode="i"/>
<item key="First Name" value="Primer Nombre" mode="i"/>
<item key="Last Name" value="Apellidos" mode="i"/>
<item key="Number" value="Número" mode="i"/>
<item key="Mobile Number" value="Número Movil" mode="i"/>
<item key="Office Number" value="Número de Oficina" mode="i"/>
<item key="Home Number" value="Numero de Casa" mode="i"/>
<item key="Service" value="Servicio" mode="i"/>
<item key="Picture" value="Foto" mode="i"/>
<item key="Ringtone" value="Timbre" mode="i"/>
<item key="Email" value="Email" mode="i"/>
<item key="Company" value="Empresa" mode="i"/>
<item key="Group" value="Grupo" mode="i"/>
<item key="Groups" value="Groupos" mode="i"/>
<item key="Co-Workers" value="Compañeros de Trabajo" mode="i"/>
<item key="Customers" value="Clientes" mode="i"/>
<item key="Family" value="Familia" mode="i"/>
<item key="Friends" value="Amigos" mode="i"/>
<item key="No Groups Defined" value="Grupo No Definido" mode="i"/>
<item key="Speed Dial" value="Marcación Rápida" mode="i"/>
<item key="Phone Book Filter" value="Filtro de Agenda" mode="i"/>

<!-- Call History -->
<item key="Missed Calls" value="Llamadas Perdidas" mode="i"/>
<item key="Received Calls" value="Llamadas Recibidas" mode="i"/>
<item key="Outgoing Calls" value="Llamadas Realizadas" mode="i"/>
<item key="All Calls" value="Todas las Llamadas" mode="i"/>
<item key="List empty" value="Lista Vacía" mode="i"/>
```

```
<!-- Buddy List -->
<item key="My Status" value="Mi Estatus" mode="i"/>
<item key="My Presence" value="Mi Presencia" mode="i"/>
<!-- Presence status -->
<item key="unknown" value="Desconocido" mode="i"/>
<item key="offline" value="Fuera de Linea" mode="i"/>
<item key="online" value="En Linea" mode="i"/>
<item key="away" value="Fuera de Oficina" mode="i"/>
<item key="xa" value="Fuera de Oficina Extendida" mode="i"/>
<item key="dnd" value="No Molestar" mode="i"/>
<item key="Online (Available)" value="En Linea (Disponible)" mode="i"/>
<item key="Offline (Invisible)" value="Fuera de Linea (Invisible)" mode="i"/>
<item key="Extended Away" value="Fuera de Oficina Extendida" mode="i"/>
<item key="DND (Do Not Disturb)" value="DND (No Molestar)" mode="i"/>

<!-- Network Directory -->
<item key="Enterprise" value="Empresa" mode="i"/>
<item key="Group Common" value="Grupo en Comun" mode="i"/>
<item key="Enterprise Common" value="Empresa en Comun" mode="i"/>
<item key="Personal" value="Personal" mode="i"/>

<!-- Settings -->
<item key="Network" value="Reseteo de Fabrica" mode="i"/>
<item key="Product Information" value="Informacion Producto" mode="i"/>
<item key="WiFi" value="WiFi" mode="i"/>
<item key="Bluetooth" value="Bluetooth" mode="i"/>
<item key="Storage" value="Almacenamiento" mode="i"/>
<item key="User Preferences" value="Preferencias Usuario" mode="i"/>
<item key="Programmable Keys" value="Teclas Programables" mode="i"/>
<item key="Line Keys" value="Teclas de Linea" mode="i"/>
<item key="Voice Services" value="Servicios de Voz" mode="i"/>
<item key="Device Admin" value="Administracion Dispositivo" mode="i"/>
<item key="Lock Admin Settings" value="Bloqueo Config. del Admin." mode="i"/>
<item key="Unlock Admin Settings" value="Desbloquear Ajustes del Admin." mode="i"/>
<!-- Network -->
<item key="Network Settings" value="Ajustes de la Red" mode="i"/>
<item key="Addressing Type" value="Addressing Type" mode="i"/>
<item key="IP Address" value="Direccion IP" mode="i"/>
<item key="Subnet Mask" value="Mascara Subred" mode="i"/>
<item key="Default Gateway" value="Direccion Gateway" mode="i"/>
<item key="DNS Server" value="Servidor DNS" mode="i"/>
<item key="DNS Query Order" value="Orden de Consulta DNS" mode="i"/>
<item key="DNS Query Delay" value="Retrazo de Consulta DNS" mode="i"/>
<item key="AC Name" value="Nombre AC" mode="i"/>
  <item key="Service Name" value="Nombre de Servicio" mode="i"/>
  <item key="Password" value="Contraseña" mode="i"/>
  <item key="VLAN" value="VLAN" mode="i"/>
  <item key="LLDP-MED" value="LLDP-MED" mode="i"/>
  <item key="ID" value="ID" mode="i"/>
  <item key="Priority" value="Prioridad" mode="i"/>
```

```

<item key="NTP Server" value="Servidor NTP" mode="i"/>
<item key="Local Time Zone" value="Zona Horaria Local" mode="i"/>
<item key="Daylight Saving Time" value="Horario de Verano" mode="i"/>
<!-- WiFi -->
<item key="WiFi Setup" value="Configuracion WiFi" mode="i"/>
<item key="OBiWiFi Setup Mode" value="Modo de configuración OBiWiFi" mode="i"/>
<item key="Security" value="Seguridad" mode="i"/>
<item key="Signal Strength" value="Intensidad de Señal" mode="i"/>
<item key="MAC Address" value="Direccion MAC" mode="i"/>
<item key="0" value="" ns="wfs"/>
<item key="1" value="Débil" ns="wfs"/>
<item key="2" value="Pobre" ns="wfs"/>
<item key="3" value="Pasable" ns="wfs"/>
<item key="4" value="Bueno" ns="wfs"/>
<item key="5" value="Excelente" ns="wfs"/>
<!-- Voice Services -->
<!--
    Highly technical entries under Voice Services are not translated.
-->
<!-- Admin Settings -->
<item key="Web Server Port" value="Puerto del Servidor Web" mode="i"/>
<item key="Web Admin Password" value="Contraseña del Administrador" mode="i"/>
<item key="Web User Password" value="Contraseña del Usuario" mode="i"/>
<item key="Syslog Server" value="Servidor Syslog" mode="i"/>
<item key="ITSP Provisioning Method" value="Método Aprovisionamiento ITSP" mode="i"/>
<item key="ITSP Provisioning Interval" value="Aprovisionamiento Interno ITSP"
mode="i"/>
<item key="ITSP Provisioning Config URL" value="Aprovisionamiento ITSP Config URL"
mode="i"/>
<item key="Auto Firmware Update Method" value="Metodos de Actualizacion Auto Firmware"
mode="i"/>
<item key="Auto Firmware Update Interval" value="Actualizacion Interno Auto Firmware"
mode="i"/>
<item key="Auto Firmware Update URL" value="Actualizacion Auto Firmware URL"
mode="i"/>
<!-- Provisioning and FW Upgrade Method -->
<item key="System Start" value="Inicio del Sistema" mode="i"/>
<item key="Periodically" value="Periodicamente" mode="i"/>
<!-- Dial App -->
<item key="Enter Number" value="Número de Entrada" mode="i"/>
<item key="Transfer Target" value="Destino de la Transferencia" mode="i"/>
<item key="Conference Target" value="Objetivo Conferencia" mode="i"/>

<!-- Soft Keys -->
<item key="Edit" value="Editar" mode="i"/>
<item key="Reboot" value="Reiniciar" mode="i"/>
<item key="End" value="Final" mode="i"/>
<item key="Hold" value="Espera" mode="i"/>
<item key="Resume" value="Resumen" mode="i"/>
<item key="Add to Conf" value="Unir a Conferencia" mode="i"/>

```

```
<item key="Conference" value="Conferencia" mode="i"/>
<item key="Park" value="Aparcar" mode="i"/>
<item key="Blind Transfer" value="Transferencia Ciega" mode="i"/>
<item key="Dispose Code" value="Deseche Código" mode="i"/>
<item key="Escalate" value="Escalar" mode="i"/>
<item key="Trace" value="Trazar" mode="i"/>
<item key="Rec.Start" value="Iniciar Grabacion" mode="i"/>
<item key="Rec.Stop" value="Parar Grabacion" mode="i"/>
<item key="Rec.Pause" value="Pausar Grabacion" mode="i"/>
<item key="Rec.Resume" value="Resumir Grabacion" mode="i"/>
<item key="Private Hold" value="Espera Privada" mode="i"/>
<item key="Answer" value="Contestar" mode="i"/>
<item key="Reject" value="Rechazar" mode="i"/>
<item key="Redial" value="Rellamar" mode="i"/>
<item key="Missed" value="Perdida" mode="i"/>
<item key="Dial" value="Marcar" mode="i"/>
<item key="Lines" value="Lineas" mode="i"/>
<item key="Switch Mode" value="Cambio Modo" mode="i"/>
<item key="Ph.Book" value="Agenda" mode="i"/>
<item key="Switch Line" value="Cambio Linea" mode="i"/>
<item key="Refresh All" value="Reiniciar Todo" mode="i"/>
<item key="+Buddy" value="Agregar Amigo" mode="i"/>
<item key="Search" value="Busqueda" mode="i"/>
<item key="Refresh" value="Reiniciar" mode="i"/>
<item key="+" value="Agregar" mode="i"/>
<item key="Add" value="Agregar" mode="i"/>
<item key="Remove" value="Remover" mode="i"/>
<item key="Clear List" value="Limpiar Lista" mode="i"/>
<item key="Edit Dial" value="Aditar Marcador" mode="i"/>
<item key="Save" value="Guardar" mode="i"/>
<item key="MyPresence" value="Mi Presencia" mode="i"/>
<item key="MyStatus" value="Mi Estatus" mode="i"/>
<item key="Filter" value="Filtrar" mode="i"/>
<item key="Select" value="Seleccionar" mode="i"/>
<item key="New" value="Nuevo" mode="i"/>
<item key="Clear" value="Limpiar" mode="i"/>
<item key="Sort" value="Ordenar" mode="i"/>
<item key="New Entry" value="Nueva Entrada" mode="i"/>
<item key="Group Filter" value="Filtrar Grupo" mode="i"/>
<item key="Import" value="Importar" mode="i"/>
<item key="Export" value="Exportar" mode="i"/>
<item key="Sync" value="Sincronizar" mode="i"/>
<item key="New Group" value="Nuevo Grupo" mode="i"/>
<item key="Uncheck All" value="No chequear Nada" mode="i"/>
<item key="Check All" value="Chequear Todo" mode="i"/>
<item key="Factory Reset" value="Reseteo de Fabrica" mode="i"/>
<item key="FW Update" value=" Actualizar FW" mode="i"/>
<item key="Pick Up" value="Contestar" mode="i"/>
<item key="Barge In" value="Interrumpir" mode="i"/>
<item key="Monitor" value="Monitorear" mode="i"/>
```

```
<item key="Sign Off" value="Cerrar Sesión" mode="i"/>
<item key="Wrap Up" value="Envolver" mode="i"/>
<item key="Scan" value="Escanear" mode="i"/>
<item key="Transfer Now" value="Transferir Ahora" mode="i"/>
<item key="Conference Now" value="Conferencia Ahora" mode="i"/>
<item key="Play" value="Seguir" mode="i"/>
<item key="Stop" value="Parar" mode="i"/>
<item key="Connect" value="Conectar" mode="i"/>
<item key="Rescan" value="Re-escanear" mode="i"/>
<item key="Disconnect" value="Desconectar" mode="i"/>
<item key="Forget" value="Olvidar" mode="i"/>
<item key="Reconnect" value="Reconectar" mode="i"/>
<item key="Explore" value="Explorar" mode="i"/>
<item key="Eject" value="Ejecutar" mode="i"/>
<item key="Mount" value="Montura" mode="i"/>
<item key="Back" value="Regresar" mode="i"/>
<item key="Copy" value="Copiar" mode="i"/>
<item key="Cancel" value="Cancelar" mode="i"/>
<item key="Paste" value="Pasar" mode="i"/>
<item key="New Folder" value="Nuevo Folder" mode="i"/>
<item key="Close" value="Cerrar" mode="i"/>
<item key="OK" value="OK" mode="i"/>
<item key="Replace" value="Reemplazar" mode="i"/>
<item key="Add As New" value="Aragar Como Nuevo" mode="i"/>

<item key="&gt;" value="&gt;"/>
<item key="&lt;" value="&lt;"/>
<item key="Backspace" value="Retroceso" mode="i"/>
<item key="Devices" value="Dispositivos" mode="i"/>

<!-- Misc -->
<item key="Not Found" value="Extraviado" mode="i"/>
<item key="No Entries" value="No Entradas" mode="i"/>
<item key="Unassigned" value="No Asignado" mode="i"/>
<item key="Enable" value="Activar" mode="i"/>
<item key="enabled" value="Activado" mode="i"/>
<item key="disabled" value="Desactivado" mode="i"/>
<item key="Not Configured" value="No Configurado" mode="i"/>
<item key="default" value="Por Defecto" mode="i"/>
<item key="On" value="Encendido" mode="i"/>
<item key="Off" value="Apagado" mode="i"/>
<item key="Non-linear" value="No Linear" mode="i"/>
<item key="Status" value="Estatus" mode="i"/>
<item key="outgoing" value="A" ns="ct"/>
<item key="received" value="De" ns="ct"/>
<item key="up" value="Arriba" mode="i"/>
<item key="down" value="Abajo" mode="i"/>
<item key="nocfg" value="No Configurado" mode="i"/>
<item key="holding" value="Esperando" mode="i" ns="cs"/>
<item key="peerring" value="Timbrando" mode="i" ns="cs"/>
```

```

<item key="proceeding" value="Timbrando" mode="i" ns="cs"/>
<item key="peerring2" value="Timbrando (Espera)" mode="i" ns="cs"/>
<item key="peerring3" value="Timbrando (Conf)" mode="i" ns="cs"/>
<item key="dialtone" value="Tono de Marcación" mode="i" ns="cs"/>
<item key="dialing" value="Marcando" mode="i" ns="cs"/>
<item key="trying" value="Tratando" mode="i" ns="cs"/>
<item key="connected" value="Conectado" mode="i" ns="cs"/>
<item key="connected-HD" value="HD Conectado" mode="i" ns="cs"/>
<item key="ended" value="Llamada Finalizada" mode="i" ns="cs"/>
<item key="ring" value="Timbrando" mode="i" ns="cs"/>
<item key="OBiTALK" value="OBiTALK" mode="i"/>
<item key="Auto" value="" mode="i"/>
<item key="Auto: " value="" mode="i"/>
<item key="OBiTALK: " value="OBiTALK: " mode="i"/>
<item key="Not Available" value="No Disponible" mode="i"/>
<item key="AA1: " value="OBi Asistente: " mode="i"/>
<item key="AA2: " value="IVR: " mode="i"/>
<item key="seized" value="Incautado" ns="cs"/>
<item key="held_private" value="Held Privado" ns="cs"/>
<item key="held" value="Held" ns="cs"/>
<item key="parked" value="Llamar Estacionado" ns="cs"/>

<!-- Misc line key data -->
<item key="error" value="error" mode="i"/>
<item key="Service Not Ready" value="Servicio no está listo" mode="i"/>
<item key="Service Subs. Error" value="Error suscripción de servicio" mode="i"/>
<item key="BLF Subs. Error" value="BLF error suscripción" mode="i"/>
<item key="Hoteling Subs. Error" value="Hoteling error suscripción" mode="i"/>
<item key="Guest" value="Invitado" mode="i"/>
<item key="No Guest" value="Los huéspedes no" mode="i"/>

<!-- ACD State -->
<item key="available" value="Disponible" mode="i"/>
<item key="unavailable" value="No Disponible" mode="i"/>
<item key="wrappingup" value="Terminando" mode="i"/>
<item key="signedoff" value="Cerrar Sesión" mode="i"/>

<!-- Input App Titles -->
<item key="New Status" value="Nuevo Estatus" mode="i"/>
<item key="Edit Status" value="Editar Estatus" mode="i"/>
<item key="New Group Name" value="Nuevo Nombre de Grupo" mode="i"/>
<item key="Enter" value="Entrar" mode="i"/>
<item key="Blind Transfer Target" value="Objetivo Transferencia ciega" mode="i"/>
<item key="Supervisor Extension" value="Supervisor de Extensión" mode="i"/>
<item key="Disposition Code" value="Disposicion de Codigo" mode="i"/>
<item key="Forward All Number" value="Remitir Todo Número" mode="i"/>
<item key="Unavailable Reason Code" value="No disponible Código de Motivo" mode="i"/>
<item key="Enter Query" value="Entrar Consulta" mode="i"/>
<item key="Hoteling Passcode" value="Codigo de Contraseña Hoteling" mode="i"/>
<item key="Hoteling Guest Extension" value="Extension Huesped Hoteling" mode="i"/>

```

```

<item key="Enter Admin Password" value="Ingrese Contraseña del Administrador"
mode="i"/>
<item key="Enter Folder Name" value="Ingrese Nombre del Archivo" mode="i"/>
<item key="Exec Assist Divert Number" value="Exec Asist Numero de Desvio" mode="i"/>
<!-- Call related alerts -->
<item key="No Active Calls" value="No hay Llamadas Activas" mode="i"/>
<item key="Call Error" value="Error de Llamada" mode="i"/>
<item key="No Call Key available to make another call" value="No Tecla de llamada
Disponible para hacer otra llamada" mode="i"/>
<item key="The requested service is not available at the moment" value="En este
momento el Servicio requerido no esta disponible" mode="i"/>
<item key="Conference Error" value="Error de Conferencia" mode="i"/>
<item key="No conference port available to add participant" value="No hay puerto de
conferencia para agregar participante" mode="i"/>
<!-- OBiWiFi -->
<item key="OBiWiFi" value="OBiWiFi" mode="i"/>
<item key="Use USB Port 1 for OBiWiFi." value="Use el puerto USB 1 para OBiWiFi."
mode="i"/>
<item key="Connected with: " value="Conectado con: " mode="i"/>
<item key="Connecting with: " value="Conectar con: " mode="i"/>
<item key="Failed to authenticate with " value="Fallado para autenticar con "
mode="i"/>
<item key="Authentication Failed" value="Falla la Autenticación" mode="i"/>
<item key="Failed to associate with " value="Error al asociar con " mode="i"/>
<item key="Association Failed" value="Asociación no" mode="i"/>
<!-- Reboot -->
<item key="Reboot" value="Reiniciar" mode="i"/>
<item key="Press OK to reboot the phone." value="Presione OK para Reiniciar el
Telefono." mode="i"/>
<!-- Factory reset -->
<item key="Press OK to reset all settings to factory default. Phone will reboot
automatically."
value="Presione OK para Resetear Todos los Ajustes de Fabrica por defecto. El
Telefono se Reiniciara Automaticamente." mode="i"/>
<!-- Admin login -->
<item key="Unlock Settings Failed" value="Desbloquear Configuración Error" mode="i"/>
<item key="Invalid Password" value="Contraseña Invalida" mode="i"/>
<!-- OBiTalk -->
<item key="Add Device to OBiTalk" value="Agregue un dispositivo a OBiTalk" mode="i"/>
<item key="Code sent to OBiTalk..." value="Codigo enviado a OBiTalk..." mode="i"/>
<item key="Device not allowed to join!" value="Dispositivo no se le permitió unirse!"
mode="i"/>
<!-- Conference bridge -->
<item key="Conf. Bridge Error" value="Error del Puente de Conferencia" mode="i"/>
<item key="Call to Conference Bridge failed" value="Llamada al Puente Conferencia
fallado" mode="i"/>
<item key="Conference Bridge ended unexpectedly" value="Puente Conferencia terminó
inesperadamente" mode="i"/>
<item key="Last Call Total Charges" value="Gasto Total de la ultima llamada"
mode="i"/>

```



```

<!-- Network directory -->
<item key="Network Directory" value="Directorio Red" mode="i"/>
<item key="refreshed" value="Actualizado" mode="i"/>
<item key="Nothing refreshed" value="Nada Actualizado" mode="i"/>
<item key="Network Directory Search" value="Busqueda de Directorio Red" mode="i"/>
<item key="No items found for your query" value="No se han encontrado en su búsqueda
artículos" mode="i"/>
<!-- Buddy list -->
<item key="Buddy List Error" value="Error de Lista de Amigos" mode="i"/>
<item key="Failed to load buddy list" value="Fracaso en Bajar Lista de Amigos"
mode="i"/>
<!-- Call history -->
<item key="Clear List" value="Limpiar Lista" mode="i"/>
<item key="Press OK to remove all entries in the selected call history"
value="Presione OK para remover todas las entradas en el historial de llamadas
seleccionadas" mode="i"/>
<item key="Duplicate Entry" value="Entrada Duplicada" mode="i"/>
<item key="Select 'Replace' or 'Add As New' to proceed." value="Seleccione
'Reemplazar' o 'Agregar como Nuevo' para proceder." mode="i"/>
<!-- Phone book -->
<item key="Remove Entry" value="Remover Entradas" mode="i"/>
<item key="Press OK to permanently remove this entry from the Phone Book."
value="Presione OK para remover permanentemente esta entrada de la Agenda."
mode="i"/>
<item key="Discard Changes?" value="Descartar Cambios?" mode="i"/>
<item key="Press OK to proceed. Changes to the Phone Book will be discarded."
value="Presione OK Proceder. Cambios a la Agenda sera descartado." mode="i"/>
<!-- Misc -->
<item key="Progress" value="Progreso" mode="i"/>
<item key="More" value="Mas" mode="i"/>
<item key="Disconnected" value="Desconectado" mode="i"/>
<item key="Secured with" value="Asegurado con" mode="i"/>
<item key="Not in range" value="Fuera de rango" mode="i"/>
<item key="Peer to peer" value="De igual a igual" mode="i"/>
<item key="Ad-hoc" value="Ad-hoc" mode="i"/>
<item key="Remembered" value="Recordado" mode="i"/>
<item key="Not Discoverable" value="No detectable" mode="i"/>
<item key="Available" value="Disponible" mode="i"/>
<item key="Seconds remaining" value="Segundos Restantes" mode="i"/>
</dictionary>
</obiphone>

```

On boot up, your phone up scans the dictionary directory for all installed dictionary files and prepares a list of language names by extracting the lang attribute in each file. The list of languages is presented to you under **Preferences > Phone > Language** in the phone UI. On the phone configuration web page the value for the **Language** parameter (under **User Settings – User Preferences**) is just a string and must be set to one of the available language names.

Phone Book Pictures

These are pictures to associate with each phone book entry. They are stored in the internal `/people/` folder. Supported formats are JPEG, PNG, GIF, and BMP. The recommended pixel dimension is no larger than 100Wx100H.



The pictures may be shown on the phone screen as part of caller-ID when the peer's number matches an entry in the phone book.

Ring Tones

Ring tone is one of the data types included in the phone customization data package. They are stored in the internal folder `/ringtones/`. Ringtone filenames must be ended with the `.wav` extension.

Phone Customization Data Package

Customizable data must be wrapped into one single data package in order to upload onto the phone. The final uploadable data package is a tar file that may be optionally zipped. The size of the tar file must be no larger than 30 MB. Inside the tar file, the data should be organized with the same folder structure as the way they are stored in the phone internal storage:

```
/backgnd Wall paper (background pictures) and logo.raw (splash screen)
/dict Dictionary files
/fonts Text font files
/ringtones Ring tone files
/people Phone book contact pictures
```

As an example, create a directory `/home/test/obiphone-data` on a Linux machine that contains a folder structure as described above. Copy all the data files to be uploaded to the phone in the corresponding folders in this directory, and issue the following command lines to create the target tar file:

```
% cd /home/test/obiphone-data
% tar -zcvf ../obiphone-data.tar.gz *
```



The tar command must be issued inside of the working directory `obiphone-data`, as the phone can't strip the first level in the tar file.

In addition, you can use the following command line to generate the MD5 checksum of the target tar file after it is created:

```
% md5sum obiphone-data.tar.gz
```

The MD5 checksum is used as the version of the target customization data package and it must be configured on the phones also in order to trigger them to download a different version of the package if the currently installed package has a different MD5 checksum from the configured MD5 checksum. This is explained further in the following section.

Uploading Customization Data Package to the Phone

After the tar file of the customization data package is created, it must be placed on a server to be uploaded to the phones when the file is requested from the server by the phones. On the phone side, the downloading of the customization data package from a server is controlled by a number of configuration parameters as described in the table below, not unlike the way phone firmware is updated. As you can tell quickly that these parameters have the same meaning as those similarly named parameters for downloading of firmware.

Note that your phone attempts to download a customization data package from the server after firmware update and configuration parameter provisioning are completed. If the MD5 checksum of the downloaded data packages does not match the configured value, the package is dropped and no data update is performed. Otherwise, the phone install the downloaded data files into the corresponding folders and then restarts (warm reboot).

Customization Data Package Parameters

Parameter Group	Parameter	Description
Provisioning	Method	This parameter controls if and when the phone should execute given DownloadURL to download the customization data package (if the installed version is not current). Choice of: <ul style="list-style-type: none"> • <code>Disabled</code> = Do not execute DownloadURL • <code>System Start</code> = Execute DownloadURL just once on system start • <code>Periodically</code> = Execute DownloadURL on system start, and then periodically at the interval specified in the <code>Interval</code> parameter Note: First download on system start occurs after firmware update and configuration provisioning are complete.
	Interval	When Method is set to <code>Periodically</code> , this is the interval in number of seconds between execution of DownloadURL . If value is 0, the phone executes DownloadURL just once on system start (that is, equivalent to setting Method to <code>System Start</code>).
	DownloadURL	In its simplest form, this is the URL to download the Phone Customization Data Package, such as <code>http://abcd.com/phone/cfg/datapkg.tar</code> The full syntax is a script similar to what can be specified in the FirmwareURL parameter for firmware update. Refer to the Polycom OBi Edition Deployment Guide for a description of all the syntaxes of this script.
	MD5Checksum	Standard MD5 checksum of the Customization Data Package (a 32-character string of hexadecimal digits). This value must be provided to the phone and it serves as the version of the target data package. The phone executes DownloadURL only if the currently installed data package has a different MD5 checksum from the value of this parameter.

Customization Data Package Parameters

Parameter Group	Parameter	Description
	Incremental	This Boolean option, if enabled, causes the phone to install the customization data files without erasing the old data files first. Otherwise, the old data files are erased before saving the downloaded data files.
	DnsLookupType	Choice of: <ul style="list-style-type: none"> • A Record Only • SRV Record Only • Try Both
	DnsSrvPrefix	Choice of: <ul style="list-style-type: none"> • No Prefix • With Prefix • Try Both
	Username	(Optional) Username for authentication if DownloadURL uses the scheme <code>http://</code> or <code>https://</code>
	Password	(Optional) Password for authentication if DownloadURL uses the scheme <code>http://</code> or <code>https://</code>

Internal Data Storage Paths for User Preferences Settings

There are a number of configuration parameters to directly or indirectly select one of several internally stored data files for the phone to perform certain tasks, such as background picture or ringtone. This section describes how the built-in and customized data files that support these settings are organized internally so that the administrator can provision these parameter values accordingly.

The data files that are selectable by configuration are organized into three levels: OBi Built-in, ITSP (or Admin) Customized, and User Customized. Each level has its own dedicated storage areas internally. The following table summarizes the user settings that utilize these data files and where those data files are stored internally for each level:

Internal Data Storage Paths for User Preferences

Configuration Parameter	Description	Internal Data Storage Folders		
		Phone Built-in	ITSP Customized	User Customized
User Preferences:: Language	<p>The parameter value must match the lang attribute of the root element of one of the dictionary xml files found under the folders on the right. When there is a conflict, the user version has the highest priority, then the ITSP version, then your phone's version.</p> <p>On each bootup the folders are scanned to create a list of available languages from the lang attributes that users can select from the Preferences menu as their Language.</p>	/data/dict/ /	/scratch/it sp/dict/	/scratch/ph one/dict/
User Preferences:: BackgroundPicture	<p>The parameter value must be the full internal path name of a picture file stored in one of the folders on the right or a URL.</p> <p>If a URL is specified in the value, it must start with <code>http://</code> or <code>https://</code> and the phone downloads and caches the data internally until it is power cycled.</p> <p>All available pictures under these folders are also listed under the Preferences menu for users to select as their wall paper picture. The filenames must have one of the following extensions: PNG, JPG, JPEG, GIF, or BMP.</p>	/data/themes/ backgnd/ /	/scratch/it sp/backgnd/ /	/scratch/ph one/backgnd/ /
User Preferences:: DefaultRingtone	<p>The parameter value must be the full internal path name of a wave file stored in one of the folders on the right or a URL.</p> <p>If a URL is specified in the value, it must start with <code>http://</code> or <code>https://</code> and the phone downloads and caches the data internally until it is power cycled.</p> <p>All available wave files under these folders are listed under a) the Default Ringtone option of the Preferences menu for users to select as their default ringtone, and b) the Ringtone field of the built-in Phone Book application for users to choose a ringtone for individual contact in phone book.</p>	/data/ring tones/	/scratch/it sp/ringtone s/	/scratch/ph one/rington es/

Internal Data Storage Paths for User Preferences

Configuration Parameter	Description	Internal Data Storage Folders		
		Phone Built-in	ITSP Customized	User Customized
There is no configuration parameter that refers to data files in these folders.	All available pictures under these folders on the right are listed under the Picture menu of the built-in Phone Book application for users to select to assign to a contact. The data filenames must be one of the following extensions: PNG, JPG, JPEG, GIF, or BMP.	/obi/people/	/scratch/it sp/people/	/scratch/ph one/people/
User Preferences::DefaultFont	The parameter value must be one of the available font family names found in the font.map files in the folders on the right. When there is a conflict, the user version has the highest priority, then the ITSP customized version, then your phone's version.	/data/fonts/	/scratch/it sp/fonts/	/scratch/ph one/fonts/

If a parameter value is to specify the filename of an internally stored data file, the full internal path must be specified. For examples:

User Settings – User Preferences::BackgroundPicture = /scratch/phone/backgnd/CherryBlossom.png

User Settings – User Preferences::DefaultRingtone = /data/ringtones/Office A.wav

Capture Your Device's Current Screen

From the phone web page, you can take a snapshot of the current phone screen and save it as a .bmp file on your host computer.

To capture your device's screen:

- 1 Enable the **Device Admin – Web Server::LCDScreenShot** option.
- 2 Open the **System Management – Device Update** web page of the phone.
- 3 Press **Snap**.

System Logs

You can capture system logs to help troubleshoot your OBi Edition VVX business IP phones.

Syslog

Your phone can be set up to send out syslog messages for troubleshooting. To capture syslog messages from your phone, you need to run a syslog server application at an IPv4 address that is reachable from the phone. On the phone side, the IP address and listening port of the syslog server are configured in **Device Admin – Syslog::Server** and **Device Admin – Syslog::Port**. The default **port** value is 514.

To include detailed SIP messages on an SP service in the syslog, use the **SPn Service::X_SipDebugOption** parameter with one of the options: `Disable`, `Log All Messages`, or `Log All Except REGISTER Messages`. **SPn Service::X_SipDebugExclusion** is a list of SIP methods (request and responses) to exclude from the log. For troubleshooting a call flow, methods such as `OPTIONS` that are used for keep-alive purpose may be excluded from the log in most cases.

Hardware and Accessories

This section provides information on configuring power management options and available OBi Edition VVX phone accessories for your users.

Powering OBi Edition VVX Phones with an Ethernet Switch Connection

OBi Edition VVX phones have a 3-port Ethernet switch: one internally connected port to the phone processor for traffic to and from the phone and two Ethernet ports—labeled **SW** and **PC**.

You can power your phone through an AC power adapter or through a Power over Ethernet (PoE) cable connected to the phone's **SW** Ethernet switch port.



Use one power source or the other, but not both.

Available Accessories

The phones support the following accessories:

- 5 VDC Power Adapter: Powers the phone when Power over Ethernet (PoE) isn't available.
- Polycom® WiFi USB Adapter: Connects the phone to a Wi-Fi network. (VVX 250, VVX 350, and VVX 450 only)
- USB headsets (VVX 250, VVX 350, and VVX 450 only)

Audio Features

Your phone enables you to create customized tone patterns and ringtones.

Tone Patterns



Tone and Ring Profile A default settings are set for North American telephone standards. Tone and Ring Profile B default settings are set for Australian telephone standards. Tone profiles for other countries are available for download from the OBiTALK forum.

Tone Profile Features of Your Phone

The general format for tone profiles follows the following format:

```
[field-1];[field-2];[field-3];...;[field - 6]
```

Use a semicolon to separate the configuration fields.



Spaces are not allowed in a tone profile pattern.

Field–1 Composition

This field describes frequency components used for tone synthesis, and it supports as many as three different frequencies.

The frequency expression is a string of numeric values with the notation + or –. The numeric values are the frequency's decimal values in Hz and amplitude in dBm (maximum 3 dBm). Different frequencies are separated by a comma.

Example: 350–18,440–18,550+2

This example means:

- The first frequency at 350 Hz with strength at –18 dBm
- The second frequency at 440 Hz with strength at –18 dBm
- The third frequency at 550 Hz with strength at +2 dBm

Field–2 Composition

This field describes the overall tone playback duration in seconds.

The expression is a numeric value, and supports as many as 3 decimated digits. The numeric value can be negative, zero, positive, or skipped:

- Negative value: tone plays indefinitely
- Zero value: tone playback is skipped
- Positive value: Normal playback duration
- No value: tone plays indefinitely

Example: 30.234

This example means:

- Tone playback terminates after 30.234 seconds

Field–3 to Field–6 Composition

Fields 3 through 6 share the same definition. Each field describes one single cadence segment. Together, these four fields form a macro-segment, which repeats until tone playback expires.

The expression is a string of numeric values with the special notation '/', '(', ')', and ','.

Its format is:

`t(f_0/on_0+off_0,f_1/on_1+off_1,f_2/on_2+off_2,f_3/on_3+off_3)`

- `t`: The cadence segment duration in seconds
 - Negative value: tone plays indefinitely
 - No value: tone plays indefinitely
 - Zero value: the duration of this particular segment is zero
 - Positive value: Normal playback duration
- `f_0/1/2/3`: A digit that describes which frequency component(s) are used for the synthesis. It can be one of following options (0 through 7):
 - 0: No frequency specified (silence)
 - 1: The first frequency
 - 2: The second frequency
 - 3: The first and second frequencies
 - 4: The third frequency
 - 5: The first and third frequencies
 - 6: The second and third frequencies
 - 7: The first and second frequencies if two or more than two frequency components, or the first frequency if only one frequency component is available.

If no value is provided for `f_0/1/2/3`, it automatically uses the combination of the first one or two available frequency components.
- `on_0/1/2/3`: The tone active time in seconds
 - Negative value: Not allowed
 - No value: infinite tone active time
 - Others: normal tone active time (as many as 3 decimated digits)
- `off_0/1/2/3`: The tone inactive time in seconds

- Negative value: Not allowed
- No value: infinite tone inactive time
- Others: normal tone inactive time (as many as 3 decimated digits)

Example: 4 (1/.3+2.34,3/2+1.5)

This example means:

- Use the first frequency to generate tone for 0.3 seconds
- Follow this tone with 2.34 seconds of silence
- Use a combination of the first and second frequencies to generate tone for 2 seconds
- Follow this tone with 1.5 seconds silence
- The cadence operates repeatedly for 4 seconds

Tone Examples

These examples show the interpretation of a few common tone patterns:

Dial Tone

DIAL, "350-18,440-18"

Dial tone is generated as a mixture of two frequency components:

350 Hz at –18 dBm and 440 Hz at –18 dBm

The expiration time is infinite, and tone active time is infinite.

Busy Tone

BUSY, "480-18,620-18;10;(.5+.5)"

Busy tone is generated as a mixture of two frequency components:

480 Hz at –18 dBm and 620 Hz at –18 dBm

The expiration time is exactly 10 seconds. It has only one cadence segment, which has tone active 0.5 second and tone inactive 0.5 second.

Prompt Tone

PROMPT, "480-16;10"

Prompt tone is generated from a single frequency component:

480 Hz at –16 dBm. The expiration time is exactly 10 seconds. It has only one cadence segment, which has tone infinite active time.

SIT Tone

SIT_1, "985-16,1428-16,1777-16;20;(1/.380+0,2/.380+0,4/.380+0,0/0+4)"

Special information tone (SIT) is generated from a set of frequency components:

- First frequency: 985 Hz at –16 dBm
- Second frequency: 1428 Hz at –16 dBm
- 3rd frequency: 1777 Hz at –16 dBm

The expiration time is exactly 20 seconds. It has only one cadence segment, which includes 4 on-off sections. The segment has infinite repeating time:

- The first on-off section: generated by the first frequency component, and it has 0.38 tone second active time and 0 inactive time.
- The second on-off section: generated by the second frequency component, and it has 0.38 tone second active time and 0 inactive time.
- The third on-off section: generated by the third frequency component, and it has 0.38 tone second active time and 0 inactive time.
- The fourth on-off section: only generate silence since no frequency component is specified. It has tone 0 second active time and 4 seconds inactive time.

Stutter Tone

```
STUTTER, "350-18,440-18;10;.6(.1+.1);(/)"
```

Stutter dial tone is generated from a mixture of two frequency components:

350 Hz at -18 dBm and 440 Hz at -18 dBm. The expiration time is exactly 10 seconds. It has two cadence segments.

- The first segment includes only one on-off sections, on 0.1 second and off 0.1 second, and on-off repeats for 0.6 second.
- The second segment includes one on-off section, and has infinite repeating time and infinite tone active time.

Ringtones and Ring Patterns

The general format of an OBi Ring Profile is as follows: [field-1];[field-2];...;[field - 5]

Use a semicolon to separate as many as five (5) configuration fields.



No spaces are allowed to be used in a tone profile pattern.

Field-1 Composition

Field-1 describes the overall ringing duration in seconds.

The expression is a numeric value, and supports as many as 3 decimated digits.

The numeric value can negative, zero, and positive:

- Negative value: Ringing lasts indefinitely
- No value: Ringing lasts infinitely
- Zero value: Ringing is skipped
- Positive value: Normal ringing duration

Example: 30.5

This example illustrates a ringing tone that terminates after 30.5 seconds.

Field-2 to Field-5 Composition

Fields 2 through 5 share the same definition, and each field describes one single cadence segment. Together, the four fields form a macro-segment, which repeats until ringing expires.

The expression is a string of numeric values with the special notation '(' , ')' and ','

It has the format as per the following construct:

`t(on_0+off_0,on_1+off_1,on_2+off_2,on_3+off_3)`

- `t`: The cadence segment duration in seconds
 - Negative value: Ringing indefinitely
 - No value: Ringing indefinitely
 - Zero value: Ringing is skipped
 - Positive value: Normal ringing duration
- `on_0/1/2/3`: The ring active time in seconds
 - Negative value: Not allowed
 - No value: Infinite ring active time
 - Others: Normal ring active time (as many as 3 decimated digits)
- `off_0/1/2/3`: The ring inactive time in seconds
 - Negative value: Not allowed
 - No value: Infinite ring inactive time
 - Others: Normal ring inactive time (as many as 3 decimated digits)

Example: `4(.3+2.34,2+1.5)`

This example illustrates a ringing tone comprised of two segments. Ringing is active for 0.3 seconds, followed by 2.34 seconds of silence, then ringing for 2 seconds, and followed by 1.5 seconds of silence.

This cadence operates repeatedly for 4 seconds.

Voice Services

A Voice Service, also known as a Line or Trunk, is an individual user account with an ITSP. You can configure the following voice services on a phone IP port:

- SP1
- SP2
- SP3
- SP4
- SP5
- SP6
- OBiTALK

An SP n service can be a generic SIP voice service. Examples of SIP/SP service include:

- An extension from a PBX
- A subscriber account with a service provider

Every SP service user account requires a user name, and oftentimes, a password for authentication. The service provider assigns an extension number or DID number to the user account. The assigned number may be the same as the user name of the account.

OBiTALK is a built-in service provided and managed by Polycom customers who can use the service for technical support and device-to-device calling among OBi devices.

ITSP Profiles

The configuration of an SP service is divided into two parts: A Service Provider part and a Service Subscriber Part. The Service Provider part comprises parameters that are common to all service subscriber accounts from that service provider. The Service Subscriber part comprises of parameters that may vary for each specific subscriber account from the service provider.

On your phone, each Service Provider part is known as an ITSP Profile that has its own parameter groups. You can configure as many as six ITSP Profiles (A through F) in a phone configuration. The Service Subscriber, known as an SP Service, includes the **SP n ::X_ServProvProfile** parameter that binds the SP service with an ITSP Profile. By default, the **SP n ::X_ServProvProfile** parameter for all SP services points to ITSP Profile A. If you want to use two different service providers with your phone, configure the settings for them in ITSP Profile A and ITSP Profile B respectively.

A common mistake is not setting the **X_ServProvProfile** parameter correctly to point to the corresponding ITSP Profile as intended.

Audio Codec Profiles

Two codec profiles are available on your phone, and they're selectable per trunk (OBiTALK, SP n , $n = 1$ through 6). To select a codec as the preferred codec in this profile, set the priority of that codec to be highest among all the enabled codecs in the profile. You can assign each of the SP and OBiTALK services to a codec profile in its corresponding configuration. The codec list to use when setting up a call on the underlying service is formed from the list of enabled codecs in the chosen profile and ordered according to the assigned priorities in the profile. For codecs with the same priority setting, the codec that appears first on the codec profile web page has a higher priority.

Specification of Target Phone Numbers

There are places within the configuration that specify a target phone number, such as a speed dial number or a call forward number.

Here, two formats specify a target phone number: A Short Number where the number itself is specified, such as 3231234, and a Full Number where the number and the service to use the number are specified, such as sp3 (14089993312).

The case-insensitive service name to use for each service in the full number specifications is:

- `spn` for SP n Service for $n = 1 - 6$
- `pp` for OBiTALK Service

Overview of Common Trunk Configuration

Trunks of every kind share some common characteristics that you can configure.

Trunk Capacity

Trunk Capacity is the maximum number of simultaneous calls allowed on the trunk. You can set the value for each service using the **MaxSessions** parameter. The default value is 2 for all services. For OBiTALK service, the maximum value is 4. For other SIP/SP services, do not set the value higher than the maximum number of simultaneous calls allowed by the service provider.

Service Enable

Before you put a trunk into service, enable the trunk in the phone configuration. There's an **Enable** parameter for each service, and this parameter is checked by default.

Service Account Credentials

All SP services require credentials; however, OBiTALK services do not require credentials. At the minimum, you must configure a user name for the SP service account using these parameters:

- **SP n Service- Service Credentials::AuthUserName**

If a password is also required for authentication to the server, put it into:

- **SPn Service- Service Credentials::AuthPassword**

In a less common situation where the user name used for SIP authentication is different from the account user name, set the account user name in **SPn Service- Service Credentials::URI** and the different user name for authentication only in **AuthUserName** (required). If **URI** isn't specified, it's used for both as the account user name and for SIP authentication. Note also that if **AuthUserName** isn't specified, your phone considers the service as disabled.

Servers

OBi Edition phones support the following servers and voice services:

- SIP Servers
- OBiTALK Servers

Another commonly used term for the server equipment is *softswitch*. A softswitch typically offers a lot of extra business productive/collaboration features in addition to basic phone services. In fact, there are a few popular open-source and commercial softswitch implementations such as BroadSoft, MetaSwitch, and FreePBX that the phones fully support.

No matter what technology the service provider side equipment uses, you must provision it into your phone configuration as a domain name or an IP address along with a port number if the server isn't listening at the standard port (5060). Note that since OBiTALK servers are known by your phone, there's no need to configure the server domains for this service. For other SIP/SP services, configure the proxy server in the **ITSP Profile X – SIP::ProxyServer** parameter. If the listening port is nonstandard, configure the correct value in **ITSP Profile X – SIP::ProxyServerPort**. The **OutboundProxy** parameter in the same parameter group is often needed when the device-facing server is a session border controller (SBC). Similarly if the outbound proxy isn't listening at the standard port, configure the correct port value in **OutboundProxyPort**. The **RegistrarServer** and **RegistrarServerPort** parameters are rarely needed; your phone assumes that the SIP Registrar is the same as the SIP Proxy Server if they aren't specified separately.

SIP Transport refers to the transport protocol to use to exchange SIP messages with the server: UDP, TCP, and TLS, which are all supported by your phone. Use the **ITSP Profile X – SIP::ProxyServerTransport** parameter to configure the transport protocol. For TCP/TLS, your phone must start a TCP/TLS connection with the **ProxyServer** and use the same connection to exchange all subsequent SIP messages. If **OutboundProxy** is specified, your phone starts the TCP/TLS connection with the **OutboundProxy** instead. With the **OutboundProxyTransport** parameter, it's possible to choose a different transport to use with the **OutboundProxy** and with the **ProxyServer**.

When you only specify a short number, your phone determines the service to use, where necessary, by going through normal digit map and call routing processing. With a specified full number, your phone uses the number and service as specified without any modification.

SIP/SP Service

You can configure as many as six SIP/SP service accounts on your phones. For the purpose of this and other documents and web pages, Internet Telephony Service Provider (ITSP) is used generically to describe the logical entity providing the SIP Trunk service to your phone. When your phone is used with an IP PBX, ITSP refers to the IP PBX.

Each ITSP configuration is grouped as an ITSP Profile. There are six ITSP profiles available, referred to as ITSP Profile A, B, C, D, E, and F, respectively. SP service account specifics are grouped under the heading

SP_n Service, where $n = 1 - 6$. An ITSP Profile includes the **ProxyServer**, **OutboundProxy**, and **DigitMap** parameters, but, it doesn't include account-specific parameters. An SP Service includes account-specific parameters such as **AuthUserName** (usually, but not necessarily, the same as the phone number of the account), **AuthPassword**, **CallerIDName**, and **X_ServProfile** (which ITSP Profile to apply the ITSP-specific parameters from). If both SP Services use the same ITSP, it's usually possible to configure just one ITSP Profile with both SP Services referring to the same profile. However, if abstraction of an ITSP Profile isn't sufficient to cover a particular ITSP, configure multiple ITSP profiles for the same ITSP and have each individual SP service using that ITSP to point to a different ITSP profile.

The SP_n Service using ITSP Profile X is enabled by your phone only when at least the following parameters are set:

- **ITSP Profile X – SIP::ProxyServer** = Not Blank
- **SP_n Service::Enabled** = `true` (or checked on the native phone web page)
- **SP_n Service::AuthUsername** = Not Blank

where $X = A - F$ and $n = 1 - 6$. Otherwise, your phone considers the service disabled.

OBiTALK Service

OBiTALK is a proprietary protocol for communications among OBi Edition phones and to OBiTALK device management servers. The protocol is intended for two main purposes:

- Peer-to-peer calling between OBi devices
- Device management by OBiTALK servers

Every OBi Edition phone comes with one instance of the OBiTALK service with the (fixed) factory-assigned 9-digit device OBi number as the userid of the service. OBi devices can call each other by dialing the other party's OBi number.

The OBiTALK service is enabled with the **OBiTALK Service::Enable** parameter and is enabled by default (unless it is disabled through ZT Customization).

Through OBiTALK, you can call another OBi device out-of-the-box without any configuration. However, your phone must be able to reach out to the Internet to make an OBiTALK call. To call another OBi number, the dial ****9** followed by the 9-digit OBi number. For your convenience, Polycom maintains an Echo Server at the reserved OBi number 222 222 222. You can call this number to do a quick Echo Test after listening to a short announcement at the beginning. If the test is successful, you'll hear your voice echoed back as you talk during the test. This serves as a rudimentary check that the network and equipment is set up and functioning correctly.

An administrator such as an ITSP may desire to limit OBiTALK calls just to the Obihai Echo Server. To do this, the administrator can change the value of **OBiTALK Service::DigitMap** to: (`<ob>222222222|ob222222222`). Obviously, you can change or add more OBi numbers to this digit map by following the same pattern. A simple way to disable OBiTALK voice calls completely is by setting **OBiTALK Service – Calling Features::MaxSessions** to 0. You won't be able to do Echo Test in that case.

The OBiTALK service also makes it possible to view and change the settings of your phones from the OBiTALK portal. If the OBiTALK service is disabled in your phone's configuration, both OBiTALK voice calls and device management features won't be available.

Directories and Contacts

Your phone supports directory search function with an external server using the Lightweight Directory Access Protocol (LDAP). To use this function, an LDAP service must be configured on the phone. Users then invoke the LDAP directory search application by selecting the option from the phone main menu or by pressing a softkey.

LDAP Service Setup

The **Network Directory** option on the main menu of the phone can be pointed to an LDAP service. The parameters to set up the LDAP service are shown below.

LDAP Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LDAP – Server</i>	Host	The hostname can be an IP address or domain name, with optional <code>ldap://</code> or <code>ldaps://</code> scheme prefix. For example: <code>192.168.15.186</code> , <code>ldap.forums.com</code> , <code>ldap://ldap.testathon.net</code> are all acceptable hostname formats. If scheme is not specified, <code>ldap://</code> is implied.
<i>IP Phone – LDAP – Server</i>	Port	LDAP Server listening (TCP) port. The standard port is 389 for <code>ldap://</code> and 636 for <code>ldaps://</code> . If the port value is 0 or blank, the phone uses the corresponding standard port.
<i>IP Phone – LDAP – Server</i>	Password	The password for authentication with the LDAP server, based on the given distinguished name specified in BindDN . Note that this parameter is not used for anonymous queries.

LDAP Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LDAP – Search Parameters</i>	BindDN	<p>A Distinguished Name (DN) that is authorized to use the LDAP service. If none is specified, the query is regarded as an anonymous one which may or may not be acceptable to the server.</p> <p>The BindDN value is usually derived from a user name that typically looks like an email address, such as <code>admin@ldap.example.com</code>. In which case, the corresponding BindDN would be: <code>CN=admin,OU=users,DC=example,DC=com</code></p> <p>Note that in the last example, the DN includes only the last two parts of the domain name for illustration purpose only. The DN may as well include the <code>DC=ldap</code> field, for example. But the key point is that this value must agree exactly with how it is specified at the server side.</p>
<i>IP Phone – LDAP – Search Parameters</i>	SearchBase	<p>This parameter specifies the starting point of the LDAP search. It is a case-insensitive comma-separated list of <code>{object}={value}</code> pairs, where <code>{object}</code> can be any of the following:</p> <ul style="list-style-type: none"> • CN (Common Name) • OU (Organization Unit) • O (Organization) • C (Country) • DC (Domain) <p>If the value is not specified, the phone by default derives the search starting point from the value of LDAP – Server:: Host. For example, if Host is <code>ldap.example.com</code>, the default SearchBase value is <code>DC=example,DC=com</code></p> <p>Note that it is a common convention to use just the last two parts of the service domain as a search base, but it is not necessarily so. The phone assumes this convention when SearchBase is not specified.</p>
<i>IP Phone – LDAP – Search Parameters</i>	ProtocolVersion	<p>Protocol version. Either 3 or 2. 3 is the default.</p>
<i>IP Phone – LDAP – Search Parameters</i>	TLS_ReqCert	<p>Controls whether to verify the server's certificate on a TLS connection. Choice of:</p> <ul style="list-style-type: none"> • never • demand <p>Default value is <code>never</code>, which means not to verify the server's certificate.</p>

LDAP Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LDAP – Search Parameters</i>	ResultsPerPage	<p>Specifies how many results to display on screen per page. Valid values are:</p> <ul style="list-style-type: none"> • 20 • 40 • 60 • 80 • 100 <p>Default value is 20.</p>
<i>IP Phone – LDAP – Search Parameters</i>	DefaultSearchFilter	<p>Default search filter to append to each search. This must be specified as a complete and valid LDAP search filter. For example:</p> <pre>((objectclass=contact)(objectclass=person))</pre> <p>Default value is: (objectclass=*)</p>
<i>IP Phone – LDAP – Search Parameters</i>	QueryFields	<p>A comma-separated list of user input LDAP attributes/Display-Name to form the query filter. Each item has three attributes separated by two slashes (/): {ldap-attr}/{display-name}/{type} where</p> <ul style="list-style-type: none"> • {ldap-attr} is the standard ldap-attribute-name, such as sn, givenName, telephoneNumber, cn, This is the only required attribute in each field. • {caption} is optional. This is the caption to display on the screen for the input box. If not specified, the {ldap-attr} value is used in its place. • {type} is either A or N, for alphanumeric or number type respectively and is case insensitive. If not specified, it is assumed to be A. <p>For example: givenName/First Name, TelephoneNumber, Mobile</p> <p>Default value is: givenName/First Name, sn/Last Name, telephoneNumber/Tel, mobile/Mobile, homePhone/Home</p>

LDAP Parameters

Parameter Group	Parameter	Description
<i>IP Phone – LDAP – Search Parameters</i>	ResultFields	<p>Comma-separated list of LDAP fields to display for each entry of the search result. Each field has three attributes separated by two slashes (/): <code>{ldap-attr}/{caption}/{type}</code> where</p> <ul style="list-style-type: none"> <code>{ldap-attr}</code> is the standard ldap-attribute-name, such as <code>sn</code>, <code>givenName</code>, <code>telephoneNumber</code>, <code>cn</code>, ... This is the only required attribute in each field. <code>{caption}</code> is optional. This is the caption to display on the screen for the displayed value. If not specified, the <code>{ldap-attr}</code> value is used in its place. <code>{type}</code> is either <code>S</code>, <code>C</code>, or <code>N</code>, for String, Callable Number, or Picture, respectively. It is case insensitive. If not specified, it is assumed to be <code>S</code>. <p>Default value is: <code>cn/Common Name,sn/Last Name,givenName/First Name,telephoneNumber/Tel/y,mobile/Mobile/y,homePhone/Home/y</code></p>
<i>IP Phone – LDAP – Search Parameters</i>	NameFieldPreference	<p>Comma-separated list of LDAP attributes to be used as the Caller ID Name to display on screen, ordered by preference. The first non-empty value in the list is used. Default value is <code>cn,givenName sn</code>.</p>
<i>IP Phone – LDAP – Search Parameters</i>	NumberFieldPreference	<p>Comma-separated list of LDAP attributes to be used as the Number to display on screen and to call by default, ordered by preference. The first non-empty value in the list is used. Default value is <code>telephoneNumber,mobile,homePhone</code>.</p>
<i>IP Phone – LDAP – Search Parameters</i>	PhotoFieldPreference	<p>Comma-separated list of LDAP attributes to be used as the photo to display on screen, ordered by preference. The first non-empty value is used. Default value is <code>thumbnailPhoto</code>.</p>
<i>IP Phone – LDAP – Search Parameters</i>	SortByAttribute	<p>The LDAP attribute to use for sorting the search results by the server. Default value is <code>cn</code></p>
SASL Parameters (not applicable for LDAP version 2)		
<i>IP Phone – LDAP – SASL Authentication Parameters</i>	SASL_AuthMethod	<p>Method to use for SASL authentication. If the value is Disabled, the phone uses Simple authentication only. Choice of:</p> <ul style="list-style-type: none"> Disabled (the default) Plain MD5
<i>IP Phone – LDAP – Search Parameters</i>	SASL_AuthCID	<p>The authentication ID for SASL authentication. The format of this ID depends on the actual SASL mechanism used.</p>

Client Authentication

LDAP v2 supports `ldap://` and `ldaps://` with Simple Authentication only. LDAP v3 adds support for TLS and SASL Authentication. Simple authentication involves sending the LDAP server the fully qualified DN of the client and the corresponding password in clear-text, which has obvious security issue unless `ldaps://` or TLS is used.

SASL (stands for Simple Authentication and Security Layer) [RFC2222] is a framework for authentication. To use SASL the **LDAP – SASLAuthMethod** parameter must be set to either `Plain` or `MD5`.

For more informations on each of these SASL mechanisms, please check for example <http://www.openldap.org>

LDAP Directory Search Application

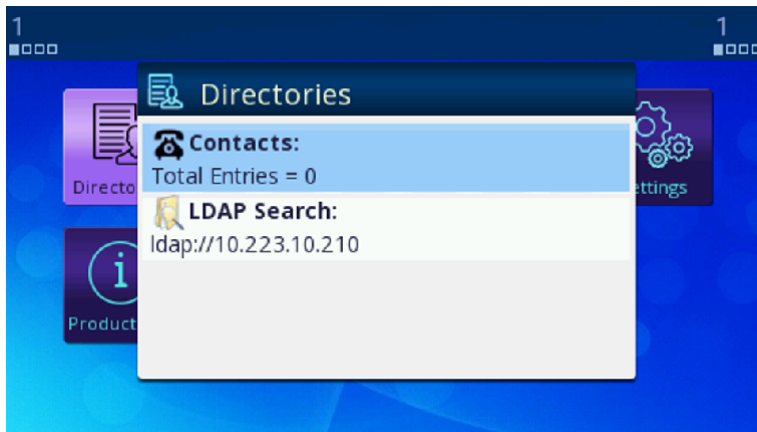
The LDAP application on the phone may be invoked either from the Main menu or with a softkey (with ID = `ldap`).

Two Main menu options can be used to invoke LDAP: **Directories** and **Network Directory**.

Invoke LDAP by Main Menu — Directories Option

Include the Main menu Item ID `directories` in the **Phone Settings – GUI Menus::MainMenu1** parameter. When the LDAP service is properly configured, the LDAP option appears under the **Directories** menu, as shown:

LDAP display example



Invoke LDAP by Main Menu – Network Directory Option

First you must include the Main menu item ID `netdir` in the **Phone Settings – GUI Menus::MainMenu1** parameter. To make the **Network Directory** option on the Main menu invoke the LDAP service, use the following settings:

IP Phone Settings – Network Directory::Enable = true (checked)

IP Phone Settings – Network Directory::VoiceService = LDAP

Invoke LDP by Softkey — LDAP

To use a softkey to invoke the LDAP application, add the `ldap` softkey to any of the configurable softkey sets. For example, configure the following value for the **Home** softkey set with the LDAP softkey in the fourth position:

```
redial,cfwd,dnd,ldap
```

Search Fields

By default, your phone presents the following search fields to you:

- Last Name (sn)
- First Name (givenName)
- Tel Num (telephoneNumber)
- Mobile Num (mobile)
- Home Num (homePhone)

One or more of these search fields may be specified per query; all the search criteria are added together to form the search filter, and the value is case-insensitive. To specify partial match of an attribute value, use a * wildcard character. For example: set Last Name = s* to query all entries with the sn that starts with an s or S.

You can customize the search fields presented to you by specifying the fields in the **QueryFields** parameter. This is a comma-separated list of fields. Refer to the LDAP parameter table above for the syntax.

Result Fields

The search results are presented in a table where a Name, a Number, and a Picture are shown, when the values are available. Which LDAP attributes in the result are to be used for the Name, Number, and Picture display can be controlled by setting the **NameFieldPreference**, **NumberFieldPreference**, and **PhotoFieldPreference** parameters. Each of these parameters is a comma-separated list of LDAP attributes arranged in order of preference. The first attribute in the list that has a non-empty value is used for the corresponding display field.

You can see more attribute values of an entry by highlighting the entry on the top-level search result table and pressing the **OK** key on the phone. This brings up the LDAP Result Details screen of the selected entry. You can customize the attributes to display on this screen with the **ResultsFields** parameter, which is a comma-separated list of fields to present to you. Refer to the LDAP parameter table for the syntax of this parameter.

Search result examples



Sorting of Results

The phone relies on the server to sort the results. Sorting is based on a single LDAP attribute which can be customized with the **SortByAttribute** parameter. The default value is `cn`.

Replace the Built-In Phone Book with LDAP

You can replace the built-in phone book with LDAP, such that when you press the **phbk** soft key or select **Contacts** from the Main menu, it launches the LDAP feature instead of the built-in (local) phone book. Below is the configuration for this:

User Preferences – Phone Book Settings::ActionURL = `phone://netdir`

Phone Settings – Network Directory::VoiceService = `LDAP`

Call Controls

Phone Level and Line Level Feature







You can configure call features as a phone-level feature or a line-level feature. A phone feature applies to all calls on the phone regardless which line a call is on. Call Waiting, for example, is a phone feature. A line feature applies only to calls on the specific line. For example, BLF is a line feature.

Some features may have a version for use as a phone feature as well as a line feature available on individual lines. For example, you can provision Do Not Disturb as a phone feature for all calls and as a line feature for each line for the SP services (SP1–SP6) and OBiTALK services.

Call States

As a call progresses from beginning to end, it goes through a number of defined stages commonly known as Call States or States. The following table includes the call states for calls on your phone.

List of Call States

Icon	Call State	Description	Available Operations	LED Pattern
	Dial Tone	Dial tone plays to prompt user to enter the target number to call.	End	Steady green
	Dialing	User is entering a target number to call.	End	Steady green
	Trying	Trying to call the dialed number but the called party hasn't started ringing yet.	End	Steady green
	Peer Ringing	The called party is ringing.	End	Fast blinking green
	Ringing	An incoming call is ringing the phone.	Reject, Answer	Fast blinking red
	Connected	Connected on a call and both sides are talking.	End, Hold, Transfer, Blind Transfer	Steady green

List of Call States

Icon	Call State	Description	Available Operations	LED Pattern
	Holding	User has placed the call on hold.	End, Resume, Add to Conf., Transfer, Blind Transfer	Slow blinking red
	Ended	Call failed due to various reasons, such as invalid number, service not available, called party busy, and so forth.	End (Remove the Call)	Slow blinking green
	Idle	No Call.	—	Off
	Registration Error	Error registering with the server.	—	Off
	Remote Hold	Remote side placed the call on hold.	End, Hold, Transfer, Blind Transfer	Steady green
	Messages Waiting	Idle state with unread messages in the mailbox.	—	Off
	Connected-HD	Call connected with HD Audio.	End, Hold, Transfer, Blind Transfer	Steady green
	Connected-HDS	Call connected with Secure HD Audio.	End, Hold, Transfer, Blind Transfer	Steady green
	Connected-S	Call connected with Secure Audio.	End, Hold, Transfer, Blind Transfer	Steady green
	Service Down	Service down.	—	Off

There are many operations that apply to a call during its course. For example, holding, resuming, or ending a call are commonly used operations. The call options available to users display on the phone as softkey options, while frequently used call operations have a feature key (such as Hold and Transfer). Softkey options are call-state sensitive and only the options that are applicable to the call at its current state display. As the call transitions from one state to another, the softkey options on the screen update accordingly.

Each call on the phone is assigned to a call key and each call key has its VLKW on the UI. To help users identify the current state of a call, an icon displays for each call state in the respective VLKW. Furthermore, the LED of each call key stays steady or blinks with a certain pattern depending on the current call state.

Core Call Features

Line Capacity

Line capacity refers to the maximum number of simultaneous calls that can be active per line. Some of the calls may be in the Holding state. The configuration of each voice service has a **MaxSessions** parameter that controls how many simultaneous calls to allow on that service. The default value is 2 for all lines. Set the number equal to or less than what the underlying service provider can support.

Complex Operations Between Multiple, Diverse Voice Services

Your phone supports call transfer, conference, and call forward operations involving calls on multiple, diverse services. This powerful feature makes the phone user-friendly by consolidating disparate services using different underlying technologies onto the same OBi phone. For example, you can call forward a caller on BroadSoft to a caller on OBiTALK.

Making Outgoing Calls

Digit Map

A digit map is a succinct way of describing a set of number patterns. The **Phone Settings::DigitMap** parameter determines the set of number patterns that users can dial. You can refer to named digit maps with the `(Mname)` syntax. For example, the default **Phone Settings::DigitMap** parameter refers to digit maps defined for SP1, SP2, SP3, SP4, and OBiTALK services with the reserved names `(Msp1)`, `(Msp2)`, `(Msp3)`, `(Msp4)`, and `(Mpp)` respectively:

```
(([1-9]x?* (Mpli) | [1-9]S9 | [1-9] [0-9]S9 | *** | **0 | **1(Msp1)|**2(Msp2)|**3(Msp3)|**4(Msp4)|*
*9(Mpp)|(Mpli))
```

This way, the digit map is more readable and organized.



The `(Mpli)` refers to the digit map of the Primary Line.

Audio Path and On-Hook or Off-Hook States

There are three audio paths for calls on the phone: handset, speakerphone, and headset. The headset audio path supports these devices: RJ9 Headset. Users can only switch on one audio path at a time and have only one headset device active when the headset audio path on. The phone is considered On-Hook if all of the audio paths for calls are off, otherwise, the phone is Off-Hook.

When an audio path is on, the other currently active audio path automatically switches off. The speakerphone and headset audio paths are each enabled by pressing the speakerphone key or the headset key. Lifting the handset from the cradle enables the handset audio path.

The following actions activate audio paths when the phone is on-hook or off-hook:

- Lifting the handset from the cradle
- Pressing the speakerphone button
- Pressing the headset button

Off-Hook Dialing

Off-hook dialing is when a user activates an audio path (handset, headset, or speakerphone) on the phone before dialing a number. When off-hook, the phone plays the dial tone, starts collecting any digits a user enters, and processes these digits according to the phone digit map.

On-Hook Dialing

In this case, a user dials the number without taking the phone off-hook first. Users must press the **Dial** softkey option to complete the dialing of the number to call. The Phone then processes the entered number according to the digit map.

Outbound Call Routes

After a user dials a complete number, the phone determines which service to use for the call by applying the call routing rules defined in the **Phone Settings::OutboundCallRoute** parameter. This parameter may refer to named digit maps defined elsewhere in the phone configuration. The default value as shown below refers to (Msp1), (Msp2), (Msp3), (Msp4), and (Mpp), which are digit maps defined for SP1, SP2, SP3, SP4, and OBiTALK services, respectively:

```
{([1-9]x?(Mpli)):pp},{**0:aa},{**:*:aa2},{(<**1:>(Msp1)):sp1},
{(<**2:>(Msp2)):sp2},{(<**3:>(Msp3)):sp3},{(<**4:>(Msp4)):sp4},{(<**9:>(Mpp)):pp},{(Mpli):pli}
```



The `pli` refers to the Primary Line.

Primary Line

The Primary Line is the preferred line to use for outgoing calls when the number entered by users doesn't include a line-section-prefix (such as `**1` or `**2`). Define the Primary Line in the **Phone Settings::PrimaryLine** parameter using one of the following case-sensitive values:

- SP1 Service
- SP2 Service
- SP3 Service
- SP4 Service
- SP5 Service
- SP6 Service
- OBiTALK Service
- Trunk Group 1

- Trunk Group 2

The reserved name `pli` found in the phone **DigitMap** and **OutboundCallRoute** parameters is substituted by the corresponding name of the primary line: `sp1`, `sp2`, `sp3`, `sp4`, `sp5`, `sp6`, `pp`, `tg1`, and `tg2` for SP1, SP2, SP3, SP4, SP5, SP6, OBiTALK, Trunk Group 1, and Trunk Group 2, respectively.

Explicitly Selecting a Line to Make a Call

If you don't select a line to use, the phone picks the configured primary line as the preferred line to make the call. You can explicitly select a line to use in one of these ways:

- Press the corresponding softkey option hidden under the Lines softkey on the phone screen.
- Press a Call Appearance Key associated with a Line (if defined and available. See [Call Keys](#) section).
- Press the Line Monitor Key for the Line (if defined).

If the selected line is different from the primary line, the phone temporarily reassigns the selected line as the primary line when performing digit map validation and call routing for the current call.

Dialing “Speed Dials 99” Numbers

You can define as many as 99 speed dial numbers in the configuration. Speed Dial numbers are placed as 1 – 99, and you can define the corresponding one/two-digit number (that is, 1 – 99) in **User Settings - Speed Dials**. These speed dial numbers are unrelated to the Speed Dial feature keys.

You can associate the speed dial numbers with phones or devices reachable via an Internet or landline service or the OBiTALK network. Be careful with the Speed Dial setup as this may conflict with any Speed Dials set up on the OBiTALK portal. The Speed Dials you set up on the OBiTALK portal always overwrite anything setup via the phone connected to the OBi. It's also possible for a user to set up a Speed Dials 99 number using a star code (by default, the star codes for this are *74, *75, and *76).

Called Party Caller ID Display

When making an outgoing call, the phone attempts to match the called number against the phone book or buddy list to find a corresponding entry. If it finds one, the phone uses that entry to provide a name and picture of the called party to show on the screen.



If a user makes a call over a SIP service, the service provider can offer a similar feature by sending called party information in a 1xx response to the INVITE or in an UPDATE request sent to the phone before the call is answered. The phone then updates the name and number display accordingly on the screen. This feature is also referred to Calling Line ID.

Handling Incoming Calls

Inbound Call Routes

Incoming calls come to the phone via any configured Line, how the phone should route the incoming call is based on the rules defined in the **InboundCallRoute** parameter of each Line. The typical way to route an incoming call is to ring the phone, so the default **InboundCallRoute** value for all services is `ph`.

Rejecting Incoming Calls

To reject a ringing call, press the **Reject** softkey as prompted by the phone display.

Caller ID Display

During a call, the caller's name and number displays on the screen when available. When only the caller number is available, the phone attempts to match the number against the phone book or buddy list to find a corresponding entry. If the phone finds a matching entry, it displays the name and picture from that entry on the screen.

If the call comes from a SIP service, the service provider provides the caller-id picture in the INVITE request by including a Call-Info header with a (http or https) URL to the picture file and the attribute "purpose=icon". For example:

```
Call-Info: <http://abc.com/user/somepic.png>;purpose=icon;org="ABC Publishing"
```

These are the acceptable picture formats: png, bmp, gif, and jpg. Notice that in the last example, the header includes an org attribute. This value propagates the **Organization** field in the Caller-ID display.

You can also specify an Action URL in the Call-Info header to show caller information when a user presses the **ciurl** softkey (you must include this key in one of the applicable softkey Set):

```
Call-Info: <https://abc.com/user/info.php?user=john.j.smith>;purpose=info
```

Ending Calls

For calls in any state, you can end the call by pressing the **End** softkey. For calls in connected state, you can end the call by hanging up (that is, on-hooking the phone).

Holding Calls

During a connected call, users can place the call on hold using one of the following methods:

- Press the **Hold** softkey to hold the highlighted call on the screen.
- Press the feature key assigned the Hold function. This holds all calls.

Resuming Calls

While a call is holding, you can resume the call using one of the following methods:

- Press the **Resume** softkey to resume only the highlighted call on the screen.
- Press the call key that hosts the call to resume the held call



Unlike Hold, the Resume function isn't assigned to a feature key.

“Foregrounding” a Call

Operations that foreground a call, that is, bring a call to the “foreground” include:

- Resuming a held call using the **Resume** softkey
- Answering a ringing call
- Initiating off-hook dialing of a new call
- Starting a new call by on-hook dialing or calling from speed dial, phone book, or call history, and so forth

Right before carrying out a foregrounding operation, the phone automatically applies the following “backgrounding” operations on all other calls based on their respective call states:

- Calls in Connected State: hold them
- Calls in Ringing or Holding State: leave them alone
- Calls in other states: end them

Call Waiting

Call Waiting refers to when there are one or more new incoming calls while there are one or more calls in the other states. If call waiting is disabled, the phone rejects all incoming calls as busy. If call waiting is enabled, incoming calls ring on the phone if there are no connected calls. Otherwise, the phone plays the call waiting tone.

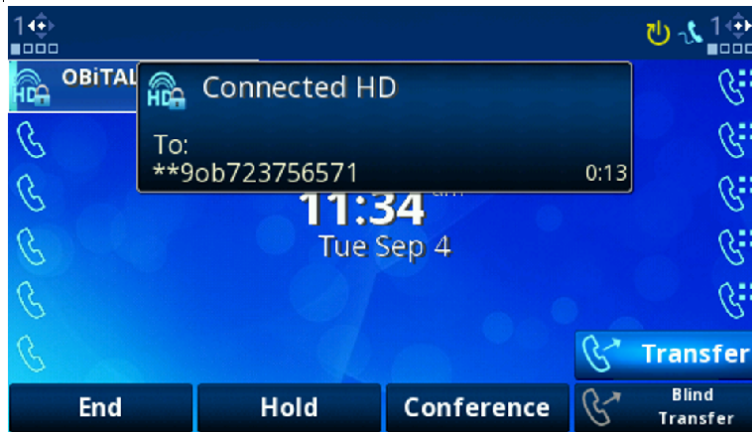
Call Transfer

Transferring a current call involves calling another remote party known as the transfer target and connecting two parties to a call while the phone user drops out of the conversation. Remote party A is the transferee, and you (the local party) are the transferor. There are three types of call transfer:

- **Attended Call Transfer:** In this case, you wait for party B to answer the call and talk to B before completing the call transfer. Then you press the **Transfer Now** softkey to complete the call transfer. This is a consulted transfer or transfer with consultation.
- **Semiattended Call Transfer:** In this case, you wait for party B's phone to ring. Before B answers to complete the call transfer, you press the **Transfer Now** softkey to complete the call transfer.
- **Unattended (or Blind) Call Transfer:** In this case, the phone completes call transfer after you enter party B's number without waiting for it to ring. You don't need to press any additional key to complete the call transfer. You won't know if B is busy or if the number entered is valid or not.

To transfer a call, first you must highlight the corresponding entry in the call list of the Calls App. The Calls App must be running on the top of the screen at the first level. The call must also be in a transferable state, which is either the Connected or Holding state. If the highlighted call is transferable, the softkeys **Transfer** and **Blind Transfer** display. Select the **Transfer** option to perform an attended or semiattended call transfer, or select the **Blind Transfer** option to perform a blind call transfer. Below is an example of a connected call with the Transfer and Blind Transfer softkeys shown.

Call transfer example



Pressing the **Transfer** softkey starts off-hook dialing and then makes the call to the dialed number. The phone applies standard handling of all current calls before the dial tone and you enter the transfer target number.

You can cancel the call transfer operation any time before the call transfer completes. While entering the target number, the phone cancels the transfer operation if you dismiss the Dial Dialog before entering a complete number. On the other hand, as soon as you enter a complete transfer target number, you can't cancel the transfer operation if it is a blind transfer. For attended and semiattended cases, the phone shows the **Complete Call Transfer** dialog along with the **Transfer Now** softkey option as shown in the picture below.

Completed call transfer example



Users can end a pending transfer operation by dismissing this dialog with the Home or Cancel Key. In that case, the phone cancels the pending transfer operation, but the call to the transfer target continues because it is reverted to a regular outgoing call. The user must explicitly end the call to the transfer target to discontinue the call. Ending the call to the transfer target at any time before call transfer completes cancels the pending call transfer. The original call, which was placed on hold automatically as a result of starting a call transfer, isn't resumed automatically when the call transfer is canceled. You must explicitly resume the call to reconnect with the transferee party.

Note also that if the transfer operation eventually fails after completion, you can't restore the original call with the transferee.

Transfer Signaling

As transferor, if the call with the transferee is on SP_m and the call with the transfer target on SP_n where SP_m and SP_n point to the same ITSP Profile X (including the case $m = n$) and **ITSP Profile X – General::SignalingProtocol** is SIP and **ITSP Profile X – SIP::X_UseRefer** is `true`, the phone attempts to transfer the call by sending a REFER request to the transferee with the transfer target's SIP URL in a Refer-To header. For other cases, the phone tries to bridge the transferee and transfer target call legs internally.

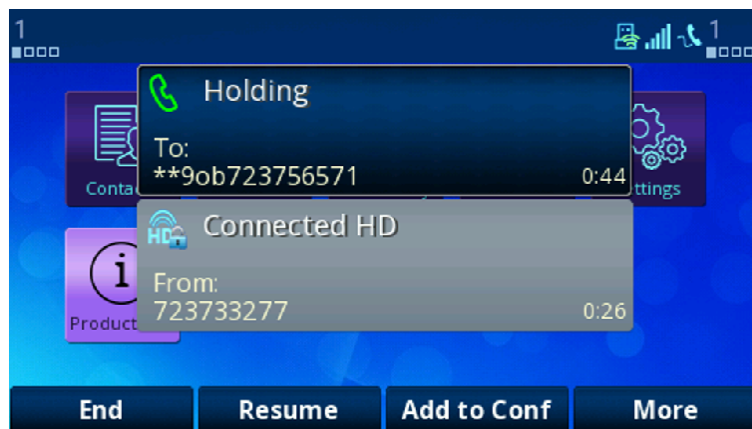
Limitations of Transfer by Internal Bridging

The phone acts as a proxy of RTP packets sent by each peer of the bridge, without any transcoding. While the phone tries to negotiate the codec to use with each call peer that would be acceptable by the other peer, the administrator should understand this limitation and configure the codec profiles accordingly. For example, if SIP service 1 is used, then a call transfer involving a SIP service 1 call leg must make sure the other call leg supports the G711U codec.

Conference Calls

A conference call is a conversation involving 2 or more remote parties. In order to start a conference, there must be at least two calls and with at least one of them in the Connected State and one of them in the Holding State. The following picture shows such scenario with two calls. To start a conference, highlight one of the calls in the Holding State and select the **Add to Conf** soft key option.

Starting a conference call



Your phone supports two methods to conference multiple parties:

- Local mixing/bridging
- External conference bridge

The UI is slightly different in each case.

Local Mixing/Bridging

After starting a 3-way conference as described earlier, you can see the two remote parties both in the Connected State.

Note: The OPUS codec doesn't support 3-way calling with both legs using OPUS.

With locally mixed N-way calls, you can see N individual call items on the screen, one for each call-leg of the N-way call. Hence you can also individually control each call-leg, such as holding or ending any one of them.

External Conference Bridge

(SIP/SP Only.) When using an external conference bridge, the conference size is not limited by the phone but by the conference bridge. Check with your conference service provider on the conference size limit. Again you can start a 3-way conference as described earlier. In this case the phone first sends a new INVITE to the SIP URL of the conference bridge to request the conference resources. If successful, the conference bridge replies a 2xx response with a Contact header that includes the context information for other conferees to access the bridge for this conference call. On that the phone maintains the call with the bridge in the Connected State and sends a REFER to both conferees to refer them to the Contact as referenced by the conference bridge. You then see that only one connected call to the conference bridge remains on the screen while the two calls with the initial two conferees are removed. Presumably the two conferees have also connected to the conference bridge on their own.

To add another conferee, make a new call to the target number (or answer a new incoming call from another party, if applicable), which automatically holds the call to the conference bridge. When the called target rings or answers, highlight the call with the conference bridge (currently in the Holding State) and select the **Add to Conf** softkey option. At that point the call to the conference bridge is resumed while the new remote party is (or will be, if it is still ringing) referred by the phone to the same conference bridge Contact (as soon as it answers) to be added to the conference bridge. You can continue to add more conferees this way until it reaches the limit set by the conference bridge.

To enable external conference bridge operation, configure the **X_ConferenceBridge** parameter on the SP*n* Service web page with the userid or (SIP) URL of the external conference bridge and also enable the option **Phone Settings::Calling Features::UseExternalConferenceBridge**. It should be noted that the phone assumes that only conferees that are on the same SP service or using the same ITSP profile as the conference bridge can be referred to the bridge. For conferees that are referable, the phone keeps them in the conference using local mixing and are subject to the local mixing limit. For example, if you have a conferee who is connected through the OBiTALK service, the phone keeps the call with that conferee in the Connected State as well as the call to the conference bridge in the Connected State and applies local mixing to the two calls.

Auto Attendant

Your phone has an Auto Attendant (AA) feature that can be invoked by including `aa` as the destination of the inbound call routing rules of a trunk (such as SP1 or OBiTALK) and have incoming calls matching those rules on that trunk routed to the AA. You can specify one of two methods to connect with the AA in the routing rules: a) Ring the AA directly and have it answer the call normally after a configurable delay, or b) Have the AA call back the current caller or another number you designated. To use the AA feature on the phone, the **Auto Attendant – Auto Attendant 1::Enable** parameter must be enabled. The phone supports only two simultaneous AA calls. Additional calls routed to the AA are rejected as busy.



In the phone configuration, the AA as described here is referred to as AA1 or Auto Attendant 1. Throughout this document and the phone configuration, AA is the same as AA1. AA2 on the other hand refers to the IVR system that is used for basic phone configuration.

AA Callback Service

Your phone offers two methods for the AA to call you back at the calling number or a number that you picked.

The first method is by statically configuring it in a trunk's **InboundCallRoute**. A rule can be added to the **InboundCallRoute** parameter to have the AA call back the caller's or any other number, if the caller hangs up before the AA answers. The rule should indicate that `aa ({callback-number})` is the target destination of the call, where `{callback-number}` is the number that the AA should call back if the caller hangs up before the AA answers the call. For example, the following rule:

```
{ (< **1> (14089913313|12121559801)) :aa ($1) }
```

says that: if 14089913313 or 12121559801 calls, the call is routed to AA. If the caller hangs up before the AA answers, the AA calls the number represented by `$1`, which is a macro that is expanded into the caller number after processing by the digit map on the left side of the colon. In this particular example, the callback number is the caller's number prepended by `**1`. The outbound service to be used for the AA to call back is determined according to the **Auto Attendant – Auto Attendant 1::OutboundCallRoute** parameter.

The **Auto Attendant – Auto Attendant 1::CallbackAnswerDelay** parameter controls the number of milliseconds before AA answers when a callback number is specified as shown in the example. The default value is 10000 ms. Without the `({callback-number})` argument, the AA behaves in the normal way and the answer delay is governed by the **AnswerDelay** parameter (in milliseconds).

The second method is by selecting AA option 3 to “Enter a callback number” after the AA answers the call, the caller explicitly enters the number to be called back by the AA. If a valid number is entered, AA says “Thank You” and “Goodbye”, and then starts calling back 2 seconds after the current call has ended. If the number entered is invalid, the AA plays the SIT tone followed by an error message. Note that the variable `$1` (representing the caller's number) is carried over to the subsequent AA callback call. The AA DigitMap can include `$1` to be used in a callback context. For example, the following rule in the AA DigitMap:

```
(<00: **1$1>|... )
```

says that if the AA dials 00, the device transforms it into the caller's number prepended by `**1`. In other words, if the caller wants the AA to call back the current number (typically the case), they can enter `00#` after selecting option 3 on the AA menu. Note that `$1` can only be used as part of a substitution element in the digit map. It must not be used for matching elements since its value is unknown.

Automated Attendant

AA Option	Default AA Announcement	What Happens Next:
1	Press 1 to continue this call.	Ring the phone
2	Press 2 to make a new call.	If “UsePIN” authentication is enabled and you enter a matching PIN, the AA immediately prompts you to enter the PIN followed by the pound (#) key. If the entered PIN is not a match, the AA gives you two additional attempts to enter the PIN. If the third attempt does not match, the AA announces a thank you message and disconnects the call.
3	Press 3 to enter a callback number.	If a valid number is entered, AA says “Thank you” and “Goodbye”, hangs up, and then calls the number back in 2 seconds. If the given number is invalid, AA plays SIT tone followed by an error message. Tip: You can dial 00# to have the AA call back your current number.

Expanded Call Features

Expanded call features go beyond the basics to provide additional features for enhanced productivity and control. Unless noted otherwise, the features described here can be performed by the phone independent of a softswitch.

Auto Answer and Intercom

Intercom, sometimes (called 2-way) paging, refers to a call with the following characteristics:

- The called phone answers automatically, usually immediately without ringing. If the handset is on the cradle, the speakerphone or the headset is turned on automatically as the call is answered. Typically the answering phone also plays a short beep to alert the called user right after answering the call. Use the option **User Preferences::IncomingPageAlertTone** to control whether the phone plays a beep when auto-answering. You can also tell the phone auto-mute when auto-answering with the option **User Preferences::AutoAnswerAutoMute**.
- The calling phone may support PTT (Push-To-Talk) in addition such that the call ends as soon as the caller releases the respective PTT key on the calling phone. PTT does not apply to the called phone. The called user talks and ends the call normally just like any other call.

In this document, an Intercom call isn't referred to as paging so it won't be confused with another paging application called Page Groups, which is based on Multicast/RTP/RTCP. See [Multicast Page Groups](#) for more information.

The phone supports two methods to signal to the called device to auto-answer the call:

- Call-Info: The phone inserts an answer-after=0 parameter in a Call-Info header in the INVITE request
- Alert-Info: The phone inserts info=alert-autoanswer;delay=0 parameters in an Alert-Info header in the INVITE request

The method to use is controlled by the **ITSP Profile X – SIP::X_AutoAnswerMethod** parameter.

Regardless of the method selected, for incoming calls, your phone processes the “answer-after” parameter in a Call-Info header or the “info” and “delay” parameters in an Alert-Info header, whichever is present in the inbound INVITE, by automatically answering the call after ringing for the number of seconds specified in those parameters (usually 0). When the call is automatically answered, all other current calls on the phone are interrupted the standard way as this is a new call added to the foreground. The auto-answer behavior can be turned off by disabling the **Phone Settings – Calling Features::AutoAnswerEnable** option. You can change the setting from the phone under the **Preferences** menu. The administrator may further define a feature key with the `Auto Answer On/Off` function to give you a shortcut to disable auto-answering intercom calls whenever you do not want to be interrupted. The color of the LED reflects the current auto-answer on or off state as a visual reminder to you. If **AutoAnswerEnable** is turned off, the intercom call is presented to you as a normal incoming call and rings the phone normally.

Depending on the method selected, for outgoing calls, your phone may request the called party to auto-answer by inserting either “answer-after=0” parameter in a Call-Info header or “info” and “delay” parameters in an Alert-Info header in the outbound INVITE request. This behavior can be invoked by dialing a star code that invokes the barge-in (or bar for short) action (*96 by default) before the call and it only applies to the next immediate outgoing call. However, the softswitch may or may not recognize this parameter or pass it through to the called phone.

A softswitch may have its own special way of letting a phone user invoke auto-answer feature on the called party. For example, FreeSwitch uses the feature code 8+{extension} to signal auto-answer. For example, you can dial 81002 to request extension 1002 to auto-answer the call.

Auto-Answer Incoming Call Based on Inbound Call Routing Rules

You can let the phone auto answer certain incoming calls based on inbound call routing rules by specifying a rule that routes the call to ph(autoans). For example:

```
{someid:ph(autoans;nobeep)},{(@.4089991234):ph(autoans;delay=2)},{ph}
```

The autoans syntax supports two optional semi-colon separated attributes:

- **nobeep**: Not to play a beep tone on answering. Default is to play the beep tone according to user preference setting
- **delay={value in seconds}**: The number of seconds to ring before auto-answering the call. Default is 0

Auto-Answer when Caller Requests to Barge-In (with OBiTALK Service)

In this scenario, the caller requests the called phone to auto-answer in order to join (or barge-in) an active call currently on the called phone. Many softswitches offer you this feature by performing the audio mixing at the server end without the phone knowing anything about the operation.

With its proprietary signaling, your phone offers the Barge-In feature at the device end with the OBiTALK service (that is available to OBi-to-OBi calls only). The calling phone requests to barge-in when you dial a star code to enable the star code variable `$barge-in1` before dialing the target OBi number. The following star code setup example uses `*98` for this purpose:

```
*98, Barge In, set($barge-in1,1)
```

The called phone automatically answers and conferences in the calling phone, subject to resources available at that moment.



The barge-in operation is only applicable if there is at least one active call on the called phone. Otherwise, the incoming call to barge-in is rejected.

Barge-In as a Coach

The calling phone to barge in may in addition request to coach the called phone instead of fully participate in the active call. The calling peer that barges in is known as the Coach, while the called peer is known as the Coachee. The original peer at the other end of the active call that the coachee has been talking to is referred to as the Patron. Precisely, coaching means: The coach hears a mix of the coachee and patron speeches, the coachee hears a mix of the coach and patron speeches, but the patron hears only the coachee. To request to coach the called peer, the coach dials a star code to enable the star code variable `$coach1` before dialing the coachee OBi number. The following star code setup example uses `*99` for this purpose:

```
*99, Barge In, set($coach1,1)
```



This feature is also known as ‘Whispering’ in some literature.

Push To Talk

The phone supports push-to-talk mode with the feature key function Speed Dial, Busy Lamp Field, and Page Group 1 and 2. See the corresponding feature key section on how to enable the PTT mode with each function.

Speed Dial

Speed Dial Feature Key

The phone administrator may allocate one or more feature keys on the phone to be used as Speed Dials. To find out which feature keys are set up as Speed Dials, you can:

- For a VLK, check the function icon on the respective VLKW on the display
- For any feature key, press and hold down the key until the respective feature key item is shown on the display. Then check the function icon of the feature key item.

Note that only the administrator can designate the function for each feature key. The user is not able to change the function of a feature key as assigned by the admin. However, you can configure the **Number** and **Service** parameter of a speed dial feature key from the phone. You can press and hold down the key until the corresponding feature key item is shown on the display, then view or change the target number and the service (or line) to use when calling from that speed dial key. To call from a speed dial key, press and release the feature key normally.

- To enable the PTT mode on a speed dial, the administrator must include the ptt flag in the Number parameter of the speed dial.

The general syntax is **Number** = {target-number} [;ptt] [;send={digit-codes}] where {target-number} can be an empty value if the number is unassigned, and {digit-codes} is a sequence of the following case-sensitive codes:

- 0-9, *, #, a, b, c, d – The DTMF digit to send to the peer. Each digit is sent with 100 ms on and 100 ms off
- S – Pause for 3 seconds
- s – Pause for 1 seconds
- U“{prompt}” – Prompts you to enter one more digits manually on the phone with the given {prompt} shown on the screen. You then press the **OK** softkey to continue
- A – Wait for the called party to answer before continuing



The phones starting executing the first code in {digits} when the call receives early media or when the call is answered, whichever happens first.

For example: `send=Ass1234U“Enter Passcode”5678`

Note that if a voice service is specified for the speed dial feature key, either in the Service field or the value of the Number field is a full number (one that includes voice service information), the phone does not apply digit map before calling the speed dial. Hence any *codes in the number are not processed and the result may not be desirable. You can include the *codes to be processed by the phone by enclosing them in a pair of [...] at the beginning of the number field. For example: you may set the number field to [*96]SP1(2113).

The phone then interprets *96 locally (to request auto-answer on the called party) and makes the call to 2113 using SP1 service.

Speed Dial Numbers

Each phone supports 99 speed dial numbers. The 99 speed dial slots are numbered from 1 to 99 and are invoked by dialing a 1 or 2-digit number corresponding to the slot number. Speed dials may be dialed from the phone or via the Auto Attendant.

The 2-digit numbers "01", "02", ..., "09" are not valid. You must dial the 1-digit number "1", "2", ..., "9" for slot number 1 through 9.

Speed dial values can be set using the configuration web page, remote provisioning, or star code (see the [Controlling Calls Using Star Codes](#) section in this document for more details). The value may be a number just like the one you normally dial, with or without any service access code prefix, such as: **9200112233, **214089991123, 4280913, and so forth. It may also include explicit trunk information with the general format TK(number), where TK= SPn (n = 1 - 6), or PP. For example, PP(ob200112233), SP2(14089991123), and so forth.

If trunk information is not specified in the speed dial entry, your phone applies **DigitMap** and **OutboundCallRoute** when making the call. Otherwise, neither **DigitMap** nor **OutboundCallRoute** is applied.

Block Caller ID

Block Caller ID (or Anonymous Call) allows you, when making an outgoing call, to prevent your name and number information from appearing on the called phone. This is a feature to can be enabled for all calls on the phone or only for calls on a specific service. For calls on an SP service, the feature may be offered locally by the phone, or by the softswitch serving that SP service on the phone. For SP1 – SP6, the parameters that are related to this service are (where the X in ITSP Profile X is the value of **ServProvProfile** of the line):

Blocking Caller ID

Parameter Group	Parameter	Description
<i>ITSP Profile X – SIP</i>	X_InsertRemotePartyID	If enabled, the phone includes Remote-Party-ID header in the INVITE. This header contains similar caller information as an unmasked FROM header. In case the FROM header is masked, the user information in this header may be used for authentication or accounting purpose.
<i>ITSP Profile X – SIP</i>	X_InsertPrivacyHdr	If enabled, the phone includes a Privacy: id header when anonymous call is enabled.
<i>ITSP Profile X – SIP</i>	X_UserAnonymousFrom	If enabled, the FROM header is totally masked.
<i>SPn Service – Calling Features</i>	AnonymousCallEnable	Enable Anonymous Call feature on this line.

Blocking Caller ID

Parameter Group	Parameter	Description
<i>SPn Service – Network Provided Services</i>	AnonymousCall	Anonymous Call service is provided by the softswitch. The UI sets and gets the setting at the service provider instead.
<i>Phone Settings</i>	AnonymousCall	Enabling this option is equivalent to enabling the feature for all services.



This service requires ITSP support.

If the feature is offered by the phone for the SIP/SP service, it does the following to the outgoing INVITE:

```
From: "Anonymous" <sip:12345@abc.com>;tag=spx13040-5345425
From: "Anonymous" <sip:anonymous@localhost>;tag=spx13040-5345425
Privacy: id
```

If the feature is offered by the softswitch, the phone does nothing but to make the option available to view and set by you from the UI.

The feature can be made available in several ways by a combination of phone and softswitch. When the feature is available, the administrator lets you enable or disable the feature from the phone using one of the following methods:

- Using a star code: To use Caller ID Block for one call only, dial *67 and then the destination number. To use Caller ID Block on a persistent basis, dial *81 from the phone. After that, all calls use the Caller ID Block feature until you cancel the Caller ID Block. To cancel Caller ID Block, dial *82 from the phone.
- Using a feature key: Administrator defines a feature key with the function **Block Caller ID**.

Block Anonymous Call

Block Anonymous Call allows you to block incoming calls that has have no identifying caller ID number. Incoming calls are presented with a busy signal or busy call treatment (such as **Call Forward On Busy**).The administrator can make this feature available to end-user configurations with one of the following methods:

- Star code: To use **Block Anonymous Call**, dial *77 from the phone. To cancel **Block Anonymous Call**, dial *87.
- Feature key with the **Block Anonymous Call** function.

This feature has both phone versions and line versions.

Calling Line ID Display

When making an outgoing call, your phone accepts SIP UPDATE method from the softswitch that may contain the called party's name and number. Your phone then updates the phone screen with the called party information from the SIP UPDATE request. There are no configuration parameters for this feature.

Your phone can also identify the called party ID from the 18x and 2xx responses to the outbound INVITE message.

Call Forwarding

Call Forwarding allows you to send incoming calls to another number of your choosing. Calls can be forwarded to a number reachable from the landline service, VoIP service, or OBiTALK network. Three types of call forwarding are supported:

- Call Forward Unconditional (same as Call Forward All)
- Call Forward On Busy
- Call Forward On No Answer.

There is one set of Call Forward settings per line and one additional set at the phone level. Incoming calls on a particular line are processed by the call forwarding rules at that line level first, then at the phone level, whichever is applicable.

Call Forward Numbers

Calls may be forwarded to numbers on the same service or on another service. Therefore each call forward number stored in your phone's configuration **MUST** include call routing information to let the device know which voice service should be used to forward the call to. The general format of a call forward number is:

TK(number)

where TK is the abbreviated name of a voice service. Valid values of TK are SP n for the SP n Voice Service where $n = 1 - 6$, or PP for the OBiTALK Service.

The number to forward to must be in the final form that is acceptable by the service provider. Your phone won't apply any Digit Map or Call Routing Rules on it.

Examples: SP1(14089991234), PP1(ob200333456)

Call Forward ALL

When you use **Call Forward ALL**, all calls are immediately forwarded to the number you indicate when you enable the feature. To enable **Call Forward ALL** from your phone, dial *72. The phone prompts you to enter the number to which the calls are forwarded. Dial the number plus the # key to hear a confirmation tone. To disable **Call Forward ALL**, dial *73. A confirmation tone is played.

Call Forward on Busy

When you use **Call Forward on Busy**, all calls are forwarded to the number you indicate only when you are already engaged in a call. To enable **Call Forward on Busy** from your phone, dial *60. The phone prompts you to enter the number to which the calls are forwarded. Dial the number plus the # key and a confirmation tone is played. To disable **Call Forward on Busy**, dial *61. A confirmation tone is played.

Call Forward on No Answer

When you use **Call Forward on No Answer**, all calls are forwarded to the number you indicate only when you do not answer the call. To enable **Call Forward on No Answer**, dial *62. Your phone prompts you to

enter the number to which the calls are forwarded. Dial the number plus the # key and a confirmation tone is played. To disable **Call Forward on No Answer**, dial *63. A confirmation tone is played.

Call Forward on No Answer

Parameter Group	Parameter	Description
<i>ITSP Profile X – SIP</i>	X_Use302ForCallFoward	
<i>SPn Service – Calling Features</i>	CallForwardUnconditionalEnable	Enable Call Forward Unconditional feature on this line
<i>SPn Service – Calling Features</i>	CallForwardUnconditionalNumber	Call Forward Unconditional Number. Must be in full number format
<i>SPn Service – Calling Features</i>	CallForwardOnBusyEnable	Enable Call Forward On Busy feature on this line
<i>SPn Service – Calling Features</i>	CallForwardOnBusyNumber	Call Forward On Busy Number. Must be in full number format
<i>SPn Service – Calling Features</i>	CallForwardOnNoAnswerEnable	Enable Call Forward On No Answer feature on this line
<i>SPn Service – Calling Features</i>	CallForwardOnNoAnswerNumber	Call Forward On No Answer Number. Must be in full number format
<i>SPn Service – Network Provided Services</i>	CallForwardOnNoAnswerRingCount	
<i>SPn Service – Network Provided Services</i>	CallForwardUnconditional	
<i>SPn Service – Network Provided Services</i>	CallForwardOnBusy	
<i>SPn Service – Network Provided Services</i>	CallForwardOnNoAnswer	

Call Forward Signaling

If the called service is on SPm and the call forward target is on SPn where SPm and SPn point to the same ITSP Profile X (including the case $m = n$) and *ITSP Profile X – General::SignalingProtocol* is SIP and *ITSP Profile X – SIP::X_Use302ForCallForward* is `true`, the phone attempts to forward the caller by replying a 302 response to the INVITE request with the forward target's SIP URL in a Contact header. For other cases, the phone tries to bridge the caller and forward to the target internally.

Limitations of Call Forward by Internal Bridging

The phone acts as a proxy of RTP packets sent by each peer of the bridge, without any transcoding. While the phone tries to negotiate the codec to use with each call peer that would be acceptable by the other peer, the administrator should understand this limitation and configure the codec profiles accordingly. For example, if SIP service 1 is used, then a call transfer involving a SIP service 1 call leg must make sure the other call leg supports the G711U codec.

Do Not Disturb

Do Not Disturb (DND) allows you to set your phone to immediately forward calls made to it to the number set up as your voicemail number or account. If no voicemail account is set up, your phone returns a busy signal to the caller until you turn off **DND**. To turn on **DND** from a phone attached to your phone, dial ***78**. To turn off **DND** from your phone, dial ***79**.

Do Not Disturb (DND)

Parameter Group	Parameter	Description
<i>SPn Service – Calling Features</i> <i>n=1-6</i>	DoNotDisturbEnable	Enables Do Not Disturb feature on this line.
<i>SPn Service – Network Provided Features</i> <i>n=1-6</i>	DoNotDisturb	
<i>Phone Settings</i>	DoNotDisturb	
<i>Phone Settings</i>	DoNotDisturbFeatureProvider	

Do Not Ring

Enabling **Do Not Ring** disables the ringer. This is equivalent to silent mode. The phone screen still indicates when a call is incoming. A feature key can be assigned with the corresponding LED indicating the on or off status of **Do Not Ring**.

Message Waiting Indication – Visual and Tone Based

Message Waiting Indication notifies you when a new voice message is available. Your phone supports both Visual and Tone based Message Waiting Indication. With Tone-based Message Waiting Indication, you are alerted to a waiting message by a “stutter” dial playing when the phone is taken off-hook. Typically, this stutter tone is removed once you listen to the message(s). Visual-based Message Waiting Indication turns on a light when there is a message waiting.



The MWI LED on your phone supports only the color red. When specifying an LED pattern for MWI, use only the **R** (red) or **X** (off) color values.

Message Waiting Indication

Parameter Group	Parameter	Description
<i>SPn Service – Calling Features</i> <i>n=1-6</i>	MWIEnable	Controls the stutter tone.
<i>SPn Service – Calling Features</i> <i>n=1-6</i>	VMWIEnable	Controls the Message Waiting Light at the top edge of your phone.

Message Waiting Indication

Parameter Group	Parameter	Description
SPn Service – Calling Features <i>n=1-6</i>	MessageWaiting	This is the message waiting state. It is configurable so that the administration can clear the status if necessary. However, the administrator normally should not provision this parameter.
ITSP Profile X – SIP	MWISubscribe	
ITSP Profile X – SIP	MWISubscribeExpires	

MWI accepts an unsolicited NOTIFY for event message-summary. You may subscribe to the message-summary event package and process NOTIFY within the subscription dialog.

Multicast Page Groups

A paging group is a multicast group that every phone, on the same LAN, that has joined the group can send audio to and receive audio from the group. A multicast group is defined with a multicast address. Each phone supports two multicast groups that are configurable by the admin. The admin may also assign a feature key for users to join or leave each group. The default settings on your phone work in most LAN environments, so usually all that is required is for the feature to be enabled and a feature key mapped to each phone for the service. Push to talk can also be enabled for Page Groups. This can be configured under **Phone Settings::Page Group 1/2**. Consult your phone administrator if to see if the group paging feature is enabled at your location.

To use a group paging feature key, do the following:

- 1 If not joined (LED off), press the key once to join. Once joined, the LED is red and your phone plays audio received from the multicast channel.
- 2 After joining (LED is red), press and hold down the key to talk (PTT or Push-to-Talk) to the group. Audio received from the multicast channel is paused while you are talking.
- 3 If clamp-on option is enabled by the admin, you can enable clamp-on so you can continue to talk without needing to keep pressing down the key. To enable clamp-on, after joining, press and release the key quickly. The LED changes to blink slowly indicate clamp-on is enabled.
- 4 To leave the group after joining (and clamp-on, if available), press and release the key quickly.

Multicast Paging Groups

Parameter Group	Parameter	Description
Page Group N <i>N = 1, 2</i>	GroupName	A user friendly name to label the group on the phone UI
Page Group N <i>N = 1, 2</i>	MulticastAddress	Must be a valid IPv4 Multicast Address. Default is 224.1.1.100 for both paging groups
Page Group N <i>n = 1, 2</i>	MulticastPort	Default is 65322 for page group 1 and 65324 for page group 2. Note that each group must use a different port number

Multicast Paging Groups

Parameter Group	Parameter	Description
Page Group N <i>N = 1, 2</i>	PushToTalk	A Boolean option to control whether the page group key (softkey or feature key) should act as a PTT key. That is, you can talk only while pressing+holding the key down. Default is <code>false</code> .
Page Group N <i>N = 1, 2</i>	TTL	TTL value of outgoing (multicast) RTP packets. Default is <code>2</code> .
Page Group N <i>N = 1, 2</i>	AudioCodec	Audio codec to use for outgoing page. Default is <code>G711U</code>
Page Group N <i>N = 1, 2</i>	TxPacketSize	Outgoing RTP packetization in milliseconds. Default is <code>20</code>
Page Group N <i>N = 1, 2</i>	RTCPTxInterval	Interval in milliseconds between sending outgoing RTCP packets when paging. No RTCP packets sent if value is <code>0</code> , which is the default. Note that a RTCP Bye packet is always sent when ending an outgoing page regardless of this setting.
Page Group N <i>N = 1, 2</i>	SilenceSuppression	A Boolean option to control if Silence Suppression should be used for outgoing page. Default is <code>false</code> .
Page Group N <i>N = 1, 2</i>	ParticipantName	A string value to identify the current user who is paging to the group in outgoing RTCP packets
Page Group N <i>N = 1, 2</i>	PlayToneOnIncomingPage	A Boolean option to control whether to play a short Paging Tone before playing a new incoming page. Default is <code>true</code> .
Page Group N <i>N = 1, 2</i>	PreferredAudioDevice	Select the preferred audio device to play incoming page. Enter either <code>System</code> , <code>Headset</code> , or <code>Speaker</code> . Default is <code>System</code> to let the phone picks according to other user preferences
Page Group N <i>N = 1, 2</i>	PageTimeout	Limit of outbound page duration in number of seconds to avoid accidentally jamming the page group channel. The duration would be unlimited if the value is <code>0</code> , which is also the default.
Page Group N <i>N = 1, 2</i>	EnableJoinLeaveGroupUserControl	A Boolean option to control whether to allow you to Join or Leave a page group by pressing (and releasing it quickly) the <code>pg1/pg2</code> softkey. Default is <code>true</code> . When it is set to <code>false</code> , press the key once to talk and press it again to go back to listen. Although in that case you can't choose the leave the page group, you can still use the AutoAnswerIntercom option from the User Preferences menu not to play any of the incoming pages.



When audio is received from the group, all your current calls are interrupted in the standard way as if a new foreground call is added. When joining two groups, the audio from the two groups are mixed by the phone when received at the same time. The phone can mix as many as two RTP streams in each page group.

Music On Hold (MOH)

The phones have native support of Music On Hold (MOH). You may configure a MOH server in the **Phone Settings – Calling Features::MOHServiceNumber** parameter. The MOH Server number can be an external SIP or OBiTALK addressable device, such as pp(ob600559558) or sp3(moh-server). The expected behavior of the MOH Server device is such that when called, it automatically answers the call and starts streaming audio to the calling device.

Your phone has a built-in MOH Server that can be invoked by specifying: `an ({prompt})` (where `an` stands for the internal announcement server), and `prompt` is a specification of the source to play. The device includes a Jazz track that can be used for music on hold. To enable the built-in track, specify the MOHServiceNumber as `an (jazz)`







Premium Call Features

Busy Lamp Field (BLF)

(SIP/SP only.) BLF is a common collaborative feature for a user to monitor the extensions of other users from their phone. BLF is a feature of the monitoring phone, The monitored extension usually does not do anything special to be monitored. The detection and notification of events occurring at the monitored extensions are carried out by the softswitch.

When BLF is supported by a specific SP service on your phone, the BLF function can be assigned to a feature key bound to that service to monitor an extension in the context of that SP service (That is another extension on the same hosted voice service or PABX system). A feature key that is assigned the BLF function is referred to as a BLF Key. One BLF Key monitors one extension only. The administrator can assign as many BLF Keys as there are feature keys available, to monitor as many extensions. The following table lists the typical events that are monitored with a BLF Key. The table also shows the operation that can be invoked when the respective event occurs, and the LED pattern indicating that particular event.

List of All BLF States

Icon	BLF Event	Description	Available Operation	LED Pattern
	Ringing	Monitored extension is ringing.	Answer the call. This operation is called a Directed Call Pickup.	Fast blinking red
	Busy	Monitored extension is on a call or making a call.	Barge in to do one of the following: <ul style="list-style-type: none"> Fully participate in the call or conference. Listen to the conversation without talking. Listen to the conversation but talk only to the monitored extension (also known as coaching or whispering). There can be other variations along these lines.	Steady red
	Holding	Monitored extension is holding one or more calls.	Resume and take over the call.	Slow blinking red
	Call Parked	A call is parked against the monitored extension.	Pick up the call from the parking slot.	Periodic short blink in red
	Idle	Monitored extension is idle.	Make or transfer a call to the extension.	Steady green
	Offline	Monitored extension is offline.		Steady amber

Single Versus Multiple BLF Event Notification

There may be multiple events happening on the monitored extension for multiple concurrent calls (in various states). For example, the monitored extension may be on a connected call while an incoming call is ringing. Under these circumstances, some softswitch implementations may only notify the monitoring phones of just one event that it deems most appropriate, while some may notify all the concurrent call events. When your phone is notified with multiple call events on the same monitored extension, it selects the call event, called the primary event, to update the BLF key LED and icon with according to the following priority:

- Ringing (for one more incoming calls)
- Holding
- Call Parked
- Busy

To see all the call events, the you can invoke the corresponding Feature Key Item by pressing and holding down the key (for about 1 second) until the item shows up on the screen. Once the feature key item is selected, the screen shows a list of call events currently happening on the monitored extension. Then you can highlight the call to apply the desired operation on the call using the available soft keys.



The level of details for each call event may not be the same for all softswitches. Some implementations may include the call peer's name and number information while others may only include the call states. Your phone shows each call state as well as its call peer name and number if they are available.

BLF with Call Park Status

As described in the call park section, your phone supports two types of call park: a) Park a call against an extension (where each extension has a single parking slot can be identified by the same extension number), or b) Park a call in an available orbit out of many. In the first case, a softswitch may include the call park status of the monitored extension in the BLF event notifications sent to the monitoring phones. At the time of writing, BroadSoft is the only softswitch that is known to support this feature. As described above, your phone supports call-park status that is present in BLF notifications.

What Happens When BLF Key is Pressed

Your phone reacts to a BLF key press based on the primary event. If it is ringing, the key press triggers a Directed Call Pickup request sent to the softswitch to answer the call on behalf of the monitored station, if the operation is available on the softswitch and enabled in the phone configuration. Otherwise, your phone handles a BLF key press the same way as pressing a speed dial by calling the monitored extension. Other operations: Barge In, Call Pickup, and Resume, can only be invoked using soft keys from within the Feature Key Item screen of the BLF key.

BLF Operation: Speed Dial

When used as a speed dial, your phone determines the number to call the monitored extension according to the following logic, based on attributes specified in the **Number** parameter of the BLF key:

- If the optional `spd` attribute is specified, call that attribute, else
- If the optional `ext` attribute is specified, call that attribute, else
- Call the `{userid}` attribute, which **MUST** be specified for a BLF key

BLF Operation: Directed Call Pickup

Directed Call Pickup can be done in one of two ways: a) Feature Code or b) INVITE+Replaces. The method to use is configured under the ITSP Profile of the SP Service that is bound to the BLF key.

For the Feature Code method, the Feature Code to use for Directed Call Pickup is also configured under the same ITSP Profile; your phone sends a normal INVITE to the number formed by concatenating the Feature Code with the number of the monitored extension (the `ext` attribute of the **Number** parameter, if it exists or else the `{userid}` attribute). (BroadSoft, FreePBX)

For the INVITE+Replaces method, your phone sends an INVITE with a Replaces header that identifies to the softswitch the ringing call to pick up from the monitored extension. (MetaSwitch)

BLF Operation: Barge In

When the monitored extension is on a connected call, the monitoring phone may barge in to join the call, if the softswitch supports the operation and if the feature is enabled in the phone configuration. This operation

requires the specification of a barge-in feature code that can be configured under the ITSP Profile of the SP service that is bound to the BLF key. To barge in, your phone sends a normal INVITE to the number formed by concatenating the barge-in feature code with the number of the monitored extension (the `ext` attribute of the **Number** parameter, if exists or else the `{userid}` attribute). (BroadSoft, FreePBX)

BLF Operation: Call Pickup

When a call is parked against the monitored extension, the monitoring phone can pick up the parked call by sending a normal INVITE to the number formed by concatenating a call pickup feature code with the number of the monitored extension (the `ext` attribute of the **Number** parameter, if exists or else the `{userid}` attribute). The call pickup feature code can be configured under the ITSP Profile of the SP service that is bound to the BLF key. (BroadSoft)

BLF Operation: Resume

The Resume operation is intended to resume (and take over) a holding call on the monitored extension. At the time of writing, there is no known softswitch that supports this operation.

BLF Configuration

Select a feature key **Key N** to configure the following parameters:

BLF Configuration

Parameter Group	Parameter	Description
Key N <i>N = 1,2,3...</i>	Function	Must be <code>Busy Lamp Field</code>
Key N <i>N = 1,2,3...</i>	Service	Select the SP service that provides this function and signaling method must be SIP. For example <code>SP1</code>

BLF Configuration

Parameter Group	Parameter	Description
Key N N = 1,2,3...	Number	<p>General Syntax:</p> <pre>[{group}/][{userid}][?][;ext={extension}] [;ptt][;spd={speed-dial}]</pre> <p>where</p> <ul style="list-style-type: none"> • {userid} is the userid of the monitored extension. It may be a generic user name, a DID number, or an internal extension number assigned to that extension. • {extension} is the (callable) extension number assigned to the monitored extension if it is different from the userid of that extension. For example, an extension may have {userid} = ObihaiUser2 and {extension}=1002. If the ext field is specified, then its value is used for speed dial, directed call pickup, barge in, and call pickup. Otherwise {userid} is used. In the case of BroadSoft, it is required to use the {extension} instead of the {userid} for such operations in this context. • if the ptt flag is specified, your phone handles the call as a push-to-talk call, when the key press is interpreted as a speed dial to the monitored extension. • {speed-dial} can be used to specified an alternative number to call when pressing the key as a speed dial. This can be useful for adding some prefix digits to force certain behavior of the call. For instance, adding a prefix 8 to the extension tells the called party to auto-answer in FreeSwitch. When combined with the ptt flag, this provides the familiar Intercom user experience.
Key N N = 1,2,3...	Name	<p>A nickname or user-friendly name for the monitored extension. It can be used by the phone in some UI elements.</p>
Key N N = 1,2,3...	Group	<p>Not used.</p>
ITSP Profile X – SIP – Feature Configuration X = A-F	X_BLFSubscribeExpires	<p>BLF implementation relies on the (SIP) subscription to the dialog event package of the monitored resource. This value specifies in seconds the expires value of the subscription. The phone renews the subscription well before the nominal expiration time.</p>

BLF Configuration

Parameter Group	Parameter	Description
<i>ITSP Profile X – SIP – Feature Configuration</i> <i>X = A-F</i>	X_DirectedCallPickupMethod	Specifies the method to use for directed call pickup (answer the call that is ringing on the monitored extension). The Choice of: <ul style="list-style-type: none"> Feature Code: Phone dials the number that is formed by prefixing the number of the monitored extension with Directed Call Pickup Feature Code. This method is used by BroadSoft, FreePBX, and FreeSwitch. INVITE+Replaces: Phone sends an INVITE with a Replaces header that specifies to the server the call that it wishes to take over. This method is used by MetaSwitch.
<i>ITSP Profile X – SIP – Feature Codes</i> <i>x = A-F</i>	DirectedCallPickup	Feature Code for directed call pickup.
<i>ITSP Profile X – SIP – Feature Codes</i> <i>x = A-F</i>	CallPickup	Feature Code for call pickup.
<i>ITSP Profile X – SIP – Feature Codes</i> <i>X = A-F</i>	BargeIn	Feature Code for barge in.
<i>SPn Service – BLF Features</i> <i>n = 1-6</i>	X_BlfBargeIn	Enable Barge In option for BLF keys associated with this SPn service.
<i>SPn Service – BLF Features</i> <i>n = 1-6</i>	X_BlfCallPickup	Enable Call Pickup option for BLF keys associated with this SPn service.
<i>SPn Service – BLF Features</i> <i>n=1-6</i>	X_BlfDirectedCallPickup	Enable Directed Call Pickup option for BLF keys associated with this SPn service.

Floating BLF Key Assignment

It is possible to reserve a block of BLF keys to members of a group of extensions without hard-coding the extension for each reserved key. The idea is to SUBSCRIBE to a single resource that includes a list of extensions to monitor, instead of subscribing to each individual extension. Upon receiving a NOTIFY from the server with the list of extensions, your phone assigns each extension to a BLF key reserved to the subscribed group using an internal algorithm. To reserve a BLF key to a member of the group, specify the Number parameter of a BLF key as: **Number = {group-name}/**

For example:

Number = sales-team/

The slash (/) at the end indicates the key is reserved to a member of the group named `sales-team`. You may also specify a preferred extension to use that key with the syntax: **Number** = {group-name}/{preferred-extension}?

For example:

Number = `sales-team/1003?`

The question-mark (?) at the end indicates that 1003 is merely a suggestion and monitors the extension only if it is present in the list of extensions returned by the server in a NOTIFY message body. If 1003 is not in the extension list from the server, your phone is free to assign this reserved BLF to another key when it is running out of free keys to assign. So, if all the BLF keys are floating without specifying they are a suggestion, the order of extension assignment follows precisely the order of the extensions in the list received from the server.

SIP for BLF

For each BLF extension that does not belong to any group, your phone subscribes to the dialog event package for each extension in the context of the SP service specified in the **Service** parameter of the corresponding feature key. For extensions belonging to the same group, your phone only maintains one subscription to the group-name and none for the individual extensions in the group. In the subscribe request, your phone indicates (in an Accept header) support for the following content-types in NOTIFY message bodies to be returned by the server

- application/multipart/related
- application/rfmi+xml
- application/dialog-info+xml

For floating key assignment, your phone expects the NOTIFY message body to include a resource list (Content-Type: application/rfmi+xml) that is compatible with RFC4662. Here is an example with two extensions specified in the resource list:

```
<?xml version="1.0" encoding="UTF-8"?>
<list xmlns="urn:ietf:params:xml:ns:rfmi" uri="sip:sales-team@as.broadworks.net"
  version="0" fullState="true">
  <resource uri="sip:ObihaiUser2@as.broadworks.net">
    <name>Obihai User2 </name>
    <instance id="DNAbROacM9" state="active" cid="7jTC13@broadworks"/>
  </resource>
  <resource uri="sip:ObihaiUser9053@as.broadworks.net">
    <name>Test User</name>
    <instance id="MT8FRckGPc" state="active" cid="cJ489p@broadworks"/>
  </resource>
</list>
```

To notify the call states of a monitored extension, your phone expects the NOTIFY message body to include a dialog-info XML (Content-Type: dialog-info+xml) that is compatible with RFC4235. Here is an example:

```
<?xml version="1.0" encoding="UTF-8"?>
<dialog-info xmlns="urn:ietf:params:xml:ns:dialog-info" version="1"
  state="full" entity="sip:ObihaiUser2@as.iopl.broadworks.net">
  <dialog id="Y2FsbGhhbGYtNjI4MjM4NzU6MA==" direction="recipient">
    <state>proceeding</state>
  </dialog>
</local>
```

```

    <identity display="ObihaiUser2 ObihaiUser2">
      sip:ObihaiUser2@as.iopl.broadworks.net
    </identity>
    <identity display="ObihaiUser2 ObihaiUser2">
      tel:+12404982562;ext=2562
    </identity>
  </local>
  <remote>
    <identity display="ObihaiUser1 ObihaiUser1">
      sip:2561@as.iopl.broadworks.net;user=phone
    </identity>
  </remote>
</dialog></dialog-info>

```

When the softswitch is capable of notifying call park status for the monitored extension, it is expected the status is reported inside a dialog-info XML as well. Here is an example:

```

<?xml version="1.0" encoding="UTF-8"?>
<dialog-info xmlns="urn:ietf:params:xml:ns:dialog-info"
  xmlns:sa="urn:ietf:params:xml:ns:sa-dialog-info"
  xmlns:bw="http://schema.broadsoft.com/callpark"
  version="3" state="partial"
  entity="sip:north00@txasdev87.net">
  <dialog id="Y2FsbGhhbGYtMzM6MA==">
    <state>confirmed</state>
    <bw:callpark>
      <bw:parked>
        <identity display="Alice south">
          sip:9726987601@as.bw.com;user=phone
        </identity>
      </bw:parked>
    </bw:callpark>
  </dialog>
</dialog-info>

```

BLF With OBiTALK Service

While most BLF implementation requires the softswitch to act as the state aggregator, OBiTALK service offers a peer-to-peer version of BLF that can be quite handy. Every OBi Edition phone can notify its phone status to a configured OBi number configured in the **OBiTALK Service – Calling Features::CallStatusAggregator** parameter. The notified phone monitors each notifying phone with a BLF function key that is configured with the Voice Service field set to OBiTALK and number set to the monitored 9-digit OBi number.

The following phone status can be monitored:

- Offline
- Idle (no calls)
- Ringing (at least 1 call ringing)
- Holding (at least 1 call holding)
- Connected (at 1 call connected)

On the monitoring phone, the following operations can be applied to the monitored OBi number:

- Call (Call the monitored OBi number)
- (Directed) Pick Up (Pick up the oldest ringing call on the monitored phone)
- Barge-In (Barge in a connected call on the monitored phone)
- Coach (Coach the monitored phone on a connected call)



The call on the monitored phone to be picked up, barged in, or coached does not need to be on OBiTALK service. If the call is not on OBiTALK service, the monitored phone bridges the call with the OBiTALK call leg with the monitoring phone that initiates the operation.

Call Park and Call Pickup

Call Park and Call Pickup (or Call Retrieval) are complementary operations not unlike call hold and call resume, except the pickup (vs. resume) operation can be carried out on extensions other than the one that parks (vs. holds) the call. Generally speaking, a “parking lot” for calls with many slots can be setup at the softswitch such that a call may be parked at an unoccupied slot from one extension and picked up by the same or another extension from the same slot. In some implementations the parking slot is referred to as a park orbit.

Each parking slot can hold one call and is uniquely identified with a numeric ID (the slot-id or orbit), such as 001, 1234, 88912, and so forth. While a user is talking to someone on the phone, they can choose to park the call at an unoccupied slot, and later they (or someone else) can pick up the call from the same slot. From a usability standpoint, it may be inconvenient for the user to first find out which parking slots are available before parking a call. A simple solution is to let the system pick an available slot to park the call, in a certain predefined range when requested by a certain user. The latter case can be referred to as “valet parking”, while the former “self parking”. The problem with valet parking is that the system still needs to communicate back to the user the parking slot ID that has been chosen, so that the call may be picked up later from that slot. In some systems, this is done by showing the parking slot ID on the phone screen, which obviously only works for phones with a display. Another strategy of choosing a parking slot is to use the parking phone’s extension number as the parking slot identifier, for easy memorization. You can think of each extension as having an associated parking slot, such that one call can be parked against one extension that has this resource enabled. When the user attempts to park a call, you can park against the current extension the call is on by default, or against another extension by explicitly entering the target extension. Similarly, you can pick up a call that is parked against your current extension, or against another extension by explicitly entering the target extension. In applications where most users normally park no more than one call at a time, you can just park against your own extension; thus avoiding parking slot collisions. With this approach, some BLF implementations also include parking slot monitoring when monitoring an extension.

Call Park Methods

Your phone supports both the types of call park described above. The method to use can be set independently for each SP service, under the ITSP Profile bound to that SP service, with the **ITSP Profile-SIP:X_CallParkMethod** parameter. The `Feature Code` method must be used for the Park-Against-An-Extension method, while the `REFER` must be used for the Park-In-An-Orbit method.

With the Feature Code method, the call park feature code must be specified under the same ITSP Profile using the **ITSP Profile-SIP-Feature Codes::Park** parameter. When user presses the “Park” softkey for a highlighted call on screen, your phone parks the call (in Holding or Connected state) against the local extension of the underlying SP service by sending a normal INVITE to the number formed by concatenating

the call park feature code with the extension number of the local extension. This INVITE (to park) is sent on the same SP service that the call to be parked is on. Note that in the context of parking a call, the local extension of the SP service is taken from the **SPn Service:X_MyExtension** parameter, if it is specified. Otherwise, the normal userid of the SP service is used. The phone UI does not have a softkey to park a call against an arbitrary extension. To do that, you must make another call by dialing the call park feature code followed by the extension to park against. Your burden can be alleviated by defining a number of speed dials with the feature code+extension preconfigured so you can press the speed dial key to park the call in one of the predefined extensions.

With the REFER method, your phone parks the call in an orbit by SIP-referring the peer of the call to be parked to the number that is the orbit to be parked at. This operation is equivalent to that of a blind transfer to a special extension. When you press the **Park** softkey of a highlighted call on screen, your phone pops up a dialog box to ask you to enter the orbit number to park the call at. Alternatively, you can press the Call Park Monitor feature key of an unoccupied park orbit to park the call with a single key press. The Call Park Monitor feature key is described in the next section.



The **Park** softkey won't be seen unless the **SPn Service–Network Provided Services:CallPark** parameter is enabled.

Call Park Monitor and Call Pickup Methods

Generally speaking, when a call is parked-against-an-extension with a feature code, you can pick it up by dialing the call pickup feature code followed by the extension number the call is parked against. To the phone this is just an ordinary call and it does not need to know, or interpret, the call pickup feature code. The user has the full responsibility of dialing the feature code and the parked-against extension explicitly and correctly. Similarly when a call is parked-in-an-orbit and if the softswitch has a way to allow calls to be picked up from an orbit that is equivalent to making a call to a special number, users can make those calls without any special support from the phone. There are scenarios when your phone can monitor if certain extensions have calls parked against them, or if certain park orbits are holding calls.




For calls parked-against-an-extension, your phone may monitor call park status by directly (SIP) subscribe to an extension's call park status (by enabling the **SPn Service–Network Provided Services:CallParkStatus** option), or indirectly through BLF monitoring of that extension (see the section Busy Lamp Field). Note that your phone supports direct subscription to call park status (in the context of an SP service) of the underlying SP's own extension only, but not an arbitrary extension. Note also that if the underlying SP is a shared line, then the direct subscription to call park status of that extension is not necessary if the softswitch already includes call park status in the notifications of the shared line status to the phone. Your phone supports call park status subscription by subscribing to the x-broadworks-callpark event package, and expects notification that includes an x-broadworks-callpark-info XML document in the message boxy (Content-Type:application/x-broadworks-callpark-info+xml). Here is an example:

```
<?xml version="1.0" encoding="UTF-8"?>
<x-broadworks-callpark-info xmlns="http://schema.broadsoft.com/callpark">
  <callpark>
    <parked>
      <identity display="Alice south">
        sip:876601@as.bw.com;user=phone
      </identity>
    </parked>
  </callpark>
</x-broadworks-callpark-info>
```

With the park-against-an-extension method, you can define a group of extensions to serve as a more generic parking lot, and use BLF keys to monitor the call park status of each key. With the `spd` attribute of the **Number** parameter, these BLF keys include the call park feature code as a prefix to act as a short-cut to park a call in an empty slot.

For calls parked-in-an-orbit, the administrator can define a feature key with the function Call Park Monitor to monitor each orbit (one feature key per orbit; each feature monitors a different orbit number). Your phone treats the Call Park Monitor function always exactly like the BLF function; it subscribes to the dialog event package for each key (or for a group of keys using the same group syntax in the **Number** parameter of the keys). It is however expected that the softswitch either notifies no calls or one call in Ringing state per orbit, as summarized in the following table.

Call Parking

Call Park Monitor Status	Description	Available Operation	Icon	LED Pattern
One Call Ringing	A call is parked in this orbit	Pickup the call. This operation is done via Directed Call Pickup		Fast blinking red
No Calls	No call parked in this orbit	Park the call on this orbit		Off
Offline	Status of the park orbit is unknown			Steady amber

The configuration of Call Park Monitor feature key is similar to that of a BLF feature key, except that the {userid}, {extension}, and {speed-dial} attributes pertain to a park orbit instead of a real extension and the `ptt` flag is not applicable.

When the user presses the Call Park Monitor key, one of the following applies:

- If there is no call parked, the highlighted call on the screen is parked
- If there is a call parked, pick up the call

Call Park Ring

When there is one or more Call Park Monitor keys defined, the Call Park Ring can be enabled to play a short ring at regular interval as an indication that at least one of the monitored call park orbits has a call parked. This feature can be enabled using the boolean parameter **Phone Settings – Calling Features::CallParkRingEnable**, which is `false` by default. There are two internal short ringtones that can be chosen for this ring: `ding` or `blorp`. Which short ring to use, as well as the ring interval and total ring duration can be set in the **User Preferences::CallParkRingtone** parameter, which takes a string value of the following general format:

```
{ring-tone-url},duration={d-value},interval={i-value}
```

where:

- {ring-tone-url} can be `ding` or `blorp` or an external URL that starts with `http://` or `https://`. For best results, the ring tone for this should be about 1 to 2 seconds long.

- `{d-value}` is the total duration in seconds to play the ring for, as long as there is a parked call. It must be greater than 0 or equal to 0 to mean play forever. The duration parameter is optional; default value is 60.
- `{i-value}` is the interval between the ring bursts in seconds. It must be greater than 0. The interval parameter is optional; default value is 10.

To let phone user change the **CallParkRingEnable** option from the User Preferences UI on the phone, add the `cpr` option in any one of the **GUI Menus::PreferencesMenu n** ($n = 1 - 6$). The default on-screen label for this option is Call Park Ring and the user can set the value to either Enable or Disable.



- The call park ring won't start in any of the following situations:
 - ▲ Phone is ringing
 - ▲ Phone is off-hook, speakerphone is on, or headset is on
- Regardless of the duration setting, the call park ring stops when any of the following happens:
 - ▲ No more calls are parked in any of the monitored parking orbits
 - ▲ Phone starts ringing
 - ▲ Phone goes off-hook, or the speakerphone or headset is turned on
- If a new call is parked while the call park ring is playing, the duration timer restarts so it can potentially ring for a full duration for the newly added call.

Tips: When call park ring is playing, an easy way to stop it (until a new call is parked) is by taking the phone off-hook or turning on the speakerphone or headset momentarily.

Shared Line and Shared Call Appearances (SCA)

(SIP/SP only.) A Shared Line is a service account or extension that is installed on a group of phones, such that if a sharing phone is using that extension, other sharing phones are notified. There can be multiple simultaneous calls on a shared line. The maximum allowed simultaneous calls on a shared line should be a fixed number. Each call on a shared line is called a Shared Call Appearance (SCA). If your phone has a shared line configured as one of the services, it should have as many call keys defined on the phones that are bound to that service. SCAs of a shared line are ordered with an index $1-n$ as SCA1, SCA2, SCA3, SCA n , where n is the maximum number of calls permitted for that line. On your phone, the SCAs are assigned sequentially in ascending order according to the VLK index of the corresponding call key bound to the shared line. For example, suppose a shared line with four SCAs is configured on a phone with VLK1, VLK3, VLK7, and VLK8 assigned as the four call keys bound to that line, then the calls hosted on VLK1, VLK3, VLK7, and VLK8 are SCA1, SCA2, SCA3, and SCA4, respectively.

There are two common implementations of SCA, namely, the Call-Info Method and the SLA/BLA (dialog) method. Use the **ITSP Profile-SIP-Feature Configuration:X_ShareLineMethod** parameter to select which method to use. It can be one of the following choices:









- `call-info`: This is the method used by BroadSoft. The phone subscribes to the call-info event package with the proxy server to receive notification of share call appearance state updates. This method also uses the line-seize event package for seizing a SCA before making a call
- `dialog;sla`: This is based on the Bridged Line Appearance draft (draft-anil-sipping-bla-02). The phone accepts subscription to the dialog;sla event package from the state-agent, and also subscribes back the same with the state-agent. The phone and state agent then exchange notifications of share call appearance state updates.

- `dialog;ma`: This is similar to the `dialog;sla` method, and is based on a more current version of the same draft (draft-anil-sipping-bla-04). The name of the event package name is changed to `dialog;ma`.


To designate an SP service as a shared line, enable the option ***SPn Service–Share line***

Features::X_ShareLine. The ***SPn Service–Calling Features::MaxSessions*** parameter should be set the value of the maximum number of call appearances allowed on the shared line. You must also have the same number Call Appearance feature keys defined to bind to this SP service. A Call Appearance key bound to a shared line behaves similarly to one that is bound to a private line, except that when the (private) call state is idle (that is, the call appearance is not being used by this phone), the SCA state is shown. The SCA state (or shared call state) indicates the call state on the sharing phone that is using that call appearance, and is communicated to this phone by softswitch via (SIP) notification. The following SCA states are supported by the phone:

List of SCA States

Icon	SCA State	Description	Available Operation	LED Pattern
	Seized	A sharing phone has seized the SCA to make an outgoing call.		Blinking red (30 ms on, 30 ms off)
	Trying	A sharing phone is trying an outgoing call.		Steady red
	Proceeding	A sharing phone is making an outgoing call and the called party is ringing.		Steady red
	Connected	A sharing phone is on a connected call with that SCA.	Barge In to: <ul style="list-style-type: none"> • Monitor the call only • Fully participate in the call 	Steady red
	Held	A sharing phone is on a call with that SCA and has placed the call on hold.	Resume and take over the call.	Slow blinking red
	Private Held	A sharing phone is on a call with that SCA and has placed the call on private hold.		Slow blinking red (100 ms on, 100 ms off)
	Call Parked	A call is parked against the extension.	Pick up the parked call.	Blinking red (50 ms on, 50 ms off)
	Idle	None of the sharing phones is using the SCA.	Seize the SCA to make an outgoing call.	Off

List of SCA States

Icon	SCA State	Description	Available Operation	LED Pattern
	Error	Protocol error.		Steady amber
	Remote Hold	Remote side placed the call on hold.	Barge In to: <ul style="list-style-type: none"> • Monitor the call only • Fully participate in the call 	Steady red

SCA implementations are based on subscribe/notify framework. With the call-info method, the subscription expires value to the call-info and line-seize events are set in **X_CallInfoSubscribesExpires** and **X_LineSeizeSubscribeExpires** (under *ITSP Profile X – SIP*), respectively. With the dialog;ma and dialog;sla methods, the subscription expires value to the respective event is controlled by the **X_DialogSubscribesExpires** parameter.

Line Seize

With any Shared line design, each sharing phone should perform a proper line-seize before attempting to make a call on a SCA. This lets the system work correctly when multiple phones are trying to make calls at the same time.

With the call-info method, a phone seizes a SCA by subscribing to the line-seize event package for that SCA, with a short expires value. If successful, the phone receives a 200 class response for the subscribe method, and also a NOTIFY to indicate that the subscription is active and what the actual expires value of the subscription allowed by the server.

With the dialog;sla or the dialog;ma method, a phone seizes a SCA by sending a NOTIFY to the state-agent with the state of the requested SCA set to `trying`. If successful, the phone receives a 200 class response for the NOTIFY request. Typically, the state-agent shortens the subscription interval with the phone that owns the SCA to detect quickly if the phone has crashed or encountered network issues.

What Happens When a Call Appearance Key is Pressed

- If the SCA is Idle, the phone tries to seize the line with a new line seize operation. You hear a Dialtone only if the subscription is successful
- If the SCA is Holding, the phone tries to resume the call
- If the SCA is Connected, the phone tries to barge-in
- If the SCA has a call parked (against its extension), the phone tries to pick up
- For other SCA states, the phone does nothing

Buddy List

A Buddy List is a contact list with presence information incorporated. Your phone supports Buddy Lists based on the XMPP standard. This feature is enabled with the option **SPn Service – Network Provided Services::BuddyList**. Note that the service can be enabled for each SP service independently. To configure your buddy list service, the following additional parameters are required:







Buddy List Parameters

Parameter Group	Parameter	Description
SPn Service – SIP Credentials	X_XmppDomain	The XMPP server domain name. The phone resolves this name as DNS A Record only. For example: impliop1.broadsoft.com
SPn Service – SIP Credentials	X_XmppUserName	The user name for authentication to the XMPP server. For example: ObihaiTester@impliop1.broadsoft.com
SPn Service – SIP Credentials	X_XmppPassword	The password for authentication to the XMPP server

You and your buddies must be using the same XMPP service in order to see the presence and status of each other (there is no XMPP federation). Each entry in the buddy list consists of the following information:

- JID (Jabber ID): Every user of an XMPP service is identified with a unique User ID called Jabber ID for historical reason. It is in the form of an email address such as: bbking@gmail.com
- Display Name (optional): The display name of the user, such as: Benjamin B. King
- Phone Number (optional): This may be a public number or an internal extension if the XMPP service is an internal service deployed within an enterprise for example
- Presence, which takes one of the following values:

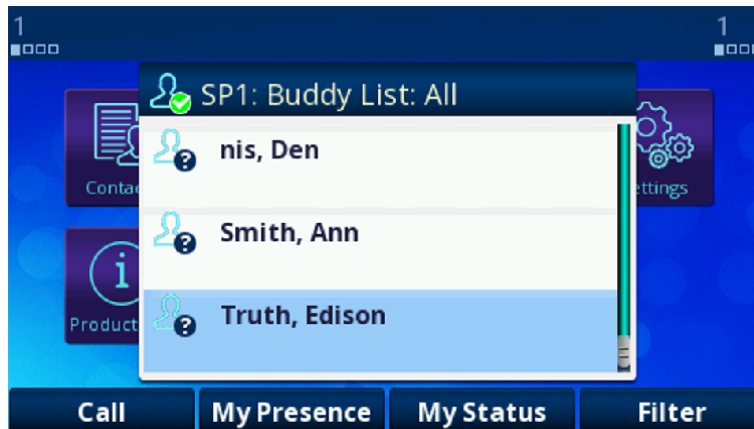
Presence Parameters

Parameter Group	Parameter	Icon
Offline	The user is offline. This is also known as “invisible” in some XMPP clients.	
Online	User is online. This is also known as “available” in some XMPP clients.	
DND	The user is busy and does not want to be disturbed. This is also known as “busy” in some XMPP clients.	
Away	The user has stepped away from the computer momentarily.	
Extended Away	This user has stepped away from the computer and may take a while to return.	
Unknown	This user's status is unknown.	

Status (optional): An arbitrary text string you enter to provide further details of your current state. For example: Having my lunch break, Back at 1 pm.

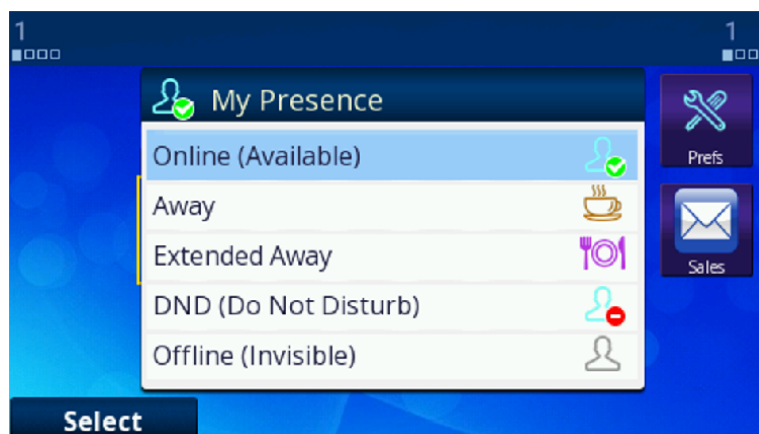
The Buddy List App is launched from the Main menu on the Home Screen. For the Buddy List App to show, it must be enabled for a service under **SPn Service – Network Provided Services::BuddyList** to display contact images, ensure you also enable the QueryVcard option further down the page. The picture below shows an example of a Buddy List.

A Buddy List filtered by the “All” group filter



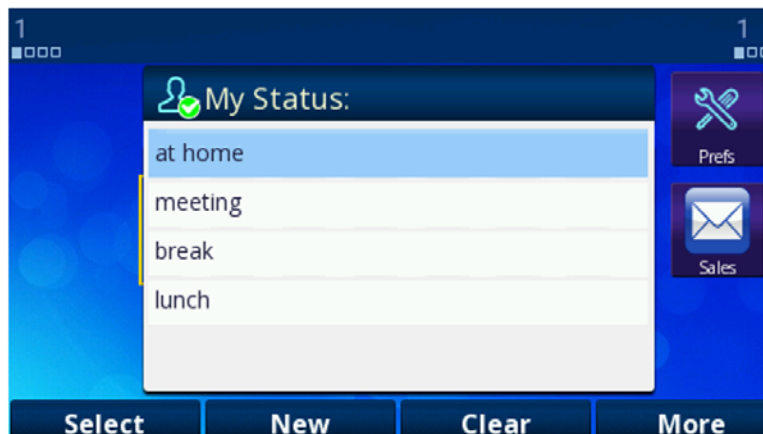
In addition to showing the buddy list as described above, the user has the option to set their own presence and status as seen by other users of the same XMPP service. The user sets their own presence by pressing the **MyPresence** softkey. The default value is *Online*. An example screen after pressing **MyPresence** is shown below. The current Presence setting is shown as an icon displayed in the title area of the screen. The value of the user selectable presence is stored in the **SPn Service::MyPresence** parameter.

Setting My Presence from the UI



The user sets their own status by pressing the **MyStatus** softkey. The default status is {blank}. Below is an example of the screen shown after pressing **MyStatus**. It shows a history of status messages that the user has entered before, with the current user status message shown in the title area of the screen. You can edit or remove an old status message, or add new ones. You can also select a different message from the list or clear the current status by pressing the **Clear** softkey. The user selected status value is stored in the **SPn Service::MyStatus** parameter. You can add new status values or remove old ones using the **New** and **Remove** softkey. All the status values are available to all SP services and are stored as a comma-separated list of phrases in the **PhoneSettings::MyStatusHistory** parameter.

Setting my status from the UI



Whenever the phone is in use (that is, off-hook), the phone automatically sets the user status to “On the phone”, with presence = DND. It restores to the last user set presence and status when the phone returns to idle.

Expanded Buddy List and Groups

If the XMPP server is from BroadSoft, your phone supports expanded buddy list and groups.

An expanded Buddy List may include basic contacts that do not have presence information. For those contacts, you won’t see a presence icon in the respective entry in the buddy list display. This is just a matter of convenience so that you can have all your contacts consolidated in one list, with or without presence information.

A Buddy List may be divided into groups such as “Friends”, “Co-workers”, and so forth. A contact may belong to more than one group or no group at all. By default all contacts in a Buddy List are in the “All” group. Most Buddy List implementation also allows the user to tag a contact as “Favorite” (sometimes called “Starred”). The phone treated this as the “Favorite” Group. Additional groups may be defined by the user.

Buddy List Management

Buddy list management refers to operations like

- Adding a contact
- Removing a contact
- Adding a group
- Removing a group
- Adding contacts to a group
- Removing contacts from a group
- Tagging or
- Un-tagging a contact as a Favorite
- Accepting presence subscriptions (invites) from other users

The phone does not support any of these management operations on the buddy list from the phone UI. Users are expected to perform such operations using compatible client software that runs on a PC, tablet, smartphone, or other device. BroadSoft users can use the BroadTouch Business Communicator client to do the same. The phone automatically picks up the changes to the buddy list from the XMPP account.

Presence Monitor

You can configure a feature key with the function `Presence Monitor` to monitor the presence of a buddy in a buddy list. Each presence monitor key monitors exactly one buddy. You can configure as many Presence Monitor keys as you need, but do not allocate the same buddy to more than one key. The following parameters are required for configuring a presence monitor key:

Function = `Presence Monitor`




Service = The SP Service where the XMPP service is offered, such as `SP3`

Number = The JID of the buddy to be monitored. This typically looks like an email address, such as: `kkytte@gmail.com`. Only the userid portion of the JID is needed. You may specify `. .` at the end such that a partial match of the JID is sufficient to find the buddy in the buddy list. For example, if the full JID of the buddy is: `abcd-12345@gmail.com`, you may specify `abcd. .` for the Number field to monitor this buddy, provided there is no other JID in the buddy list that starts with `abcd`.

Call Recording Controls

(SIP/BroadSoft.) Your phone supports the call recording functions available with a BroadSoft application server, by providing the controls for call recording during a call. This feature can be enabled on a per SP service basis by enabling the option ***SPn Service – Network Provided Services::CallRecording***. When the feature is enabled, the phone shows one of the following call recording states in the call items of the Calls App:

Call Recording Controls

Call Recording State	Description	Available Softkey Options	Icon
<code>off</code>	Recording has been turned off	Rec.Start	
<code>on</code>	Recording has been turned on	Rec.Stop, Rec.Pause	
<code>paused</code>	Recording has been turned on but paused at the moment	Rec.Resume	
<code>na</code>	Feature is not enabled		

The softkey options for recording controls options are available only when the call is in the Connected or the Holding state.

Hold and Talk Event Package

(SIP Only.) Your phone supports unsolicited notification of the hold and talk event package in the context of an SP service. Note that these notifications are sent outside of any dialogs and are not dialog specific. When the phone receives a hold event notification, it holds all the connected calls on the underlying service. When the phone receives a talk event notification, it answers all the incoming calls and resumes all the holding calls on the underlying service.

There is no configuration for these features.

Advice of Charges (AOC)

(SP/SIP Only.) Your phone accepts contents of the Content-Type: application/vnd.etsi.aoc+xml in the message body of an INFO or BYE request, or 2xx response to a BYE request. It parses the charges information and displays them on the screen in the corresponding call item of the Calls App. The final charges information received with a BYE request or 2xx response to a BYE request is displayed in a separate pop up window at the end of the call.

There is no configuration for this feature.

Emergency Calls

The administrator can define one or more numbers as emergency numbers by adding the prefix EM# to those numbers using the **Phone Settings::DigitMap** and a corresponding rule in **Phone Settings::OutboundCallRoute** to route those calls to a specific voice service to handle the call. The following example defines an emergency number 911 and routes the call to go out from SP1 when the number is dialed:

```
DigitMap = (<EM#>911|other rules ...)
```

```
OutboundCallRoute = {(911):sp1},{(311):sp2}, other rules ...
```

The phone detects that you are calling an emergency number if the number to call has the prefix EM# after applying the phone digit map on the dialed number. It then applies the emergency call treatment to that call for the duration of the call:

- You cannot hold or end the call;. Only the remote party can end it.
- You can start the call with the headset, but you cannot switch to use a headset subsequently after the call is started. You can only switch between the handset and speakerphone.
- You cannot start or resume any other calls.
- You cannot press the **Home** key to get to the Home screen to start another App.
- Call waiting is disabled.
- All the feature keys are disabled.

Call Diversion History

(SIP Only). The softswitch may keep a history of Diversion headers for each call forward transaction as it tries to ring an extension. A Diversion header is added each time the called extension redirects (or diverts) the call. When an INVITE message arrives at your phone after a series of redirections, it may include a

history of all the diversion headers (with the latest one appearing first). Your phone shows the call diversion history in the call item of the Calls App, if available.

There is no configuration for this feature.

Digit Map Configuration

A digit map can be used to match digits to ensure a complete number is dialed, transform dialed digits, and block numbers from being dialed. Its structure is a series of rules that read from left to right. Each digit map is composed of one or more rules surrounded by parentheses (). These parentheses MUST NOT be omitted.

Your phone applies the first rule that matches the format of the dialed number, so it's important you get your rule order correct.

A digit map is a group of rules written in the following fashion:

```
Digit Map = ( rule | rule | rule | rule | rule )
```

A vertical bar | separates each rule. Once again, the digit map must include the parentheses (), or your phone won't read the entered text as a digit map.

You can include "white space" within your digit map rule to make it more readable. Your phone ignores the spaces when reading the rules in the digit map. This helps make it easier to read your rules. The following example shows a rule to match phone numbers dialed to the London area code from within the United Kingdom:

```
(020[378]xxxxxxx) is equivalent to (020 [378]xxx xxxx)
```

The syntax used in this example is discussed later.

Digit Map Elements

The digit map rule serves to match numbers based on the characteristics of the form and content the entered number (or alphanumeric string, for example, when entering a SIP URI on your phone).

A rule is made up of a series of elements. Each element of the rule is matched from left to right in sequence with the entered string of numbers (or characters).

The following tables detail all available elements with which to create a rule:

<p>Any combination of: 0-9 * # + - A-Z a-z Excluding: m M s S x X</p>	<p>Literals</p> <p>It matches digit sequences with exactly the same literals.</p> <p>m, M, s, S, x, X have special meaning in the digit map syntax. Use literals to explicitly match a string.</p>
<p>Example:</p> <p>To explicitly match the New York phone number 1-212-555-7722 as dialed from within North America (according to the NANP), create the following rule from literals:</p> <p>12125557722 or 1 212 555 7722 — spaces can be added for clarity.</p>	

‘ ‘	<p>‘Quoted Literals’</p> <p>Everything inside a pair of single quotes is treated as a literal except for the single quote ‘ character itself.</p> <p>Example:</p> <p>To explicitly match the SIP URI <code>matt@sipservice.com</code>, include ‘ ‘ to surround the address, because the address includes reserved characters. Write your matching rule as:</p> <pre>'matt@sipservice.com'</pre>
-----	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

x	<p>x Wildcard</p> <p>The “lowercase x” – x – wildcard matches any digit from 0-9. It is important to note that x is case-sensitive.</p> <p>Example:</p> <p>If we consider phone numbers within a specific area code, all of the same length, we can write a rule that explicitly matches digits at the start of the number followed by any sequence of digits of a defined, fixed length. When dialing a number in the Dalton, Georgia area code of 706 from within North America, we would dial 1 706 then the following 7 digits of the number, as such we would write our matching rule as follows:</p> <pre>1 706 xxx xxxxx</pre>
---	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

.	<p>. Matching Function</p> <p>The “full stop” or “dot” - . - matching function matches 0 or more of the previous element, such as x, X or @.</p> <p>It’s typically used to capture a string of entered numbers or characters of arbitrary length.</p> <p>Example:</p> <p>The expression <code>xx.</code> would catch any number that is entered with 1 or more digits. Although we have used two x’s in our rule, the dot implies 0 or more x, so in the event of <code>0*x</code> the rule <code>xx.</code> caters for a single digit number being entered. Let’s say we want to match any international numbers dialed to New Zealand (country code 64) but don’t want to define all the possibilities of the NZ number plan within our digit map. In this case we can write one of the following rules:</p> <pre>011 64 xx. Using US international dialing notation 0011 64 xx. Using Australian international dialing notation 00 64 xx. Using Standardized international dialing notation</pre>
---	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[]	<p>Set</p> <p>“Square brackets” – [] – enclose a set of numerals and/or characters that are used to match a single digit or character.</p> <p>It’s useful to use a set to match specific digits that form part of a number. Alphanumeric and wildcard characters are allowed inside a set, such as [x], [X#], [@#], [a-zA-Zx] .</p> <p>Example:</p> <p>Let’s look at specifically matching the numbers 1,2,5,6,7 and 8. We can write this set as [125-8] where we have specified the digits 1 and 2, then written the numbers 5,6,7,8 as the sequence 5-8. To put this into context, let’s look at the London number range where each number can take the form of 020 3xxx xxxx, 020 7xxx xxxx, or 020 8xxx xxxx – while we could write each of these as individual rules, it is tidier to represent them as follows:</p> <p>020 [378]xxx xxxx</p>
------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

[^]	<p>Exclusion Set</p> <p>An exclusion set includes a leading caret, or “hat”, within a set of “square brackets” - [^] .</p> <p>An exclusion set matches any single alphanumeric character that is not within the set.</p> <p>Example:</p> <p>To match any arbitrarily long sequence of digits that does not start with * , write your matching rule as follows:</p> <p>[^*]xx.</p> <p>In the case of Sydney, Australia, local numbers are 8 digits long, starting with any digit between 3 and 9. We could write this rule as [3-9]xxx xxxx or using an exclusion set we could write:</p> <p>[^0-2]xxx xxxx</p>
-------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

!	<p>! Call Bar</p> <p>To bar users from calling numbers that match a rule, add an “exclamation mark” - ! - in front of that rule in the digit map. The rule is then referred to as a barring rule.</p> <p>Example:</p> <p>To bar all calls to 1900 numbers (regardless of length), use the following rule:</p> <p>!1900xx.</p>
----------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

< elements : literals >	<p>Element to Literal Transformation</p> <p>An element-to-literal transformation allows substituting the digit sequence matching elements with the given literals. The expression is contained within a set of “pointy brackets” - < > - and elements are separated from literals using a colon:</p>
<p>Usage Notes:</p> <p>Single quote ‘ ‘ syntax is NOT needed or allowed for the literals in this context. Special characters may be used here as they do not apply in this context either.</p> <p>Elements can be empty, in which case the colon - : - can be omitted. This case is useful for inserting some extra digits in certain part of the dialed digits.</p> <p>The literals part can be empty also, but the colon - : - MUST NOT be omitted. This case is useful for removing part of the dialed digits. Elements and literals MUST NOT both be empty.</p> <p>Example:</p> <p>Use this rule to remove digits from a dialed number, add digits to a dialed number, or transform a dialed number. To transform one number into another, for example to transform the entered digits of 112 to send out 000, write:</p> <p><112:000> Take 112 and replace it with 000 .</p> <p>Next, strip off preceding digits in a number. In this case, if you dial 02 xxxx xxxx, this rule sends on just the xxxx xxxx without the 02:</p> <p><02:>xxxx xxxx Take 02, replace it with nothing, then match the next 8 digits.</p> <p>To add to the start of a dialed number, for example to add 02 to the start of an 8-digit number, write:</p> <p><02>xxxx xxxx or <:02>xxxx xxxx</p>	

S, S0, S1 ... S9	<p>Digit Timer</p> <p>The digit timer should only be used either as the first element of a rule (for a hot or warm line implementation) or as the last element of a rule as a means of overriding the default inter-digit timer. The digit timer – S – is case-sensitive.</p>
<p>Example:</p> <p>The notation S0, S1, S2, S9 gives digit timer values of 0, 1, 2 and 9 seconds respectively. S is equivalent to S1 (1 second). S0 is the same as “blank”. You can concatenate multiple S elements together if you need more than 9 seconds timeout, such as S9S5 for a 14-second timeout. To create a hotline to the number 1-408-890-6000 we would write:</p> <p><S0:14088906000></p>	

The next two elements, (map) and (Mlabel), imply that the phone digit maps are recursive. Recursive digit maps allow digit maps to be reused and make their specification more compact and readable. It is important that you do not specify digit maps that lead to infinite recursion. For example, a digit map must not include a named embedded digit map that references itself.

(map)	<p>Embedded Digit Map</p> <p>An embedded digit map contains elements within a pair of brackets – () – that are used to match subsequent digits.</p> <p>Example:</p> <p>To match any number that starts with *74, followed by 1 or 2 digits we would write our matching rule as follows:</p> <pre>*74 (x xx)</pre>
--------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

(Mlabel)	<p>Named Embedded Digit Map</p> <p>A named embedded digit map is used for matching subsequent digits, where label refers to one of abbreviated terminal names or a user-defined digit map label. As an example, (Msp1) refers directly to the digit map contained within Voice Services > SP1 Service of the phone's native web portal.</p> <p>Example:</p> <p>Let's consider one of the phone's default digit map rules that directs a call preceded by **2 to SP2. We want to take all the digits after **2 to be matched with SP2's digit map. To do this we write our rule as:</p> <pre>**2 (Msp2)</pre>
-----------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

The following advanced elements allow you to further compact your rules for more elegant digit map syntax:

X	<p>X Wildcard</p> <p>The "uppercase x" – X – wildcard matches any digit from 0-9 as well as * making it useful for use digit maps that include star codes. It is important to note that X is case-sensitive.</p> <p>Usage Notes:</p> <p>X is equivalent to [x*] or [0-9*x]</p> <p>Example:</p> <p>To catch all three digit numbers including star codes, write your matching rule as follows:</p> <pre>XXX</pre>
----------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

@	<p>@ Wildcard</p> <p>The "at symbol" - @ - wildcard matches any alphanumeric character except #.</p> <p>Example:</p> <p>To match any arbitrarily long alphanumeric sequence (except #) that does not start with * we would write our matching rule as follows:</p> <pre>[^*]@@.</pre>
----------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

?	<p>? Matching Function</p> <p>The “question mark” - ? - matching function matches zero or one of the previous element, such as x, X or @.</p> <p>This function can be used to capture a string of entered numbers or characters of multiple known fixed lengths.</p> <p>Example:</p> <p>The expression <code>xxxx?</code> matches any entered number that is either 3 or 4 digits in length. Let’s say we live in an area where local numbers vary in length. This example uses the Brampton area in the UK that has 4- and 5-digit local numbers. Rather than using two rules in our digit map such as <code>(... xxxx xxxxx ...)</code> to match the local numbers, you can write a more elegant matching rule as follows:</p> <p><code>xxxxx?</code></p>
---	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Digit Map Rule Examples

Here are some further examples of digit map rules:

Digit Map Rules

Function	Digit Map Rule
Match any 11-digit number starting with 1-408.	<code>1 408 xxx xxxx</code>
Match any 7-digit number and prepend 1408 to the number when making the call.	<code><1408> xxx xxxx</code> or <code><:1408> xxx xxxx</code>
Match any number that starts with 011 followed by one or more digits.	<code>011xx.</code>
Add '+' to any 11-digit number that starts with 1.	<code><+>1xxxxxxxxxxx</code>
Match any number that starts with **1 020 3, 7 or 8 followed by 7 digits and remove the **1 prefix when making the call.	<code><**1:>020 [378]xxx xxxx</code>
Create a hotline to 1234.	<code><:1234> or <S0:1234></code>
Create a warm line to 1234 that is called if the user doesn't enter any digits within 4 seconds.	<code><S4:1234></code>
Match any number with at least 8 digits that ends with 8537683, such as 1 510 853 7683, 9 853 7683.	<code>xx.853 7683</code>
Match any number with at least 10 digits that ends in 408-890-6000, such as 1 408 890 6000, 001 408 890 6000, +1 408 890 6000.	<code>@. 408 890 6000</code>
Add a # to the end of any number with 1 or more digits.	<code>xx.<#></code>
Bar calls to premium rate numbers beginning with 084, 087, and 09.	<code>!08[47]x. and !09x.</code>

Now we can create an example digit map using a few of the rules above. One important function of a digit map is to determine if the user has entered sufficient digits during dialing, given the array of number combinations available, a digit map normally contains more than one rule. To create a digit map that includes

a few of the example rules in the table, we write our digit map enclosed in parentheses () and with each rule separated by a | bar as follows:

```
(<1408> xxx xxxx | @. 408 890 6000 | xx.<#> | !08[47]x. | !09x.)
```

Spaces can be used in the digit map. It is fine to include spaces to help make your digit map more readable.

Matching Against Multiple Rules in Digit Map

One important function of a digit map is to determine if sufficient digits have been entered by the user during dialing. A digit map normally contains more than one rule. The Digit Map Processor (DMP) must return the best matched rule at some point, or declare the input digit sequence is invalid. The DMP keeps refining its decision as each digit is entered until it reaches a final decision, or is forced to make a timely decision when the interdigit timer expires.

The DMP restarts the interdigit timer on every newly entered digit. The duration of this timer can be either long or short. By default, the long and the short timer values are set to 10 seconds and 2 seconds, and are configurable under the **Phone Settings** group via the **DigitMapLongTimer** and **DigitMapShortTimer** parameters respectively. Whether to use the long or short interdigit timer depends on the current rule matching states. The DMP maintains a matching state for each rule in the digit map as it processes each input digit. The following states are defined:

- **Partially Matched (PM)** – The rule partially matches the accumulated input sequence. Initially all rules are in this state before any digit is entered. Rules in this state have the potential of becoming EM or IM as more digits are entered. Example: 1234 partially matches the rules xxxxxxxx, 1xxxx, 1234567, <123:>xxxx.
- **Exactly Matched (EM)** – The rule exactly matches the accumulated input sequence. However, any further input digit turns this rule into the MM state. Example: 1234 exactly matches the rules xxxxx, 1234, 1xxx, <123:5678>x
- **Indefinitely Matched (IM)** – The rule matches the accumulated input sequence indefinitely, with a variable length such that the rule can potentially stay as IM as more matching digits are entered. Example: 011853 indefinitely matches the rules xx., 011xx., <011:>xx.
- **Mismatch (MM)** – The rule does not match the accumulated input sequence. This state doesn't change as more digits are entered. Example: 1234 mismatches the rules 123, 1xx, 12345

Rules in the EM or IM state are candidates to be selected by the DMP. After processing a new digit, the DMP returns a final decision if either of the following conditions holds:

- All rules are in the MM state. DMP returns an error.
- One or more rules are in the EM state with no rules in the IM state. DMP returns the best matched EM rule. If the best matched rule is a barring rule, DMP returns an error instead.

Otherwise, the DMP starts the short interdigit timer if there is at least one rule in the EM state, or else the long one. When the interdigit timer expires, the DMP makes a timely decision by returning the best matched rule at that moment if one is found, or else a timeout error. Again if the best matched rule in this case is a barring rule, the DMP returns an error instead.



The timer to wait for the first input digit is NOT governed by the interdigit timer, but the duration of dial tone being played and could be a lot lengthier than the long interdigit timer.

The best-matched rule is the one that has the most specific literals matching the input digit sequence. For example, the input sequence 1234 matches the rule 123x better than 1xxx. On the other hand, an EM rule is always selected over an IM rule.

Finally, the default interdigit timer can be overridden by appending the *Sn* element at the end of the rule where *n* = 0–9 designating the number of seconds to wait before triggering the rule with a timer event.

Here are some more examples:

Consider the simple digit map (<1408>xxx xxxxx). As soon as 7 digits are entered, the DMP returns a complete number by prepending the accumulated digits with 1408.

Consider another simple map (xx.). After the user dials one or more digits, the DMP returns the accumulated digits as a complete number when the long interdigit timer expires.

If we combine the last two maps into one: (xx. | <1408>xxx xxxxx). After the user dials 1 or more digits but fewer than 7 digits, the DMP returns the accumulated digits as a complete number when the (long) interdigit timer expires. As soon as 7 digits are entered, the DMP returns 1408 followed by the accumulated 7-digits when the (short) interdigit expires. On the 8th digit and beyond, however, the DMP considers the first rule only, and returns the accumulated digits as-is when the (long) interdigit timer expires.

Now add an S4 timer to the second rule: (xx. | <1408>xxx xxxxxS4). In this case, the DMP behaves exactly the same as the last, except that the short interdigit timer the DMP uses upon receiving the seventh digit is overridden by a 4-second timer. Hence, the user has as long as 4 seconds instead of 2 to dial the eighth digit.

Forcing Interdigit Timeout With The Hash/Pound (#) Key

When dialing, you can force an interdigit timeout with a # key instead of waiting for the DMP to timeout its own long or short timer. This is allowed as long as the # key does not match the current element of any PM rules. Otherwise, the # key is consumed by the DMP instead of triggering a timeout.

Consider the digit map (33xx.). If the user enters 333#, the DMP returns immediately with the number 333.

Now consider the digit map (33xx. | 333#1234x.). If the user enters 333#, the DMP doesn't return, but continues to wait for further input or its interdigit timer to expire.



The first rule "33xx." is now in the MM state since the digit # does not match "x". You can continue to enter 1234#, or 1234 and wait for a long interdigit timeout for the DMP to successfully return 333#1234.

Invoke Second Dial Tone in Digit Map

You can tell the phone to start a tone after a certain pattern of digits have been dialed by specifying the element {t=<tone>} within a digit map, where <tone> is a 1- to 3-letter name of the tone to play. The tone stops when the next digit is entered. For example:

```
(**1{t=di2}(Msp) | **8{t=od}(Msp4))
```

tells the device to play Second Dial Tone when **1 is dialed, or play Outside Dial Tone when **8 is dialed.

Here is a full list of acceptable (case insensitive) values of <tone>:

- bu = Busy Tone
- cf = Call Forwarded Dial Tone
- cm = Confirmation Tone

- `co` = Conference Tone
- `cw1` – `cw10` = Call Waiting Tone 1 through 10, respectively
- `di` = Dial Tone
- `di2` = Second Dial Tone
- `fb` = Fast Busy Tone
- `ho` = Holding Tone
- `od` = Outside Dial Tone
- `pr` = Prompt Tone
- `rb` = Ringback Tone
- `ro` = Reorder Tone (same as fast busy)
- `si1` – `si4` = SIT TONE 1 – 4, respectively
- `st` = Stutter Tone
- `0` – `9`, `*`, `#`, `a` – `d` = DTMF 0 – 9, *, #, A – D, respectively

Change Interdigit Long Timer Dynamically After Partial Match

Your phone starts off with the interdigit long timer set to the configured **DigitMapLongTimer** value when processing a new digit sequence by a digit map. You may change the long timer as some patterns are partially matched by embedding the syntax `{L=<time>}` within a rule in the digit map, where `<time>` is the desired number of seconds for the long timer.

For example: `(011 853 xxxx xxxx{L=5}x. |xx.)`. Here the long timer is shortened to 5 seconds after the user has entered `011 853` + 8 digits. Hence, your phone declares that a complete number is collected in 5 seconds when no more digits are received. Without the `{L=5}` syntax, you have to wait for 10 seconds (by default) for the same to happen.

User-defined Digit Maps

There are 10 user definable digit maps available under the **User Settings – User Defined Digit Maps** section of the device configuration web page. These digit maps are referred to as User Defined Digit Map 1 to 10. Each user-defined digit map is specified with 2 parameters:

- **Label**: An arbitrary string for referencing this digit map in other digit map specification. The value should be 2 to 16 characters long. For example, “friends”. In this case, `(Mfriends)` can be referenced in other digit maps, such as **Phone Settings::DigitMap**
- **DigitMap**

By default both parameters are empty, except for User Defined Digit Map 1 (see below).

A User-defined Digit Map For IPv4 Dialing

The default values of the parameters for User Defined Digit Map 1 are set the following values to support IPv4 dialing:

Label = `ipd`

```
Digit Map = (xx.<*:@>xx?x?<*:.>xx?x?<*:.>xx?x?<*:.>xx?x? |
xx.<*:@>xx?x?<*:.>xx?x?<*:.>xx?x?<*:.>xx?x?<*:.>xx?x?x?x?)
```

The map (*Mipd*) is referenced in the default setting of the **DigitMap** in ITSP Profile A and B. It supports the following two forms of IPv4 dialing:

- <user-id>*<a>**<c>*<d>
- <user-id>*<a>**<c>*<d>*<port>

where <user-id> is an arbitrary length numeric user-id, such as 100345, <port> is a port number in the range 0–65535, and each of <a>, , <c>, <d> is a 1- to 3-digit pattern in the range 1–255 that identifies one byte of an IP address. The dialed number is translated into <user-id>@<a>..<c>.<d> and <user-id>@<a>..<c>.<d>:<port>, respectively. Here are some examples:

1234*192*168*15*113 maps to 1234@192.168.15.113

123456*192*168*15*180*5061 maps to 123456@192.168.15.180:5061

Controlling Calls Using Star Codes

Star codes are short sequences of digits where each sequence serves as a command to your phone to perform certain operations. Each sequence usually starts with the * key followed by a 2-digit code (such as *69), hence the term star code. A typical operation to carry out is to set the value of one or more configuration parameters. Your phone enables you to issue a star code in the same way you dial a number to make a call. Every star code and its operation is defined with a short Star Code Script parameter. The set of star codes that can be dialed from the phone is collectively referred to as a Star Code Profile.

Your phone has two star code profiles available in its configuration, known as Star Code Profile A and B respectively. Each profile has 30 star code script parameters, known as Code1 to Code30. You can select which star code profile to use by setting **Phone Settings::StarCodeProfile** to A or B, or None if star code is not to be used.

A star code script is defined with the help of a number of predefined variables and actions. Each variable represents one or one group of configuration parameters. An action can be checking or setting the value of a variable, collecting a phone number from the user, or calling a certain number.

Star Code Script Variables (VAR)

A star code script variable or VAR can be trunk-specific or global (non-trunk-specific). The general format of a global variable is \$var. The general format of a trunk-specific variable is TK(\$var), where TK is the abbreviated name of a trunk (SP1 to SP6, or PP). If TK is not specified for a trunk-specific variable, it implies all the applicable trunks in the system.



SP*n* is the SP*n* Service where *n* = 1 through 6 and PP is the OBiTALK Service. Each service is also referred to as a *trunk* in this document.

Here is a list of the supported \$var:

- \$CFA = call forward unconditional enable (trunk-specific; valid values: 0 for disable, 1 for enable)/
- \$CFB = call forward busy enable (trunk-specific; valid values: 0 for disable, 1 for enable).

- `$CFN` = call forward no-answer enable (trunk-specific; valid values: 0 for disable, 1 for enable).
- `$CFAN` = call forward unconditional number (trunk-specific; valid value: a token representing a call forward number).
- `$CFBN` = call forward busy number (trunk-specific; valid value: a token representing a call forward number).
- `$CFNN` = call forward no-answer number (trunk-specific; valid value: a token representing a call forward number).
- `$MWS` = message waiting state (trunk-specific; valid value: 0 for no new messages, 1 for one or more new messages).
- `$DND` = do-not-disturb enable (trunk-specific; valid values: 0 for disable, 1 for enable).
- `$BAC` = block-anonymous caller enable (trunk-specific; valid values: 0 for disable, 1 for enable).
- `$BCI` = block outbound caller-ID enable (trunk-specific; valid values: 0 for disable, 1 for enable).
- `$CWA` = call-waiting enable (global; valid values: 0 for disable, 1 for enable).
- `$BCI1` = block caller-ID once (global; valid value: 1 for enable).
- `$UBCI1` = unblock caller-ID once (global; valid value: 1 for enable).
- `$LBM1` = Loopback media (audio samples) once in the next call.
- `$LBP1` = Loopback RTP packets once in the next call.
- `$CDM1` = Codecs to enable in the next call (temporarily overriding any codec preferences in device configuration). Each bit of its value represents one audio codec:
 - Bit0 (LSB) = G711u
 - Bit1 = G711a
 - Bit6 = G729
 Other bits are reserved.
- `$LDN` = last dialed number (for redial) (global; read-only).
- `$BAR1` = Enable Barge-In 1 on the next call (global; valid value: 1 for enable).
- `$Bxrn` = Blind Transfer Target Number (global; valid value: a token representing the target number).
- `$LCR` = last caller's number (for call return) (global; read-only).
- `$SPD[n]` = number for the speed dial n ($n = 1 - 99$) (global; valid value: literal or token representing a phone number).
- `$CODE` = the digit(s) representing the variable part of a star code (see examples below; read-only).

Variable names are case-insensitive.

Star Code Script Actions (ACT)

The general format of an action: `ACT(par, par, ...)`

The following actions are supported:

- `set (VAR, token)` = Set the given `VAR` to the value represented by `token`.
- `call (token)` = Call the number represented by `token`. **Phone Settings::OutboundCallRoute** is applied when making the call (but not the **DigitMap**).
- `rpdi (token)` = Repeat dial the number represented by `token`.

- `coll (VAR)` = Collect a number from the user and store it as the value of the parameter(s) represented by `VAR`. The number is collected with **Phone Settings::DigitMap** applied.
- `say (token)` = Display the value represented by `token` in an on-screen notification message. Values are announced as a list of alphabets or numbers, where `token` can be a literal (such as 1234) or another variable (such as `$CFAN` or `SP1 ($CFBN)`).
- `wifiap` = Enable Wi-Fi access point mode for quick Wi-Fi configuration from a web browser.
- `blst` = Add the number of the last caller to the **X_BlockCallers** list.

Action names are case insensitive.

Star Code Script Format

General Format: `code, name, action1, action2, action3, ...`

- `code` = The star code, such as *72. It may contain a variable part enclosed in parentheses, such as *74(x|xx). The variable part as entered by the user are stored in the variable `$CODE`.
- `name` = A descriptive name of the function of this star code, such as Call Forward Unconditional.
- `action1, action2, ...` = A valid action with parameters.

Actions are carried out one-by-one in the order as specified in the script.

Restrictions:

- At most 1 `coll` action per code.
- Either 1 `say` or 1 `coll` action at most per code, and it must be the last action in the script.

Star Code Script Examples

The following examples are taken from some of the default star code scripts in your phone.

- *69, Call Return, `call ($LCR)`
Calls the number of the caller who last rang your phone.
- *07, Redial, `call ($LDN)`
Redials the last dialed number.
- *72, Call Forward Unconditional, `coll ($cfan), set ($cfa, 1)`
Your phone collects a number from you according to the digit map, then sets the **CallForwardUnconditionalNumber** on all trunks to the collected value, and finally sets the **CallForwardUnconditionalEnable** on all trunks to **Yes**.
To modify the script to enable **CallForwardUnconditional** on SP1 only, change it to
➤ *72, Call Forward Unconditional SP1, `coll (SP1 ($cfan)), set (SP1 ($cfa), 1)`
- *67, Block Caller ID Once, `set ($BCI1, 1)`
Enables masking caller ID information one time for the next call on any trunk.

Customizing Devices

This section provides information on customizing Polycom OBi Edition VVX business IP phones.

Applications

Applications enable the functionalities of the OBi Edition VVX business IP phones.

Built-In Phone Applications

You can launch most of the built-in phone apps from the Home screen. The following apps are available:

Built-In Phone Applications

Display Caption (Default)	Action URL	Summary
Contacts	Phone://phone-book	A list of contacts where you can add, remove, or modify entries.
Current Calls	Phone://calls	A list of all the current calls on the phone, where you can manipulate the calls individually, or start more calls. The app also pops up automatically when a new call is added
Call History	Phone://call-history Secondary Apps: Phone://call-history/all Phone://call-history/received Phone://call-history/missed Phone://call-history/outgoing	A list of as many as 200 previous calls with time-stamp, duration, call statistics, and other information for each entry. The call history is further divided into four groups: All Calls, Missed Calls, Outgoing Calls, and Received Calls, which you can access via a corresponding shortcut as a secondary app.
Preferences	Phone://preferences	A list of user preferences settings, such as background picture, screen saver, language, screen brightness. The order of the items and contents is configurable.
Settings	Phone://settings	A list of phone settings (usually for admin) such as Network, Firmware Update. The order of items and contents is configurable.
Product Info	Phone://prod-info	Information about the product such as hardware and software versions, device MAC, and so forth. This is a read-only list. The order of the items and contents is configurable.
Buddy List	Phone://buddy	A Buddy List service based on XMPP.

You can find more information on the operations of these apps in the [Polycom VVX Business IP Phones, OBi Edition User Guide](#).

OBiPhoneXML Applications

The OBi Edition VVX business IP phones provide a programmatic interface for developing third-party applications that you can download and execute on the phones. You can develop these applications (called OBiPhoneXML apps) using a proprietary mark-up language called OBiPhoneXML to create solutions involving deployed phones and servers.

Typical applications include:

- Access to corporate directories
- Voicemail navigation
- Call queue monitoring
- Integration with customer databases

For more information, see the [Polycom OBiPhoneXML Applications API Guide](#).

Softkey Set Customization

The set of softkeys available on the phone screen at any given time is a softkey Set. The current App state determines which softkey set displays on the phone. Often, the set that displays is based on the highlighted item on a screen with multiple items listed. You can customize some of the softkey sets by entering a list of softkey specifications as a comma-separated list in the corresponding softkey parameters.

Softkey Set Parameter Syntax

The value of a softkey set parameter is a comma-separated list of softkey specifications, with the following general format: `{softkey-spec}[|{softkey-spec}]`, `{softkey-spec}[|{softkey-spec}]`, and so forth.

A `{softkey-spec}` is a softkey specification, and the `|{softkey-spec}` syntax lets you specify an optional alternative softkey to show when the given softkey is hidden. A softkey is hidden when its hidden condition is matched. The hidden condition for each softkey is defined internally. For example, the `missed|lines` option specifies that the **Missed** softkey displays when it isn't hidden, otherwise the **Lines** softkey displays. If the alternative softkey is an empty key (unspecified or not found), an **<Empty>** softkey displays as the alternative. Otherwise, a hidden softkey is packed and doesn't occupy any softkey slot on the display.

For example: the `set barge, monitor, , newcall` shows **Barge In**, **Monitor**, **<Empty>**, and **New Call** when `barge` and `monitor` aren't hidden, and **<Empty>**, **New Call** when `barge` and `monitor` are hidden. On the other hand, the `set barge|,monitor|,,newcall` shows the same when `barge` and `monitor` aren't hidden, but **<Empty>**, **<Empty>**, **<Empty>**, **New Call** when they are.

Softkey Specification

General format: `{id}[?][;attr[;attr[...]]]`

`{id}` is the standard softkey ID, as shown in the table below.

`?` is an optional syntax to designate if the key shouldn't be hidden if the hidden condition is matched.

You can include one or more `attr` attribute elements separated by a semicolon (;). Attributes are usually in the following format `{attribute-name} [= {attribute-value}]`. You can enclose `{attribute-value}` in double quotes (") to preserve all the enclosed characters. Otherwise, all the white spaces are removed. The enclosing double quotes aren't counted as part of the value.

For example: `number=**922222222, name="Echo Server", or url="http://abc.com/call.png, auth-token=abcdefg"`. That is, the phone interprets all contents within the double quotes as part of the attribute value.

All softkeys support the optional label and icon attribute, which is the text and icon that shows on the screen inside the softkey window. Without the label or icon attributes specified, the default label and icon of the softkey is used. Some softkeys that you can enable or disable offer a `label1` and `icon1` option that is used when the underlying function is in the "on" state. The **acd** softkey also supports `label2`, `label3`, `icon2`, and `icon3` options to cover the four states of an ACD Agent.

In addition, all softkeys support the optional style attribute that specifies how to style the icons and labels for the softkey. The value of the style attribute should be equal to the ID attribute of the `<style>` elements in the `<SoftKeyStyles>` XML specified for the **Screen Item Customization::SoftKeyStyles** parameter. If you don't specify the style attribute or it's not found, the `<style>` element with `id=default` is applied to the softkey. Otherwise, the softkey is styled with the internal default style.

You can specify the following aspects in a softkey style:

- Softkey background picture when the key is in normal state, pressed down or highlighted state, and alert state. You can set the picture as an internal picture file or an http or https URL.
- The placement of the label and icon within the softkey window (left/right/center/top/bottom), which you can adjust with x and y offset
- The color, size, and font for the label
- The height and width of the softkey icon and label

Assignable Softkeys

You can assign these softkeys to the configurable softkey sets.

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
aans	(User Preferences) Auto Answer Intercom Calls		Auto Answer	Auto Answer			Any
acd	ACD Agent Sign On or Off		Signed Out	Available	Unavailable	Wrapping Up	Any
achis	All Calls History		All Calls				Any

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
acturl	Action URL		XML App				Any
bac	(User Preferences) Block Anonymous Call Example: bac;label="Hide CID"; label1="Show CID"; number>**9222222222		Block Anonymous Call	Block Anonymous Call			Any
bci	(User Preferences) Block Caller ID (also known as Anonymous Call)		Block Caller ID	Block Caller ID			Any
cfa	(User Preferences) Call Forward Unconditional (also known as Call Forward All) enable/disable.		Call Forward	Call Forwarded			Any
chis	Call History		Call History				Any
cwa	(User Preferences) Call Waiting enable/disable		Call Waiting				Any
dnd	(User Preferences) Do Not Disturb		Do Not Disturb	Do Not Disturb			Any
dnr	(User Preferences) Do Not Ring		Do Not Ring	Do Not Ring			Any
emlogin	A login utility to collect a user name and password from the user and trigger ITSP provisioning (by executing the currently configured ITSP ConfigURL) and use the given user name/password for authentication. The ITSP can use this step for binding a device to the login user name.		Prov Login				Any
lcr	(Last) Call Return		Call Return				Any
ldap	LDAP directory search		LDAP				Any
ldn	Last Number Redial		Redial				Any
lines	Shows a list of available of lines on screen to choose from to make a new call.		Lines				Any
mchis	Same as missed		Missed				Any

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
missed	Missed Calls History	No New Missed Calls	Missed				Any
mwi	Message Status (also known as Message Waiting Indication)		Messages				Any
ochis	Same as redial.		Redial				Any
phbk	Invokes the Phone Book app.		Phone Book				Any
rchis	Received Calls History		Received Calls				Any
redial	Outgoing Calls History		Redial List				Any
pg1	Page Group 1		Page Group 1	Page Group 1			Any
pg2	Page Group 2		Page Group 2	Page Group 2			Any

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
sd	<p>Speed Dial. It takes four additional optional attributes:</p> <ul style="list-style-type: none"> • name: the caller name corresponding to the speed dial number • number: the speed dial number, which can be full or simple • ptt: (no value) enable Push-To-Talk option • send: {digit-codes} to automatically dial some digits after the call, where {digit-codes} is a sequence of the following case-sensitive codes: <ul style="list-style-type: none"> ▲ 0-9,*,#,a,b,c,d – The DTMF digit to send to the peer. Each digit is sent with 100 ms on and 100 ms off. ▲ S – Pause for 3 seconds. ▲ s – Pause for 1 second. ▲ U”{prompt}” – Prompt users to enter one or more digits manually with the given {prompt} shown on the screen. Users then press the OK softkey to continue. ▲ A – Wait for the called party to answer before continuing. <p>Note that the phones start executing the first code in {digits} when the call receives early media or when the call is answered, whichever happens first.</p>		Speed Dial				Any

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
sd (cont.)	<p>For example: <code>send=Ass1234U"Enter Passcode"5678</code></p> <p>It's acceptable to have the send attribute specified in a speed dial without a number (for example, <code>sd;send=1234#;label=Park</code>). In this case, the speed dial is used to send out the list of digits on a connected call, which you should only use in the CallConnected softkey set.</p> <p>Example: <code>sd;label="Echo Test"; number>**9222222222 sd;send=1234;label=Park</code></p>		Speed Dial				Any
ciurl	Invokes the Call-Info URL of the call as an action URL.	Call-Info URL not available	Call Info				Ringing, CallConnecting, CallConnected, CallHolding
dial	Calls the entered number.		Dial				Dialing, OnDialing
backspace	Removes the digit or character to the left of the cursor in the input box.		Backspace				Dialing, OnDialing
mm	Returns to the Main menu to select an item as the number to call.		Main Menu				Dialtone
mode	Switching Input Mode		Switch Mode				Dialtone, Dialing, OnDialing
switch.line	Switching Line		Switch Line				OnDialing
answer	Answers the selected incoming (and ringing) call.		Answer				Ringing

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
join	Transfers one conferee to the other in a 3-way (local-mixed) conference call.	Less than or more than two calls on the phone are currently in the Connected state.	Join				CallConnected
reject	Rejects the selected incoming (and ringing) call as busy.		Reject				Ringing
newcall	Starts a new call with dial tone.		New Call				Ringing, CallConnected, CallHolding, CallConnecting, CallPark ed, SCAInUse
end	Ends the selected call.		End				CallConnecting, CallConnected, XferTrying, XferRinging, XferConnected, ConfTrying, ConfRinging, ConfConnected
hold	Holds the selected call. This key is mutually exclusive with the Resume key.		Hold				CallConnected
privhold	Holds the selected call privately when the Call Appearance is an SCA.		Private Hold				CallConnected
resume	Resumes the selected call. This key is mutually exclusive with the Hold key.		Resume				CallHolding

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
conf			Conference				CallConnected, CallHolding
add2conf	Resumes the holding call to add to the current conference.	No other call on the phone that is currently in the Connected state	Add To Conf				CallHolding
add2conf	Answers the incoming call to directly add it to the current conference.	No other call on the phone that is currently in the Connected state	Add To Conf				Ringing
transfer	<p>Transfers the selected call. It takes these additional optional attributes:</p> <ul style="list-style-type: none"> <code>name</code>: the caller name corresponding to the speed dial number <code>number</code>: the speed dial number, which can be full or simple <p>Example:</p> <pre>transfer;label="Supervisor" ; number=sp2(1034)</pre>		Transfer				CallConnected, CallHolding

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
bxfer	<p>Blind transfers the selected call. It takes these additional optional attributes:</p> <ul style="list-style-type: none"> name: the caller name corresponding to the speed dial number number: the speed dial number, which can be full or simple <p>Note that bxfer doesn't time out on dialing the target number and doesn't apply digit map on the dialed number.</p> <p>Example: <code>bxfer;label="To Vmail"; number=sp1(*28)</code></p>		Blind Transfer				CallConnected, CallHolding
bxfer2	<p>Same as transfer softkey with respect to collecting the target number from the user, but a blind transfer is complete after the target number is collected. In contrast to bxfer, bxfer2 plays dial tone, applies digit map, and would time out as user enters the target number.</p>		Blind Transfer				CallConnected, CallHolding
cbctl	<p>Invokes conference bridge control app to view who's on the bridge or to apply certain control on the selected participants.</p>	Peer isn't a conference bridge.	Bridge Control				CallConnected
park	<p>Parks the selected call against the current extension.</p>		Park				CallConnected, CallHolding
dispcode	<p>Enters disposition code for the selected call.</p>	Disposition Code service not enabled	Dispos e Code				CallConnected, CallHolding
trace	<p>Starts a call trace for the selected call.</p>	Call Trace service not enabled.	Trace				CallConnected, CallHolding
escalate	<p>Escalates the selected call to a supervisor.</p>	Escalate service not enabled.	Escalate				CallConnected, CallHolding

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
rec.start	Starts call recording.	Call Recording service not enabled.	Rec.Start				CallConnected, CallHolding
rec.stop	Stops call recording.	Call Recording service not enabled.	Rec.Stop				CallConnected, CallHolding
rec.pause	Pauses call recording	Call Recording service not enabled.	Rec.Pause				CallConnected, CallHolding
rec.resume	Resumes call recording.	Call Recording service not enabled.	Rec.Resume				CallConnected, CallHolding
xfer.now	Completes call transfer operation		Transfer Now				XferRing, XferConnected
conf.now	Starts the conference now. That adds the conference target to the conference by resuming the original call where the conference function was started.		Conference Now				ConfRing, ConfConnected
split	Turns the call to conference target (that hasn't joined the conference yet) into a regular call, at the same call state. This is equivalent to pressing the Cancel key on the phone while the screen is showing an overlay of the call progress with the conference target.		Split				ConfTrying, ConfRing, ConfConnected
split	Breaks up a merged conference call item into individual call items on screen with each conferee and places all conferee calls on hold.	Highlighted call item is NOT a merged conference call.	Split				Conferencing
barge	Barges in on a (shared) call that is in progress or holding at another phone.	(Share) Call isn't connected or holding.	Barge In				SCAInUse

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
monitor	Monitors a (share) call that is in progress on another phone.	(Share) Call isn't connected.	Monitor				SCAInUse
pickup	Picks up a parked call at the highlighted extension	No call parked	Pickup				CallParked
mute	Selectively mutes a call.		Mute	UnMute			CallConnected, CallConnecting
coach	Requests to coach the call peer, or request to get out of coaching from the call peer		Coach	No Coach			CallConnected
coachee	Enables Coachee mode on this call, or disable Coachee mode on this call.		Coachee	No Coachee			CallConnected
nohc	No Hold New Call – Starts a new call without placing the current calls on hold first.		No Hold Call				CallConnected
divert	Diverts a ringing incoming call to a target number that user enters.		Divert				Ringing
ns.acd	Sets ACD Agent State on the selected SP <i>n</i> service.	Feature Sync Disabled	Signed Out	Available	Unavailable	Wrapping Up	NetServices
ns.bci	Turns on or off Anonymous Call feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Anonymous	Anonymous			NetServices
ns.dnd	Turns on or off Do Not Disturb feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Do Not Disturb	Do Not Disturb			NetServices
ns.exec	Turns on or off Executive Call Filtering feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Exec Filter	Exec Filter			NetServices
ns.xass	Turns on or off Executive Assistant Divert on the selected SP <i>n</i> service.	Feature Sync Disabled	Exec Assist	Exec Assist			NetServices
ns.rmoff	Turns on or off Remote Office feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Rm Office	Rm Office			NetServices
ns.simring	Turns on or off Simultaneous Ringing feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Simlt Ring	Simlt Ring			NetServices

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
ns.cfa	Turns on or off Call Forward All (also known as Unconditional) feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Call Forward	Call Forwarded			NetServices
ns.cfb	Turns on or off Call Forward Busy feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Cfwd Busy	Cfwd Busy			NetServices
ns.cfna	Turns on or off Call Forward No Answer feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Cfwd No Ans	Cfwd No Ans			NetServices
ns.secclass	Gets or sets Security Calls on the selected SP <i>n</i> service.	Feature Sync Disabled	Sec Class				NetServices
ns.dispcode	Sends a call disposition code on the selected SP <i>n</i> service.	Feature Sync Disabled	Disp Code				NetServices
ns.hotel	Gets or sets Hoteling feature on the selected SP <i>n</i> service.	Feature Sync Disabled	Hoteling				NetServices
ns.bwanw	Gets or sets BroadWorks Anywhere feature on the selected SP <i>n</i> service.	Feature Sync Disabled	BW Anywhere				NetServices
ns.dir	Loads, views, or refreshes the network directory on the selected SP <i>n</i> service.	Feature Sync Disabled	Dir				NetServices
ns.clog	Loads, views, or refreshes the network call history on the selected SP <i>n</i> service.	Feature Sync Disabled	CLog				NetServices
ns.buddy	Loads, views, or refreshes the Buddy List on the selected SP <i>n</i> service.	Feature Sync Disabled	Buddy List				NetServices
blf.call	Calls the monitored extension.		Call				BLFCall
blf.answer	Answers the highlighted call that is ringing on the monitored extension.	X_BlfDirectedCallPickup is disabled for the SP <i>n</i> service.	Answer				BLFCall

Softkey ID	Description	Hidden Condition	Default Label	Default Label1	Default Label2	Default Label3	Where
blf.pickup	Picks up a call that is currently parked against the monitored extension	X_BlfCallPickup is disabled for the SP <i>n</i> service.	Pick Up				BLFCall
blf.barge	Barges in the highlighted call that is currently active on the monitored extension.	X_BlfBargeIn is disabled for the SP <i>n</i> service.	Barge In				BLFCall
blf.monitor	Barges in to monitor only the highlighted call that is currently active on the monitored extension.	X_BlfBarge is disabled for the SP <i>n</i> service.	Monitor				BLFCall
blf.coach	Barges in to coach (also known as whisper) only the highlighted call that is currently active on the monitored extension.	X_BlfWhisper is disabled for the SP <i>n</i> service.	Coach				BLFCall

Softkey Set Parameters

The following table lists the softkey set parameters:

Softkey Set Parameters

Parameter Group	Parameter	Description
IP Phone – Soft Keys – Softkey Sets	Home	The softkey set shown on the Home Screen. The default is <code>redial,cfa,dnd,missed lines</code>
IP Phone – Soft Keys – Softkey Sets	SCAInUse	The softkey set shown on the Calls app screen when the selected call is a shared Call Appearance on another phone. The default is <code>barge ,monitor ,,newcall</code>
IP Phone – Soft Keys – Soft Key Sets	Dialtone	Dialtone plays before first digit is entered. Default is <code>redial,phbk,mode,lines</code>
IP Phone – Soft Keys – Soft Key Sets	Dialing	Dialing a number with at least 1 digit entered. Default is <code>redial,dial,mode,backspace</code>
IP Phone – Soft Keys – Soft Key Sets	OnDialing	On-hook dialing. Default is <code>switch.line,dial,mode,backspace</code>
IP Phone – Soft Keys – Soft Key Sets	CallParked	On-hook dialing. Default is <code>switch.line,dial,mode,backspace</code>

Softkey Set Parameters

Parameter Group	Parameter	Description
<i>IP Phone – Soft Keys – Soft Key Sets</i>	CallConnecting	Trying outgoing call, including the case when called party is ringing. Default is <code>end,,,newcall</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	CallConnected	Call is connected. Default is <code>end,hold,conf,transfer,privhold,park,dispcode,escalate,trace,rec.start,rec.stop,rec.pause,rec.resume,tomobile</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	CallHolding	Call is holding. Default is <code>end, resume, add2conf, conf, transfer, park, dispcode, escalate, trace, rec.start, rec.stop, rec.pause, rec.resume, tomobile</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	Ringin	Incoming call ringing. Default is <code>answer, reject</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	CallError	Outgoing call error encountered, such as called party busy or not found. Default is <code>end,,,newcall</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	XferTrying	Trying to call the transfer target. Default is <code>end</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	XferRingin	Transfer target is ringing. Default is <code>end,,,xfer.now</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	XferConnected	Connected with the transfer target, including the case when the call is Holding. Default is: <code>end,hold, resume, xfer.now</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	ConfTrying	Trying to call the new conferee <code>end</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	ConfRingin	Conferee is ringing. Default is <code>end,,,conf.now</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	ConfConnected	Connected with the conferee, including the case when the call is Holding. Default is <code>end,hold, resume, conf.now</code>
<i>IP Phone – Soft Keys – Soft Key Sets</i>	Conferencing	A virtual call item representing all the conferee calls in a conference call. All conferee calls by definition are in the Connected state. Default is: <code>end,split,newcall</code>

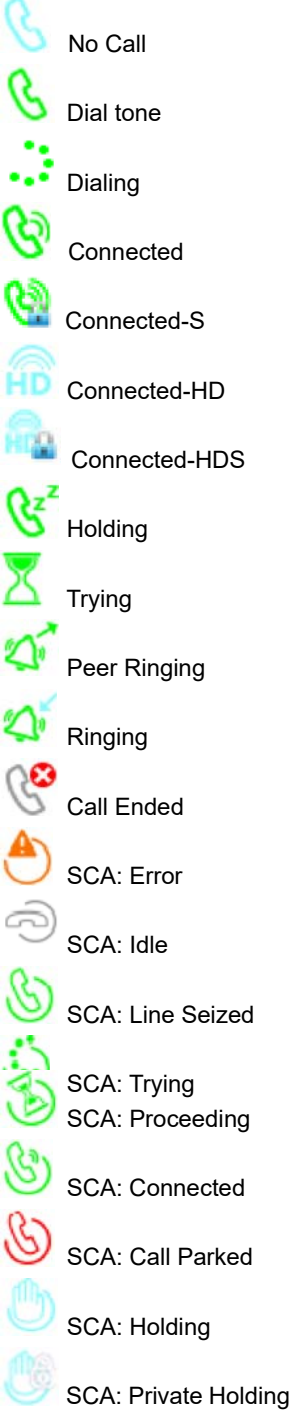





















Softkey Set Parameters

Parameter Group	Parameter	Description
<i>IP Phone – Soft Keys – Soft Key Sets</i>	NetServices	The softkeys at the top level of the Network Services app is where one of the available SP services is highlighted. Default is: ns.bci, ns.dnd, ns.cfa, ns.bac, ns.acd
<i>IP Phone – Soft Keys – Soft Key Sets</i>	BLFCall	The softkeys to present when user press+hold a BLF key to bring up a list of calls on the monitored extension. Default is: blf.call, blf.answer, blf.pickup, blf.barge, blf.ach, blf.monitor














Feature Keys

You can configure feature keys on the phone to perform one of many different predefined functions with a corresponding multicolor LED that shows the status of the assigned function instance. Virtual Line Keys, and Programmable Keys, and are all feature keys that you can configure. Phone administrators can configure each feature key to perform one of the following functions listed in this table:







List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
<p>Call Appearance</p>	<p>Makes or receives one call, and the key is known as a call key in this case. You must have an unused (idle) call key available to make or receive a new call. The phone administrator should allocate as many call keys on the phone as the maximum number of concurrent calls it's expected to handle.</p> <p>The VLKW (Virtual Line Key Window) shows nothing but the idle phone icon (shown on the right column) when no active call is assigned to the key. Otherwise, the icon changes to reflect the current call state (Call States and Call State Icons are described in the Section Making and Receiving Calls and VLKW shows call information, if available. VLKW may also change its background color to further reflect the current state.</p> <p>A call key may be bound to a voice service and is called a bound call key in that case (otherwise it's known as an unbound call key). A bound call key is used to make/receive calls on the bound voice service only. This is one of the ways for a user to select a specific line to make a call.</p> <p>A call key may be bound to a service that is a Shared line. In that case when no call is on that key, the VLKW information reflects the state of the respective Shared Call Appearance (SCA). See the Shared Line and Shared Call Appearances (SCA) section for more details on this topic.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Optional. The service or line with which to bind the key • MaxCalls: Number of calls managed by this call key 	 <ul style="list-style-type: none">  No Call  Dial tone  Dialing  Connected  Connected-S  Connected-HD  Connected-HDS  Holding  Trying  Peer Ringing  Ringing  Call Ended  SCA: Error  SCA: Idle  SCA: Line Seized  SCA: Trying  SCA: Proceeding  SCA: Connected  SCA: Call Parked  SCA: Holding  SCA: Private Holding


List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
<p>Line Monitor</p>	<p>Monitor a Line (a voice service installed on the phone). The Line events that are monitored include:</p> <ul style="list-style-type: none"> • Idle: no calls • Ringing: at least one incoming call • In Use: at least one active call • Holding: at least one call holding <p>VLKW shows the monitored service name and account user name (usually same as the account DID number or extension). This function must be bound to the specific voice service that it monitors.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Required. The service or line to monitor 	 Idle  Ringing  In Use  Holding
<p>BLF (Busy Lamp Field)</p>	<p>Monitor the call state of another extension. A BLF key must be bound to a service (as configured by the phone administrator). The call events that are monitored include:</p> <ul style="list-style-type: none"> • Ringing: at least one incoming call • Holding: at least one call holding • Busy: at least one active call • Idle: no calls • Call parked (against the monitored extension) <p>VLKW shows the bound service name and the monitored extension (or DID number, or account user name). This function must be bound to a specific voice service.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Required. The service that provides the monitoring function • Number: Required. The extension (on the specified service) to monitor. It may contain multiple attributes <ul style="list-style-type: none"> ▲ ptt – (no value). Make the call a Push-to-Talk when calling the monitored extension ▲ spd – An alternative number to call when calling the monitored extension ▲ bx – (no value). Enable one-touch blind transfer behavior 	 Idle  Busy  Call Parked  Ringing  Offline  Holding
<p>Call Park Monitor</p>	<p>Monitor the call park status of a single orbit in a parking lot. Per default LED setting, the key's LED is turned off when no call is parked in that orbit, or otherwise solid red. When the orbit isn't occupied, press the CPM key once to park the highlighted call on screen onto that orbit (the call must be in the Connected or Holding state to be parked). When the orbit is occupied, press the CPM key to retrieve the call as an additional call on the current phone.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Required. The service that provides the parking lot orbit: parking and monitoring function • Number: Required. The parking lot orbit to park and monitor 	 Idle  One Call Ringing  Offline








List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
Presence Monitor	<p>Monitor the presence/status of one buddy in a Buddy List. It also serves as speed dial to that buddy.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Required. The service that provides the XMPP service for this function • Number: Required. The JID of the buddy in the buddy list to monitor. Only the user id portion of the JID is needed. You may specify at the end such that a partial match of the JID is enough. For example, if the JID of the buddy is: <code>abcd-12345@gmail.com</code>, you may specify <code>abcd</code> for the Number field to monitor this buddy, provided there's no other JID in the buddy list that starts with <code>abcd</code>. 	 Online  Offline  Extended Away  Away  Do Not Disturb  Unknown











List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
Speed Dial	<p>If a number has not been assigned, VLKW shows no textual information. Otherwise it shows the speed dial number configured and, if the speed dial is bound to a service, the name of the service that it's bound to. If the speed dial has a display name configured, the VLIW shows the display name (such as John Seymour) instead of the assigned number.</p> <p>Parameters:</p> <p>Service: Optional. The suggested service the make the call with</p> <p>Number: Required. The number to call. It may include the following attributes:</p> <ul style="list-style-type: none"> • <code>ptt</code> (with no value): make the key a Push-To-Talk key. • <code>bx</code> – (with no value). Enable one-touch blind transfer behavior • <code>send={digit-codes}</code> to automatically dial some digits after the call, where {digit-codes} is a sequence of the following case-sensitive codes: <ul style="list-style-type: none"> ▲ 0-9,*,#,a,b,c,d – The DTMF digit to send to the peer. Each digit is sent with 100 ms on and 100 ms off. ▲ S – Pause for 3 seconds. ▲ s – Pause for 1 second. ▲ U"{prompt}" – Prompt the user to enter one more digit manually on the phone with the given {prompt} shown on the screen, then press OK to continue. ▲ A – Wait for the called party to answer before continuing. <p>Note that the phones starting executing the first code in {digits} when the call receives early media or when the call is answered, whichever happens first.</p> <p>For example: <code>send=Ass1234U"Enter Passcode"5678</code></p> <p>It's acceptable to have the send attribute specified in a speed dial without a number (for example, <code>Number=;send=1234#</code>). In this case, the speed dial can be used to send out the list of digits on a connected call.</p> <p>Note that if a voice service is specified for the speed dial either in the Service Field or the given number is a full number (one that includes voice service information), the phone doesn't apply digit map before calling the speed dial. Hence any *codes in the number aren't processed and the result may not be desirable. You can include the *codes that the phone needs to process by enclosing them in a pair of [...] at the beginning of the number field. For example: you may set the number field to <code>[*96]SP1(2113);ptt</code>. The phone then interprets *96 locally (to request auto-answer on the called party) and makes the call to 2113 using SP1 service.</p>	 Speed Dial







List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
Speed Dial (continued)	<p>Using Speed Dial to store a Feature Access Code prefix</p> <p>There are many applications where a function the Service Provider can perform by calling a feature access code prefix followed by a target number. For example, the prefix *48 followed by the target extension such as 1002 may invoke from the Service Provider the call-monitoring function on extension 1002. You can just dial *481002 directly. You can also store *48 in a speed dial and label it as Monitor on the UI. To mark the speed dial as a prefix, you append 2 or more dots to the number, such as *48... When user presses the Monitor speed dial, the phone shows a dial box on screen with *48 entered. Then you can continue to dial the target extension, or press another Speed Dial or BLF that is the target extension to monitor.</p>	 Speed Dial
Do Not Disturb	<p>Turn the Do Not Disturb feature on or off. If the function is bound to a specific voice service, it's applied to incoming calls on that service only. Otherwise, it's applied system wide to all incoming calls no matter which service the calls come from.</p> <p>Parameters:</p> <ul style="list-style-type: none"> Service: Optional. The service to apply this feature on. If no service is specified, the feature applies to all calls the phone 	 On  Off
Do Not Ring	<p>Turn the Do Not Ring feature on or off. When the feature is enabled, incoming calls come through like normally but phone doesn't play audible ring (call waiting tone is played during call waiting).</p> <p>This function can't bound to any specific voice service. It's applied system-wide to all incoming calls no matter which service the calls come from.</p> <p>Parameters: None</p>	 On  Off
Block Anonymous Callers	<p>Turn the Block Anonymous Caller feature on or off. If the function is bound to a specific service, it's applied to incoming calls on that service only. Otherwise, it's applied system wide to all incoming calls no matter which service the calls come from. If this feature is enabled, the phone rejects all incoming calls with Caller ID (name/number) hidden (blocked).</p> <p>Parameters:</p> <ul style="list-style-type: none"> Service: Optional. The service to apply this feature on. If no service is specified, the feature applies to all calls on the phone. 	 On  Off








List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
Block Outgoing Caller ID	<p>Turn the Block Caller ID feature on or off. If the function is bound to a specific service, it's applied to outgoing calls on that service only. Otherwise, it's applied system wide to all outgoing calls no matter which service is used for the calls. If this feature is turned on, the phone attempts to hide user's caller ID on outbound calls so the called party can't see who's calling.</p> <p>Parameters:</p> <ul style="list-style-type: none"> Service: Optional. The service to apply this feature on. If no service is specified, the feature applies to all calls on the phone. 	 On  Off
Call Forward All	<p>Turn the Call Forward All Calls feature on or off. If the function is bound to a specific service, it's applied to incoming calls on that service only. Otherwise, it's applied system wide to all incoming calls no matter which service the calls come from.</p> <p>Parameters:</p> <ul style="list-style-type: none"> Service: Optional. The service to apply this feature on. If no service is specified, the feature applies to all calls on the phone. 	 On  Off
Auto Answer	<p>Turn the Auto Answer (Intercom Calls) feature on or off. Normally this feature is turned on so that incoming intercom calls are answered automatically by the phone on the speakerphone or headset. If this feature is turned off, incoming intercom calls are treated as regular calls and the phone rings normally.</p> <p>This function cannot be bound to any specific voice service. It's applied system-wide to all incoming calls, no matter which service the calls come from.</p> <p>Parameters: None</p>	 On  Off
Call Waiting	<p>Turn the Call Waiting feature on or off. Normally, this feature is enabled and you can accept more incoming calls while already on a call. If this feature is turned off, all incoming calls are rejected as busy when you're on a call.</p> <p>This function cannot be bound to any specific voice service.</p> <p>Parameters: None</p>	 On  Off
Message Status (Monitor Voicemail Status)	<p>Monitor the number of messages in a mailbox. The function must be bound to a SP service that has the MWI (Message Waiting Indication) feature enabled. The VLKW shows if and how many new messages are available in the mailbox.</p> <p>Parameters:</p> <ul style="list-style-type: none"> Service: Required. The SP service that provides the Voicemail service. Number: Optional. The number to call (to check voicemail). It may include the following attributes: <ul style="list-style-type: none"> ^ mailbox={mailbox-id} where {mailbox-id} is third-party mailbox to monitor, if it isn't the same as the main mailbox for the given SP Service. Don't specify this attribute if you're monitoring the main mailbox for the given SP Service 	 Messages  No messages






List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
Hold	Hold all calls that are in the Connected State. VLKW shows how many calls are currently in Holding State. The LED turns red if there's a least one call in the Holding State, or turned off otherwise. This function cannot be bound to any specific voice service. Parameters: None	 Hold
Add to Conference	Add all calls that are in the Holding State to the current conversation. VLKW shows how many calls are in the Holding State. The LED turns red is there is at least one call in the Holding state, or turned off otherwise. This function cannot be bound to any specific voice service. Parameters: None	 Add to Conference
Join/Leave Page Group 1	Join/Leave Page Group 1. If the user joins the group, then the phone automatically turns on the speaker when anyone in the group starts a page. You can also start the page by pressing down the feature key. This is known as PTT or Push-to-Talk (otherwise you're just listening). If allowed by the phone admin, you can also "Clamp On" the feature key to talk continuously without needing to hold down the key. This function cannot be bound to any specific voice service. Parameters: None	 Left  Joined
Join/Leave Page Group 2	Join/Leave Page Group 1. If the user joins the group, then the phone automatically turns on the speaker when anyone in the group starts a page. You can also start the page by pressing down the feature key. This is known as PTT or Push-to-Talk (otherwise you're just listening). If allowed by the phone admin, you can also "Clamp On" the feature key to talk continuously without needing to hold down the key. This function cannot be bound to any specific voice service. Parameters: None	 Left  Joined

List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
Change ACD Agent State	<p>Change/monitor an ACD (or Call-Center) Agent State to one of the following values:</p> <ul style="list-style-type: none"> • Available (to take new calls) • Unavailable (to take new calls) • Signed Off • Wrapping Up (the last call) <p>ACD stands for Automated Call Distribution is the primary way a call-center distributes calls among a number of agents working for the call-center. The ACD controller should only send incoming call to an agent whose state is "Available".</p> <p>Note that an agent can't change state to "Signed Off" or "Wrapping Up" directly by pressing the feature key. To change to these states, an agent must use the corresponding feature key menu item from the UI (invoked by pressing and holding down the feature key), or some other means provided by the softswitch, such as a web portal.</p> <p>This function must be bound to a specific voice service. The ACD agent handles calls on the bound service only with respect to the underlying Call Center. The Call Center isn't aware of calls the agent makes or receives with other voice services installed on the phone.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Required. The service that provides the ACD Agent function. 	 Signed Off  Available  Wrapping Up  Unavailable
Guest User Login/Logout	<p>This feature is also known as Hoteling on some softswitches. The phone may be set up to be used temporarily by a guest, such as a visiting employee or temp worker. The guest can press the key and enter a user-id and password to log in and start using the guest phone with a personal extension temporarily (until the guest logs out or the server logs the user out remotely).</p> <p>This function must be bound to a specific voice service (that supports this feature).</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Required. The service that provides the Guest Login function. 	 Guest Logged In  Guest Logged Out
Enter Disposition Code for the last call.	<p>Enter a Disposition Code for the last call. This is used by a Call Center agent to enter a disposition code for the last customer call. This function must be bound to a specific voice service (that supports this feature).</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Required. The service that provides the Disposition Code function 	 Enter Disposition Code

List of Feature Keys and Their Functions

Feature Key Function	Description	Icon
Next Tab	<p>Press this key to switch to the next VLK Tab or cycle back to VLK Tab 1.</p> <p>This function can't be bound to any specific voice service.</p> <p>Parameters: None</p> <p>Note that the multiple Line Key Tabs feature can be disabled also by disabling the option User Preferences::LineKeyTabs. In that case the "Next Tab" function does nothing.</p>	 Next Tab
Transfer	<p>Invoke the call transfer function on the currently highlighted call on the screen when the Calls App is at the top of the display stack. The call must be in a transferable state, that is, Holding or Connected State.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Service: Optional. The suggested service to use to make the call to the transfer target, if the Number parameter is also specified • Number: Optional. The number of the transfer target to call. User aren't prompted to enter the transfer target number if this parameter is specified 	 Transfer
Blind Transfer	<p>Invoke the blind call transfer function on the currently highlighted call on the screen when the Calls App is at the top of the display stack. The call must be in a transferable state, that is, Holding or Connected State.</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Number: Optional. The number of the transfer target to transfer the call to. Users aren't prompted to enter the transfer target number if this parameter is specified 	 Blind Transfer
Blind Transfer 2	<p>Same as the Transfer function, collecting the target number from the user, but a blind transfer is performed after the target number is collected. In contrast to Blind Transfer, Blind Transfer 2 plays dial tone, applies digit map, and would time out as user enters the target number.</p>	 Blind Transfer 2
Action URL	<p>Launch an OBiPhoneXML Application at the given URL</p> <p>Parameters:</p> <ul style="list-style-type: none"> • Number: Required. The URL of the OBiPhoneXML Application to run • Name: A name to identify the application on the phone screen 	 Launch Action URL

Preassigned Feature Keys (PFKs)

The OBi Edition phones have no dedicated programmable keys. Instead, they have hard keys wired to the most commonly used features with function-specific icons printed on the physical keys. Theoretically, though not recommended, you can reprogram these hard keys to provide other functions, like the programmable keys on the phones.

The default (and recommended) functions of these hard keys are:

- Transfer (all phones)
 - Default **Function** = `Transfer`

Two valid alternatives to this behavior are to change **Function** to `Blind Transfer` or `Blind Transfer 2` when pressing the key would provide corresponding behavior, respectively.
- Messages (V VX 350 and V VX 450 only)
 - Default **Function** = `Message Status`

You should set up the **Number** and **Service** parameters of this key to access voicemail.
- Hold (all phones)
 - Default **Function** = `Hold`

Feature Key Configuration Parameters

You can configure each feature key independently. The configuration of each feature key comprises of the following set of parameters:

- **Function:** Select the function to assign to this feature key.
- **Service:** The service to bind the key to.

Required for:

- ACD Sign On or Off
- Busy Lamp Field
- Call Park Monitor
- Disposition Code
- Hoteling
- Exec Filter On or Off
- Exec Assistant
- Line Monitor
- Message Status
- Presence Monitor
- Security Class

Optional for:

- Block Anonymous Call
- Block Caller ID
- Call Appearance
- Call Forward
- Do Not Disturb
- Transfer
- Speed Dial

Not used otherwise.

- **Number:** Required for:
 - BLF

- Call Park Monitor
- Presence Monitor
- Speed Dial

Optional for:

- Transfer
- Blind Transfer
- **Name:** Optional for all functions. It's used as a nickname to refer to the entity specified in the **Number** parameter.
- **MaxCalls:** Maximum number of calls to overload on the key. This is only applicable if the function is Call Appearance. Therefore, this parameter isn't available under Programmable Keys.
- **Group:** A short name referencing the Line Key Customization Group to use to customize the layout of this Line Key.

It's very common to have multiple feature keys defined with the same function, such as **Call Appearance** and **Speed Dial**. It isn't advisable to have a particular monitor function with the same monitored entity configured on more than one key. For example, don't assign more than one BLF key to monitor the same extension or assign more than one Message Status key to monitor the same mailbox. The phone can update only one of the keys when the status of the monitored entity changes.

Highlights of Feature Key Functions

This section highlights and defines the most commonly used feature key functions.

Call Keys

A Call feature key or Call key is a Virtual Line Key (VLK) with the call function assigned. Each call carried out on the phone needs an assigned call key. That is, you need at least one call key to make or receive a call. Each call key can hold exactly one call. Usually, there are at least a few call keys defined on the phone to handle multiple call scenarios, such as call waiting and conference calls. If there's a new incoming call but no more call keys to assign the call to, the call receives the busy treatment.

You can configure a call key to a specific voice service account installed on the phone or unbound to any service. You can configure a Bound call key to handle calls on the bound service account only, while an unbound call key can handle calls on any service account.

For incoming calls, the phone automatically assigns the call to an open call key. It first attempts to find a call key that is bound to the service account where the incoming call is on. If none is found, it then tries to look for an unbound call key to assign the call to. If none is found, the call is rejected with busy treatment.

A call key can be "overloaded" with as many as 24 calls. You can configure the number of calls to overload on a call key using the **MaxCall** parameter. When the key is overloaded, the LED pattern and the key display shows the states of only one call at a time. This call is referred to as the call-in-focus. The phone automatically selects the call-in-focus. The operation triggered by pressing the call key applies to the call-in-focus only according to its current state.

Line Monitor Keys

Unlike a line key, a Line Monitor key monitors the status of a line. Pressing the key results in:

- Answering an incoming call if there's one. If there are more than one, answer the oldest one
- Starting the dial tone, if there is spare capacity

Speed Dial Keys

The Speed Dial function lets users configure a speed dial number from the phone UI. Users can press and hold the **Speed Dial** key until the feature key item displays on the screen. From there, users can input speed dial details. You can configure the service to use for calling with the speed dial. This function supports the Push-to-Talk (PTT) option.

In addition to the speed dial feature keys, there are 99 configurable speed dial numbers that users can access by dialing the corresponding 1-2-digit speed dial code (that is, 1 – 99). These 99 speed dial storages are referred to as the “Speed Dials 99” feature that is entirely independent of the speed dial feature keys discussed here.

Busy Lamp Field Keys

Users can press a BLF key to monitor the call status of another extension. This feature operates in the context of an SP service. In many cases, the BLF key also acts as a speed dial key to call or transfer a call to the monitored extension. This key also supports PTT when it's used to call the monitored extension. Note that the call or call transfer to the monitored extension uses the same underlying SP service.

Call Park Monitor Keys

Call Park Monitor (CPM) is very similar to BLF, except each key monitors a call park orbit instead of another user extension. It only indicates the status whether a call is parked on the orbit or not. Press the key once to park the highlighted call or retrieve the call from the orbit.

Presence Monitor

This function is used together with a Buddy List. You can configure a feature key to monitor the presence/status of a buddy in a buddy list. You can also use this key as a speed dial to that buddy.

The Buddy List feature operates in the context of an SP service. Pressing the Presence Monitor key to call a buddy uses the same underlying SP service to make the call. The PTT option is not available with this key when calling the buddy.

Group Page Keys

The phone supports two multicast page groups called Page Group 1 and Page Group 2. Set a feature key with the Page Group 1 or Page Group 2 function to use the respective page group. Paging is one way. The phone doesn't play incoming audio when the user is talking. The configuration of each page group has a Push-To-Talk option that you can enable to require users to press the **Page** key to talk. The **Page** key also lets the user join or leave the group with one key press. For example when you don't want to receive an incoming page, you can temporarily leave the group. The LED color also reflects the current group-joining status as a reminder to the user.

Line Keys and Virtual Line Keys

There are twelve (12) physical Line Keys (LKs) on the VVX 450, six (6) on the VVX 350, four (4) on the VVX 250, and two (2) on the VVX 150. On the VVX 450, the line keys are split half and half to the left and right of the LCD display. On the VVX 350, VVX 250, and VVX 150, the line keys are on the left side of the display.

By means of multiple “tabs” or pages on the VVX 450, VVX 350, and VVX 250, one single physical line key becomes multiple Virtual Line Keys (VLKs), one VLK per physical line key per tab. Tabs display across the top left and right of the screen, with the active tab highlighted. The VVX 150 doesn’t support virtual line keys.

These models support four tabs, providing a total of 48 VLKs on the VVX 450, 24 on the VVX 350, and 16 on the VVX 250. The VVX 150 doesn’t support tabs.

On the VVX 450, the VLKs are further listed and numbered onscreen as **Left Line Keys (LLK)** for the line keys on the left side of the display, and simply **Line Keys (LK)** for the line keys on the right side of the display. On the VVX 250 and VVX 350, which have only one row of physical line keys on the left, the VLKs are all labeled **Left Line Keys (LLK)**.

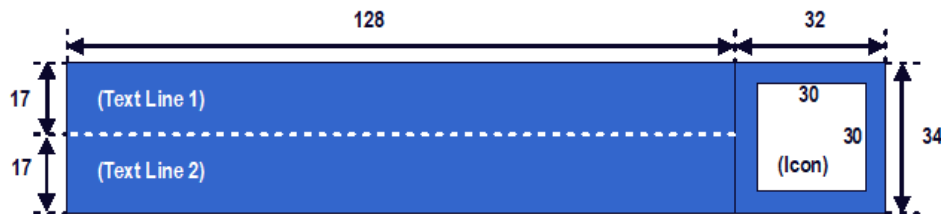
The virtual line keys are numbered and grouped into tabs as follows:

Virtual Line Key Labels on VVX 250, VVX 350, and VVX 450

VLK Groups within Tabs	Tab 1	Tab 2	Tab 3	Tab 4
VVX 250 4 LKs = 16 VLKS	LLK1 – LLK4	LLK5 – LLK8	LLK9 – LLK12	LLK13 – LLK16
VVX 350 6 LKs = 24 VLKS	LLK1 – LLK6	LLK7 – LLK12	LLK13 – LLK18	LLK19 – LLK24
VVX 450 12 LKs = 48 VLKS (2 groups of 6 LKs)	LK1 – LK6 LLK1 – LLK6	LK7 – LK12 LLK7 – LLK12	LK13 – LK18 LLK13 – LLK18	LK19 – LK24 LLK19 – LLK24

Line Key Window Customization

Like Key Window dimensions



A Line Key Window (also known as Screen Tile) is a 160Wx34H-pixel area next to each Line Key on the screen. As shown in the picture above, this window is divided into three nonoverlapping regions: Text Line 1, Text Line 2, and Icon. The information shown in the window depends on the function assigned to the corresponding line key (which is a feature key) and the current state of the assigned function.

You can customize the information shown on Text Line 1 and 2 for the following feature key functions with the parameters in the following table:

- Call Appearance
- Speed Dial
- Busy Lamp Field
- Action URL

Line Key Window Customization Parameters

Parameter Group	Parameter	Description
Call Appearance		
<i>IP Phone – Feature Key Customization – Call Appearance</i>	Enable	Enables using customized display for Call Appearance feature keys.
<i>IP Phone – Feature Key Customization – Call Appearance</i>	TextLine1	Textual information to show on Text Line 1 area during idle.
<i>IP Phone – Feature Key Customization – Call Appearance</i>	TextLine2	Textual information to show on Text Line 2 area during idle.
<i>IP Phone – Feature Key Customization – Call Appearance</i>	TextLine1InCall	Textual information to show on Text Line 1 area during a call.
<i>IP Phone – Feature Key Customization – Call Appearance</i>	TextLine2InCall	Textual information to show on Text Line 2 area during a call.
Busy Lamp Field		
<i>IP Phone – Feature Key Customization – Busy Lamp Field</i>	Enable	Enables using customized display for Busy Lamp Field feature keys.
<i>IP Phone – Feature Key Customization – Busy Lamp Field</i>	TextLine1	Textual information to show on Text Line 1 area.
<i>IP Phone – Feature Key Customization – Busy Lamp Field</i>	TextLine2	Textual information to show on Text Line 2 area.
<i>IP Phone – Feature Key Customization – Call Appearance</i>	TextLine1InCall	Textual information to show on Text Line 1 area when the monitored extension has one or more calls.
<i>IP Phone – Feature Key Customization – Call Appearance</i>	TextLine2InCall	Textual information to show on Text Line 2 area when the monitored extension has one or more calls.

Speed Dial

Line Key Window Customization Parameters

Parameter Group	Parameter	Description
<i>IP Phone – Feature Key Customization – Speed Dial</i>	Enable	Enables using customized display for Speed Dial feature keys.
<i>IP Phone – Feature Key Customization – Speed Dial</i>	TextLine1	Textual information to show on Text Line 1 area.
<i>IP Phone – Feature Key Customization – Speed Dial</i>	TextLine2	Textual information to show on Text Line 2 area.
Action URL		
<i>IP Phone – Feature Key Customization – Action URL</i>	Enable	Enables using customized display for Action URL feature keys.
<i>IP Phone – Feature Key Customization – Action URL</i>	TextLine1	Textual information to show on Text Line 1 area.
<i>IP Phone – Feature Key Customization – Action URL</i>	TextLine2	Textual information to show on Text Line 2 area.
Group n ($n = 1, 2, 3, 4$)		
<i>IP Phone – Feature Key Customization – Group n</i>	ID	The ID to reference this Group.
<i>IP Phone – Feature Key Customization – Group n</i>	TextLine1	Textual information to show on Text Line 1 area.
<i>IP Phone – Feature Key Customization – Group n</i>	TextLine2	Textual information to show on Text Line 2 area.
<i>IP Phone – Feature Key Customization – Group n</i>	Icon	Internal icon path or HTTP/HTTPS URL.

You can use macros to specify the textual contents for Text Line 1 and Text Line 2 areas. You must precede a macro with a \$, such as \$number or \$(number). The enclosing parenthesis is required when a character that is a legal macro character follows the macro. A legal macro character is one of {a-z A-Z . _ 0-9}.

The following macros are available.

Macros for Line Key Window Customizations

Macro	Description	Notes
\$name	The Name field as configured for the feature key.	
\$number	The Number field as configured for the feature key.	

Macros for Line Key Window Customizations

Macro	Description	Notes
<code>\$service</code>	The default name of the service specified in the Service field for the feature key: SP1, ..., SP6, OBiTALK.	Expands into the word <code>Auto</code> if service field isn't specified.
<code>\$service.displayLabel</code>	The X_DisplayLabel parameter per SP n service or the DisplayLabel parameter per OBiTALK service.	{Blank} if service field isn't specified.
<code>\$service.number</code>	For SP1 – SP6, this is the User ID of the SIP account, which is derived from the AuthUserName or URI parameter of the SP service. For OBiTALK, this is the OBi number of the phone.	{Blank} if service field isn't specified.
<code>\$service.displayNumber</code>	The X_DisplayNumber parameter per SP n service or the DisplayNumber parameter per OBiTALK service.	{Blank} if service field isn't specified.
<code>\$func</code>	Short internal name of the function assigned to the key: <ul style="list-style-type: none"> • <code>call</code> for Call Appearance • <code>blf</code> for Busy Lamp Field • <code>spd</code> for Speed Dial • <code>acturl</code> for Action URL 	{Blank} if function isn't specified.
<code>\$funcname</code>	Long internal name of the function assigned to the key: <ul style="list-style-type: none"> • <code>Call</code> for Call Appearance • <code>Busy Lamp</code> for Busy Lamp Field • <code>Speed Dial</code> for Speed Dial • <code>XML APP</code> for Action URL 	{Blank} if function isn't specified.
<code>\$call.service</code> and <code>\$call.service.x</code> where $x = \text{number, displayLabel, or displayNumber}$	Same as <code>\$service</code> and <code>\$service.x</code> applying to the service the call is on.	{Blank} if service isn't defined for the call yet.
<code>\$call.number</code>	Call peer's Number	
<code>\$call.name</code>	Call peer's Name	
<code>\$value</code>	For BLF, this is number of calls currently active on the monitored entity. The value is {blank} for other functions.	
<code>\$blf.ext</code>	The extension of the entity monitored by the feature key.	
<code>@id</code>	Line Key Hard Key ID. Expands into: <ul style="list-style-type: none"> • 1,2 for VVX 150, VVX 250 • 1,2,3 for VVX 350, VVX 450 	

Macros for Line Key Window Customizations

Macro	Description	Notes
@p	Line Key Page. Expands into: <ul style="list-style-type: none"> • 1 for VVX 150, VVX 250 • 1,2,3,4 for VVX 350, VVX 450 	
@n	Line Key ID. Expands into: <ul style="list-style-type: none"> • 1,2 for VVX 150 • 1 – 16 for VVX 250 • 1 – 24 for VVX 350 • 1 – 48 for VVX 450 	
\$eval	See the Polycom® OBiPhoneXML Applications API Guide .	
@text1	The default information displayed on the (Text Line 1) section of the Line Key Window. Applicable in the TextLine1 , TextLine2 , TextLine1InCall , and TextLine2InCall parameters	
@text2	The default information displayed on the (Text Line 2) section of the Line Key Window. Applicable in the TextLine1 , TextLine2 , TextLine1InCall , and TextLine2InCall parameters	
@icon	The default icon displayed on the (Icon) section of the Line Key Window. Applicable to the Icon parameter.	

Example

In this example, the information that normally displays on line 1 and line 2 of all Call Appearance Line Keys is reversed. In addition, during a call, "--" is added to the beginning of line 1. Below are the settings for this example.

Call Appearance Line Key Settings

Parameter Group	Parameter	Value
Call Appearance		
IP Phone – Feature Key Customization – Call Appearance	Enable	true (that is, the option is checked)
IP Phone – Feature Key Customization – Call Appearance	TextLine1	@text2
IP Phone – Feature Key Customization – Call Appearance	TextLine2	@text1
IP Phone – Feature Key Customization – Call Appearance	TextLine1InCall	-- @text2

Call Appearance Line Key Settings

Parameter Group	Parameter	Value
<i>IP Phone – Feature Key Customization – Call Appearance</i>	TextLine2InCall	@text1

Line Key Tabs

By default, the line keys area to the right of the screen has four tabs of VLKs, and only one tab is visible at any time. You can disable this feature using the option **User Preferences::LineKeyTabs**. When disabled, the phone shows no line key tabs in the status bar and only one page of VLKs is available (none on the VVX 150, VLK 1-4 on the VVX 250, and VLK 1-6 on the VVX 350 and VVX 450). The status icons also display in the right corner. To handle more calls in this case, you may enable multiple calls per Call (Appearance) Key as described in the next section.

Calls App Behavior

The Calls App shows a list of current calls on the phone. Calls controlled under the same call key are grouped together in the list with calls on call keys with a smaller VLK index shown first. New calls are added to the beginning of the call group of the same call key.

The phone can only highlight one call on the Calls App screen, and the corresponding VLKW shows a white bounding box to identify the call key the highlighted call belongs to. As a user navigates through the calls on the screen, the white bounding box moves accordingly.

Normally, the calls display together on the Calls App screen. Hence it's possible to see two calls belonging to the same call key on the same screen. If this isn't desirable, you can disable the option **User Preferences::PackCallsOnDisplay**, which causes the phone to display only the calls belonging to the same call key.

IP Phone Settings

Phone Settings

Settings are divided into the following groups:

- Phone Settings
- Line Keys
- Programmable Keys

DigitMap and OutboundCallRoute

The **DigitMap** controls what number you can dial and applies the given transformation to the dialed number. It can refer to the **DigitMap** parameter values in other parameter groups for better readability and organization. The **OutboundCallRoute** determines which service to use based on the dialed number, after validation and transformation by the **DigitMap**.

Primary Line

You can select the Primary Line for the Phone and for the AA, respectively, using the **Phone Settings::PrimaryLine** and **Auto Attendant::PrimaryLine** parameters. The primary line is the default line to use when there is no explicitly selected line and no line-selection prefix (line access code) has been dialed. For example, when going off-hook to get Dialtone, the phone tries to allocate a call key that is bound to the primary line for the call, if one is available.

You can make one of the SP Services, OBiTALK or TG1/TG2 as the Primary Line for outbound calls. The Primary Line for the Phone and the Auto Attendant is configured separately. The following list summarizes the choices available for selection as the primary line:

- SP1 Service
- SP2 Service
- SP3 Service
- SP4 Service
- SP5 Service
- SP6 Service
- OBiTALK Service
- Trunk Group 1
- Trunk Group 2

You can select a specific Line to use when making a call explicitly by pressing a call key or line monitor key bound to that line or a softkey corresponding to that line. You can also dial a Line's access code before the destination number. The default service route access codes are defined as:

- ** 1 for SP1
- ** 2 for SP2
- ** 3 for SP3
- ** 4 for SP4
- ** 9 for OBiTALK

Service route access codes for calling from the Phone can be customized if necessary by modifying **Phone Settings::DigitMap** and **Phone Settings::OutboundCallRoute**. Service route access codes for calling via the Auto Attendant can be customized if necessary by modifying **Auto Attendant::DigitMap** and **Auto Attendant::OutboundCallRoute**.



The phone handles the **PrimaryLine** setting by substituting internally all occurrences of `pli` with the abbreviated name of the trunk named as the primary line in the **DigitMap** and **OutboundCallRoute** parameters of the same parameter group.

Network Directory

The **Enable** option in this group is for the enabling and displaying the Network Directory option on the Main menu. The **VoiceService** option determines which SP service's network directory function to invoke when the Main menu option is selected by the user.



Only one Network Directory option can be shown on the Main menu. The **Name** parameter is reserved for future use.

Buddy List

The **Enable** option in this group is for the enabling and displaying of the Buddy List option on the Main menu. The **VoiceService** option determines which SP service's buddy list function to invoke when the Main menu option is selected by the user.



Only one Buddy List option can be shown on the Main menu. The **Name** parameter is reserved for future use.

User Preferences Settings

The **CallForwardUnconditionalFeatureProvider** determines which **CallForwardUnconditional** parameter the Call Forward option under User Preferences should control. Note that this is the same setting the **Call Forward** softkey (in the Home Screen) controls. There is the Phone version and also one version for each voice service. If an *SPn* service is selected and if the **SPn – Network Provided Services::CallFowardAlways** is `true`, then the user preferences and softkey option reflects and controls the setting of the feature at the server side.

The **DoNotDisturbFeatureProvider** determines which **DoNotDisturb** parameter the Do Not Disturb option under User Preferences should control. There is the Phone version and also one version for each voice service. If an *SPn* service is selected and if the **SPn – Network Provided Services::DoNotDisturb** is `true`, then the user preferences option reflects and controls the setting of the feature at the server side.

Page Groups 1 and 2

GroupName is a nickname to refer to the page group; not used anywhere at the moment.

MulticastAddress and **MulticastPort** define the multicast address of the group to join, and **TTL** sets the TTL value of the outgoing multicast packets. **ParticipantName** is a name to identify to the group the user of this phone via RTCP messages.

To use a page group effectively, there must be a feature key assigned with the corresponding page group function. You can press the key once to talk to the group, or use PTT to talk if the **PushToTalk** option is also enabled in the page group configuration.

Line Keys

This group is used for the configuration of the Line Keys as Feature Keys.

There are twelve (12) physical Line Keys (LKs) on the VVX 450, six (6) on the VVX 350, four (4) on the VVX 250, and two (2) on the VVX 150. On the VVX 450, the line keys are split half and half to the left and right of the LCD display. On the VVX 350, VVX 250, and VVX 150, the line keys are on the left side of the display.

Network

SIP Registration

You can set up your phones to periodically register with the **ProxyServer** or the **RegistrarServer** by enabling the **SPn Service::X_RegisterEnable** parameter. **ProxyServer** and **RegistrarServer** could be different, although they're rarely so in practice. **ProxyServer** is a required parameter that you must configure on your phone while **RegistrarServer** is optional and assumed to be the same as the **ProxyServer** if not specified in the configuration.



If the server isn't listening at the standard port, configure the correct port value in **ProxyServerPort** (and **RegistrarServerPort** as needed).

The main purpose of registration is to create and maintain a dynamic binding of the SIP/SP account to your phone's local contact address. Service providers can also rely on this periodic message to see if your phone is online and functional. Each phone takes only one local IP address that is either statically assigned in the phone's configuration or dynamically obtained from a local DHCP server. The method to get an IP address assigned is determined by the value of **WAN Settings::AddressingType**.

The **SPn** services for $n = 1 - 6$ each use a different local contact port for sending and receiving SIP messages (the default is 5060, 5061, 5062, ..., and 5065 respectively). You can configure this port in the **SPn Service::X_UserAgentPort** parameter. **ProxyServer** and **RegistrarServer** must use the same transport protocol for SIP messages that you can set in the **ProxyServerTransport** parameter.

Your phones support UDP, TCP, and TLS for SIP transport. The default server port is 5060 for UDP/TCP and 5061 for TLS. When using TCP/TLS, the phones initiate a TCP/TLS connection only with the **ProxyServer**. All subsequent SIP messages are exchanged between the phones, and the servers must use the same connection. If for any reason the connection is closed, your phone attempts to re-establish the connection following an exponential back-off retry pattern.



Dynamic address binding through periodic registration isn't strictly necessary if the local IP address of your phone doesn't change. You can statically configure your phone's contact address on the Registration Server. Also, your network administrator may be able to reconfigure the duration of your DHCP lease so that your DHCP address effectively doesn't change.

Here is a typical REGISTER request generated by your phone:

```
REGISTER sip:as.xyz.broadworks.net:5060 SIP/2.0
Call-ID: 7107d244@192.168.15.207
Content-Length: 0
CSeq: 10722 REGISTER
From: <sip:3134445567@as.xyz.broadworks.net>;tag=SP337b73f3bf7a504c3
```

```
Max-Forwards: 70
To: <sip:2404982564@as.xyz.broadworks.net>
Via: SIP/2.0/UDP 192.168.15.207:5062;branch=z9hG4bK-f9e9e56c;rport
User-Agent: OBIHAI/OBi1062-5.0.0.1542
Contact: <sip:2404982564@192.168.15.207:5062>;expires=60;
  +sip.instance="<urn:uuid:00000000-0000-0000-0000-9abcde700065>"
Allow: ACK,BYE,CANCEL,INFO,INVITE,NOTIFY,OPTIONS,PRACK,REFER,UPDATE
Supported: replaces, eventlist, record-aware
```

In this example, your phone doesn't use the Expires header in REGISTER requests. Instead, the Expires value (in seconds) is encoded as a parameter in the Contact header. The two methods are equivalent in this usage per RFC3261. Note that your phone also includes the +sip-instance parameter in the Contact header that specifies your phone's MAC address in the UUID. You can suppress this parameter by disabling the **ITSP Profile X – SIP:X_RegisterIncludeInstance** option.

In some cases, your phone may not receive any response to its REGISTER from the server if an upstream router blocks the outgoing message sent by it. To cope with such a case, tell your phone to try other SIP user agent ports by specifying a comma-separated list of as many as 10 alternative ports in the **SPn Service:X_UserAgentPorts** parameter. Your phone then cycles through those ports to retry REGISTER until it receives a response from the server.

Third-Party Registration

You can configure your phone to use a third-party registration that registers it for an Address of Record (AOR) that isn't the same as the account user-id. That is, the user-id in the TO header of the SIP REGISTER request is different from that in the FROM header, which always carries the account user-id.

One application is in the implementation of a shared line using the Bridged Line Appearance (BLR) method. To enable third-party registration, set the user-id to register for in the **SPx Service – SIP Credentials:X_ShareLineUserID** parameter.

Registration Period

You can configure the nominal registration Expires header value (implemented as a Contact header parameter value in seconds) used by your phone in REGISTER requests using the **ITSP Profile X – SIP:RegistrationPeriod** parameter. The server determines the actual Expires value. The server may reject the REGISTER request with 423 with a Min-Expires header value (in seconds). If that happens, your phone quickly retries with an Expires header value equal to the Min-Expires header value from the server. When the server accepts the registration, it replies with a 2xx response for the REGISTER and includes an Expires parameter value in the Contact header that matches the Contact your phone uses in the REGISTER request. However, if it isn't found in the Contact, your phone takes the server-supplied Expires value from the Expires header of the 2xx response. If still not found, your phone assumes the server-supplied value is 3600 seconds.

If the server-supplied Expires value is less than the Expires header value used by your phone, it takes the server version to compute the next renewal interval. Otherwise, your phone uses its own Expires header value to do the same.



The server shouldn't supply explicitly or implicitly an Expires value that is larger than what your phone has asked for, as that would be a protocol violation. Your phone, however, ignores such an error.

Your phone computes the next renewal time by subtracting a percentage of the Expires value derived from the 2xx response returned by the server. Use the *ITSP Profile – SIP:X_RegistrationMargin* parameter to control how the margin to subtract is computed. For example: If **X_RegistrationMargin** is 0 or not specified, the renewal time is half-way before the expiration, if the Expires value is less than 1200s, or 600s before the expiration otherwise. If **X_RegistrationMargin** ≥ 1 , it's interpreted as the number of seconds (with any fractional part dropped) before the expiration time to renew registration. If **X_RegistrationMargin** < 1 , it's interpreted as the fraction of the current expires value to subtract from the expiration time to get the time for the next renewal. For example, if **X_RegistrationMargin** = 0.01 and the Expires value is 300, then the next renewal goes out 3 seconds before the expiration time.

REGISTER Final Non-2xx Response Handling

When registration encounters an error, your phone can schedule retries based on the type of error. Each recognizable error type is represented by a 3-digit code. Error codes 300 – 699 are the error response codes returned by the server, while 900-999 are used to indicate other errors. The following 9xx error codes are related to registration:

- 900 = Timeout waiting for a response from the server
- 901 = Cannot resolve the server name or the host is not reachable

For 3xx class responses with a valid Contact header, your phone follows the given Contact to retry registration quickly. If a valid Contact is not found, or if the number of consecutive redirections has reached 5, your phone considers the 300 response an error and performs standard error handling.

For 401 and 407 responses with a valid Proxy-Authenticate or WWW-Authenticate header, your phone retries registration quickly and includes the properly computed Proxy-Authorization or Authorization header. However, if the error response contains no valid Proxy-Authenticate or WWW-Authenticate header, and if the number of consecutive 401/407 responses received has reached 2, your phone considers the 401/407 a true error and performs standard error handling.

For 423 responses with a valid Min-Expires header value, your phone retries registration quickly with a new Expires value that conforms to the Min-Expires value from the server. However, if the Min-Expires header is not present in the response or the value is not larger than the current Expires value sent by your phone, your phone considers it an error and performs standard error handling.

For 5xx – 6xx responses with a Retry-After header, your phone schedules a retry after the specified value.

The standard handling is by waiting for a certain number of seconds before trying to register again. The number of seconds to wait is determined by the rules specified in the *ITSP Profile X – SIP:RegisterRetryResponseCodes* parameter on the actual error code. The format of this parameter is the same as a digit map. Let's consider the default value of this parameter:

```
(<40[17]:w120>|<40[34]:w120>|<99[01]:w120-200>|[4-9]xx)
```

Each rule is a substitution where a certain error code or error code pattern is mapped to the number of seconds to wait. With this example, your phone waits for 120 seconds for 401 and 407 error codes, 120 seconds for 403 and 404 error codes, randomly between 120 and 200 seconds for 990 and 991 error codes, and a fixed default value for all other error codes. The fixed default value is configured in the *ITSP Profile X – SIP:RegisterRetryInterval* parameter. The syntax $w\{a\}-\{b\}$ specifies a random range of between $\{a\}$

seconds and {b} seconds. Error codes not covered by these rules cause your phone not to retry registration after the error.

SIP Outbound Proxy Server

An **OutboundProxy** server can be configured on your phone such that all outbound requests are sent via the outbound proxy server instead of directly to the **ProxyServer** or **RegistrarServer**. If the outbound proxy server is listening at a non-standard port, the correct port value must be specified in the **OutboundProxyPort** parameter. The **OutboundProxy** may use a different transport protocol from the **ProxyServer**. The transport protocol to use to communicate with the **OutboundProxy** can be set in the **OutboundProxyTransport** parameters. If **OutboundProxyTransport** is TCP or TLS, your phone initiates a TCP or TLS connection only with the **OutboundProxy**. All subsequent messages exchanged between your phone and the servers **MUST** use the same connection. If for any reason the connection is closed, your phone attempts to re-establish the connection with the **OutboundProxy** following an exponential back-off retry pattern.

Even though your phone only exchanges messages directly with the **OutboundProxy**, the **ProxyServer**, **ProxyServerPort**, and **ProxyServerTransport** parameters are still very much relevant and important since the SIP requests sent by your phone to the server are still formed based on these values, not based on the **OutboundProxy** value. In fact, the **OutboundProxy** value should never appear in the SIP requests generated by your phone (unless the **OutboundProxy** has the same value as the **ProxyServer**).

Some server implementations include the outbound proxy server in a Record-Route header such that your phone should not respect the locally configured **OutboundProxy** value after the initial INVITE is sent for a new call. This behavior can be achieved by enabling the option **ITSP Profile X – SIP:X_BypassOutboundProxyInCall**. However, this option has no effect when the **OutboundProxyTransport** is TCP or TLS, as your phone always uses the same connection to send messages to the server.

DNS Lookup of SIP Servers

When sending out SIP requests to the server, your phone looks up the IP address of the server, using DNS query if the server is specified as a domain name instead of an IP address. If an Outbound Proxy Server is configured, it is used instead of the SIP Proxy Server or SIP Registration Server. The resolution of the server domain name into IP address is performed in the following manner:

- If **ITSP Profile X – SIP:X_DnsSrvAutoPrefix** is enabled, resolve the name as DNS A Record, DNS SRV Record and as DNS SRV record with a service prefix prepended to the name in parallel by sending 3 queries to each DNS server at the same time. The service prefix to prepend to the name depends on the transport protocol being used; for SIP, `_sip._udp.` for UDP, `_sip._tcp.` for TCP and `_sip._tls.` for TLS. If more than one valid result is returned from the queries, the DNS SRV result for the name with prefix has the highest priority, then the DNS SRV result for the name without the prefix, then the DNS A result.
- Otherwise, resolve the name as a DNS A record and as a DNS SRV Record in parallel by sending 2 queries to each DNS server. If both queries return a valid result, the DNS SRV result is taken over the DNS A result.
- If no valid results are returned, your phone considers the SIP request failed with the error code 901.

If the result from the DNS query is an SRV record, the server port is also taken from that record (the server port value configured on your phone is ignored). Otherwise, the server port is taken from the configured

value. If no value is specified, 5060 is used. We recommend setting the **ProxyServerPort** to 0 (that is, the unspecified value) if DNS SRV lookup is intended for the service.

NAT Traversal Considerations

If your phone is located behind NAT with respect to the service provider equipment, it can discover the mapped external address corresponding to its local SIP contact address as seen by the server. This may help in some cases where a gateway router's SIP ALG implementation is causing communication problems between your phone and the server. Your phone can discover the mapped external address in one of the following ways:

- From the “received=” and “rport=” parameters of the VIA header of the REGISTER response sent by the server. These two parameters tell your phone its mapped IP address and port number, respectively. This method is used if periodic registration is enabled on your phone.
- From the response to a STUN binding request your phone sent to a STUN server. This method is used by enabling **X_KeepAliveEnable** and setting the **X_KeepAliveMsgType** parameter to “stun”. In this case, the STUN server is taken from the **X_KeepAliveServer** parameter, if it is specified. Otherwise, the keep-alive messages are sent to the same server where a REGISTER request would be sent to. The latter is the most effective way of using STUN to discover the mapped external contact address.
- From the value of the **ITSP Profile X – SIP::X_PublicIPAddress** parameter.

If discovered by one of the methods above, your phone uses the discovered external IP address and port to replace its private address and port when generating SIP requests when the **ITSP Profile X – SIP::X_DiscoverPublicAddress** option is also enabled. The substitution of private addresses with public addresses applies to the Contact header of any SIP requests and the `c=` line in SDP. If the option **X_UsePublicIPAddressInVia** is also enabled, the Via address is also substituted. However, this usually isn't necessary.

Your phone can also include an empty `rport` parameter in the Via header of outbound SIP requests if the option **ITSP Profile X – SIP::X_UseRport** option is enabled. This parameter is sometimes needed to prompt the server to insert an `rport` parameter value in the response. It should also prompt the server to send the response to the port where the request originated from (that is, according to the source port of the IP header of the packet). However, as such behavior on the server is considered standard by many, the empty `rport` parameter has become superfluous in practice.

Keep Alive Messages

In addition to periodic registration with the server, your phone can be instructed to send out periodic keep alive messages on the same network path to keep the NAT pinhole open. For this purpose, it is recommended the keep alive messages are sent to the same proxy server responsible for registration. The keep alive messages may be dropped by the server unprocessed. However if STUN binding request are used as keep alive messages, it is recommended that the server return a valid STUN binding response to each request. The parameters that control the sending of keep alive messages are:

Keep Alive Messages

Parameter Group	Parameter	Description
<i>SPn Service</i>	X_KeepAliveEnable	Enable sending of keep alive message. If set to <code>true</code> , your phone sends periodic keep-alive messages to the destination specified in X_KeepAliveServer and X_KeepAliveServerPort , at the interval specified in X_KeepAliveExpires . The content of this message is the ASCII string <code>keep-alive\r\n</code>
<i>SPn Service</i>	X_KeepAliveExpires	Keep alive period in seconds.
<i>SPn Service</i>	X_KeepAliveMsgType	The type of keep alive messages to send out periodically if <code>keep-alive</code> is enabled. One of the following: <ul style="list-style-type: none"> <code>keep-alive</code>: The string <code>keep-alive</code> for SIP transport over UDP, or CRLF (e.g., carriage return and line feed characters) for SIP transport over TCP or TLS. <code>empty</code>: A blank line <code>stun</code>: A standard STUN binding request. Your phone uses the binding response to form its contact address for REGISTRATION. <code>custom</code>: use the value of X_CustomKeepAliveMsg <code>options</code>: a valid SIP OPTIONS request. No response to this request from the server triggers a proxy failover if proxy redundancy is enabled. <code>notify</code>: a valid SIP NOTIFY request. No response to this request from the server triggers a proxy failover if proxy redundancy is enabled.
<i>SPn Service</i>	X_CustomKeepAliveMsg	Defines the custom message to be used when X_KeepAliveMsgType is <code>custom</code> . The value should have the following format: <code>mtd=NOTIFY;event={whatever};user={anyone}</code> Where <ul style="list-style-type: none"> <code>NOTIFY</code> may be replaced by any other SIP method, such as <code>PING</code>, the <code>event</code> parameter is optional and is only applicable if the method is <code>NOTIFY</code>. If <code>event</code> is not specified, the 'keep-alive' event is used with <code>NOTIFY</code>. The <code>user</code> parameter is optional. If not specified, the request-uri won't have a <code>userid</code>, and the <code>TO</code> header field uses the same <code>userid</code> as the <code>FROM</code> header, which is the local account <code>userid</code>. If <code>user</code> is specified, it is used as the <code>userid</code> in the Request-URI and <code>TO</code> header. SIP messages for <code>keep-alive</code> are sent only once without retransmission. Response to the SIP messages are ignored by the phone.

SIP Proxy Server Redundancy and Dual REGISTRATION

Server Redundancy specifically refers to your phone's capability to:

- Look for a working server to REGISTER with from a list of candidates
- Switch to another server once the server that it currently registers with becomes unresponsive

As such, device registration must be enabled to use the server redundancy feature. Other SIP requests, such as INVITE or SUBSCRIBE, are sent to the same server that your phone currently registers with.

If the Outbound Proxy Server is provided, server redundancy is applied to the Outbound Proxy Server instead of the REGISTRATION server. Server redundancy behavior is enabled by enabling the ***ITSP Profile X – SIP::X_ProxyServerRedundancy*** parameter (which is disabled by default).

Another requirement for using the server redundancy feature is that the underlying server must be configured in your phone as a domain name instead of an IP address. This allows your phone to collect a list of candidate servers based on DNS query. The domain name may be looked up as DNS A record or DNS SRV record. For A records, all the IP addresses returned by the DNS server are considered to have the same priority. For SRV records, the hosts returned by the DNS server can be each assigned a different priority.

After a list of candidate servers is obtained, your phone first looks for a working server according to the stated priority. A working server means one that your phone can successfully register with. This is known as the Primary Server. Subsequently, your phone maintains registration with the primary server the usual way. However, if no working server is found after traversing the entire list, your phone takes a short break and repeats the search in the same order.

While maintaining registration with the Primary Server, your phone continually attempts to fall back to one of the candidate servers that has higher priority than the primary server, if any. The list of candidate servers that your phone is trying to fall back on is known as the primary fallback list, which may be empty.

In addition, your phone can be configured to maintain a secondary registration with a server that has lower or equal priority than the primary server. Secondary registration can be enabled by setting the ***X_SecondaryRegistration*** parameter to `true`. If ***X_ProxyServerRedundancy*** is `false`, however, ***X_SecondaryRegistration*** does not have any effect. If this feature is enabled, as soon as a primary server is found, your phone searches for a working secondary server in the same manner from the list of candidate servers that are of lower or equal priority than the primary server. Similarly, once a secondary server is found, your phone forms a secondary fallback list to continually attempt to fall back on if the list is not empty.

The interval for checking the primary fallback list and the secondary fallback list are configured in the ***X_CheckPrimaryFallbackInterval*** and ***X_CheckSecondaryFallbackInterval*** parameters, respectively. These parameters are specified in seconds and the default value is 60 for both.



- Secondary server exists implies primary server exists.
- If the secondary server exists, it immediately becomes the primary server when the current primary server fails. Your phone then starts searching for a new secondary server if the candidate set is not empty.
- The candidate list may change (lengthened, shortened, priority changed, and so forth) on every DNS renewal (based on the entry's TTL). Your phone rearranges the primary and secondary servers and fallback lists accordingly, whichever is applicable.

If the server redundancy feature is disabled, your phone resolves only one IP address from the server's domain name and won't attempt to try other IP addresses if the server is not responding.

DNS NAPTR Support

With DNS Name Authority Pointer (NAPTR) lookup, the phone can discover the hosts to access SIP service for a given domain, and the SIP transport, preference, and order of each host, and what types DNS records to use for each listed host (SRV or A record). This feature complies with RFC 2915.

This list describes key facts of the NAPTR feature:

- NAPTR can be enabled for each ITSP Profile.
- If enabled, the phone only attempts NAPTR lookup of the domain name specified in the **OutboundProxy** parameter if configured, or otherwise the **ProxyServer** parameter (on the corresponding *ITSP Profile x – SIP* web page). It will not fall back to perform regular DNS SRV or DNS A record lookup if NAPTR records are not available.
- NAPTR lookup can be used whether the **X_ProxyServerRedundancy** parameter is enabled or not, although it would be more common to have the **X_ProxyServerRedundancy** parameter enabled with NAPTR lookup.
- If a NAPTR record indicates the host name is to be looked up as an SRV record, the phone considers the host name is complete and doesn't add any prefix to it when querying the DNS server for its SRV record, regardless the settings of the **X_DnsSrv** and **X_DnsSrvAutoPrefix** parameters.
- When more than one of UDP, TCP, or TLS are available and all are at the same order and preference, the phone considers UDP the highest priority, then TCP, then TLS.
- With NAPTR enabled, the phone can failover/fallback among UDP/TCP/TLS transports, or change to a different transport based on the result of each NAPTR lookup (which is renewed at half of TTL value from the DNS server).

SIP Privacy

Your phone observes inbound caller privacy and decodes the caller's name and number from SIP INVITE requests by checking the `FROM`, `P-Asserted-Identity` (PAID for short), and `Remote-Party-ID` (RPID for short) message headers. All these headers may carry the caller's name and number information.

If PAID is present, your phone takes the name and number from it. Otherwise, it takes name and number from RPID if it is present, or from the `FROM` header otherwise. RPID, if present, includes the privacy setting desired by the caller. The privacy setting may indicate one of the following options:

- `off` = No privacy requested. Your phone shows name and number.
- `full` = Full privacy requested. Your phone hides both name and number.
- `name` = Name privacy requested. Your phone shows the number but hides the name.
- `uri` = URI privacy requested. Your phone shows the name but hides the number.

Regardless, if PAID exists or not, your phone always takes the privacy setting from the RPID if it is present in the INVITE request.



If the resulting caller name is “Anonymous” (case-insensitive), your phone treats it as if the caller is requesting full privacy.

For outbound calls, the caller’s preferred privacy setting can be stated by phone in an `RPID` header of the outbound `INVITE` request. To enable this behavior, the **ITSP Profile X – SIP::X_InsertRemotePartyID** parameter must be set to `true`, which is the default value of this parameter. Your phone supports only two outbound caller privacy settings: `privacy=off` or `privacy=full`. The `RPID` header generated by your phone carries the same name and number as the `FROM` header. If outbound caller-ID is blocked, phone sets `privacy=full` in `RPID` and also sets the display name in the `FROM` and `RPID` headers to “Anonymous” for backward compatibility. Your phone won’t insert `PAID` in outbound `INVITE` requests. You can further instruct your phone to use `sip:anonymous@localhost` in the `FROM` header by enabling the option **X_UseAnonymousFROM** (that is, in this case your phone uses `From: “Anonymous” <sip:anonymous@localhost>`).

Your phone also includes a `Privacy: id` header if **X_InsertPrivacyHdr** is also enabled.

STUN and ICE

Your phone supports standard STUN based on RFC3489 and RFC5389 for passing inbound RTP packets to phone when behind NAT. The parameters that control the STUN feature can be found under the section **ITSP Profile X – General:**

- **STUNEnable** – Enable this feature (default is `false`).
- **STUNServer** – The IP address or domain name of the external STUN server to use. STUN feature is disabled if this value is blank, which is the default.
- **X_STUNServerPort** – The STUN Server’s listening UDP port. Default value 3478 (standard STUN port).

It should be noted that the STUN feature used in this context is only for RTP packets, not SIP signaling packets (which typically do not require STUN). Your phone sends out a STUN binding request right before making or answering a call on SPx. If the request is successful, your phone decodes the mapped external address and port from the binding response and uses them in the `m=` and `c=` lines of its SDP offer or answer sent to the peer device. If the request fails, such as STUN server not found or not responding, the call goes on without using an external address in the SDP.

Standard RTP requires the use of even numbered ports in the `m=` line. If the external port is not an even number, your phone changes the local RTP port, retries STUN, and continues to do this as many as four times or until an even-numbered external port is found. If the fourth trial still results in an odd-numbered external port number, the call goes on without using external address in the SDP.

Your phone supports standard ICE based on RFC5245. ICE is done on a per-call basis to automatically discover which peer address is the best route for sending RTP packets. To enable ICE on your phone, set the **ITSP Profile X – General::X_ICEEnable** parameter to `yes` (or `true`). The default is `no` (or `false`).

Note that ICE is more effective if STUN is also enabled. However, STUN not a requirement for using ICE on your phone. If STUN is enabled and an external RTP address different from its local address is discovered, your phone offers two ICE candidates in its SDP:

- The local (host) address (highest priority)
- The external (`srfx` or server reflexive) address

Otherwise only the local host candidate is shown in your phone's SDP.



Your phone uses the srflx address in the m= and c= lines of the SDP if STUN is enabled and successful.

If ICE is enabled and the peer's SDP has more than one candidate, your phone sends STUN requests to each peer candidate from its local RTP port. As soon as it receives a response from the highest priority candidate, your phone concludes ICE and uses this candidate to communicate with the peer subsequently. Otherwise your phone allows as long as 5 seconds to wait for a response from all candidates and selects the highest priority one that responds. Once ICE is completed successfully, your phone further applies the symmetric RTP concept to determine the peer's RTP address (that is, sends to the address where the peer's RTP packets are coming from).

ITSP Driven Distinctive Ringing

Your phone offers 10 ring and 10 call-waiting tone patterns in each ring profile. These patterns are numbered from 1 to 10. Each pattern also comes with a configurable name. A different default ring may be assigned to each trunk on your phone.

An ITSP can instruct your phone which ring to use (by name) for a call routed to SP*n* by inserting an Alert-Info header in the SIP INVITE sent to your phone. The Alert-Info must include a URI. For example:

Alert-Info: <http://www.xyz.com/some-folder/bellcore-dr4>

When your phone receives this, it looks for a ring tone name or call-waiting tone name in the ring profile that matches the Alert-Info URI. Ring tone names are not case sensitive when compared. If a match is found, your phone plays the corresponding ring or call-waiting tone. Otherwise, your phone plays the default ring.

RTP Statistics – the X-RTP-Stat Header

When ending an established call, your phone can include a summary of the RTP statistics collected during the call in the SIP BYE request or the 200 response to the SIP BYE request sent by the peer device. The summary is carried in an X-RTP-Stat header in the form of a comma-separated list of fields. The reported fields are:

- PS = Number of Packets Sent
- PR = Number of Packets Received
- OS = Number of bytes sent
- OR = Number of bytes received
- PL = Number of packets lost
- JI = Jitter in milliseconds
- LA = Decode latency or jitter buffer size in milliseconds
- DU = Call duration in seconds
- EN = Last Encoder Used
- DE = Last Decoder Used

For example:

X-RTP-Stat:PS=1234,OS=34560,PR=1236,OR=24720,JI=1,DU=1230,PL=0,EN=G711U,DE=G711U

To enable the X-RTP-Stat feature, the *ITSP Profile X – SIP:X_InsertRTPStats* parameter must be set to true.

RTCP

Your phone supports RTCP (RFC 3550) and RTCP-XR (RFC3611) with MOS statistics for VQ reporting.

Media Loopback Service

Your phone supports the media loopback draft as described in draft-music-media-loopback-13.txt. Your phone supports the following media loopback features:

- Loopback modes: `loopback-source` and `loopback-mirror`
- Loopback types: `rtp-media-loopback` and `rtp-packet-loopback`
- Loopback packet formats: `encaprtsp`, `loopbkprimer`

When acting as a loopback mirror, your phone always sends primer packets so that incoming packets can get through NAT or a firewall. The media loopback feature is controlled by the following parameters (under the *Phone Settings – Calling Features* section):

- **AcceptMediaLoopback** – Enable your phone to accept incoming calls that request media loopback. Default is YES.
- **MediaLoopbackAnswerDelay** – The delay in milliseconds before your phone answers a media loopback call. Default is 0.
- **MediaLoopbackMaxDuration** – The maximum duration to allow for an incoming media loopback call. Default is 0, which means the duration is unlimited.



Your phone rejects an incoming media loopback call if:

- Phone is off hook.
- Phone is ringing.
- One or more calls are on hold.

Your phone terminates an inbound media loopback call already in progress when:

- Phone is off-hook.
- Phone is ringing.

To make an outgoing loopback call, dial one of the following star codes before dialing the target number:

- *03 – Make a Media Loopback Call.
- *04 – Make a RTP Packet Loopback Call.



Outbound Media Loopback Call is not subject to a call duration limit. It lasts until you hang up or until the called number ends the call.

A SIP/SP Configuration Example

The following table details a configuration example where the ITSP Profile to use is B with two SP services, SP1 and SP3, both pointing to the same ITSP Profile.

Configuration Example

Parameter Group	Parameter	Example Value	Notes
<i>ITSP Profile B – General</i>	SignalingProtocol	SIP	Standard default value is SIP
<i>ITSP Profile B – SIP</i>	ProxyServer	sip.phonepower.com	
<i>ITSP Profile B – SIP</i>	OutboundProxy	sbc.phonepower.com	
<i>ITSP Profile B – SIP</i>	OutboundProxy		
<i>SP1 Service</i>	X_ServProvProfile	B	
<i>SP1 Service – Credentials</i>	AuthUserName	3409991003	
<i>SP1 Service – Credentials</i>	AuthPassword	i9cik#dkL	
<i>SP1 Service – Credentials</i>	X_MyExtension	1003	
<i>SP3 Service</i>	X_ServProvProfile	B	
<i>SP3 Service – Credentials</i>	AuthUserName	3409991005	
<i>SP3 Service – Credentials</i>	AuthPassword	c98dfa74{	
<i>SP3 Service – Credentials</i>	X_MyExtension	1005	
<i>Phone Settings</i>	PrimaryLine	SP3	

Call Routing

Call Routing is the process by which your phone sets up a call bridge or an endpoint call based on such information as the trunk on which the call originates, the caller's number, the called number, and so on. Call Routing Rules are parameters used to instruct your phone how to route calls. A call may transform into a call bridge or an endpoint call after being routed by your phone according to the given routing rules.

Every call has to originate from somewhere. From your phone's perspective, calls originating from the trunk side are considered Inbound Calls, while calls originating from an endpoint are Outbound Calls. The call routing rule syntaxes for inbound calls and outbound calls are slightly different and are explained below. Call Routing Rule configuration relies heavily on digit maps. If you are not familiar with how digit maps work yet, please read the [Digit Map Configuration](#) section in this document first.

Basic Incoming Call Handling

For each incoming call from a specific trunk, your phone handles the call in the following order:

- Ignore/reject the call if the trunk is disabled
- Forward the call if native per-line
- Call Forward Unconditional feature is enabled on the service
- Apply Busy Treatment to the call if native per-line Do Not Disturb feature is enabled on the service
- Apply Busy Treatment to the call if the number of existing calls on the trunk exceeds the limit set for the service
- Apply the rules in the **InboundCallRoute** parameter of the service to determine where to send the call. A common destination for an incoming call is `ph` (to ring your phone).



- Busy Treatment refers to whether to reject/ignore the call or apply native per-line Call Forward On Busy if the feature is enabled on the service.
- When it comes to reject/ignore a call, the decision whether to reject or ignore is based on the service.
- You can configure **InboundCallRoute** to enable your phone for complex call handling. For example:
 - ▲ Ring the phone, the AA, and one or more cell phone numbers via `SPn` simultaneously. Whoever answers first takes the call.
 - ▲ If the caller number ends in 4281234 or 3357, ring the AA and a cell phone number simultaneously.

See the [Call Routing](#) section for information on the **InboundCallRoute** parameter.

Inbound Call Route Configuration

Every trunk has a corresponding **InboundCallRoute** in your phone's configuration. It is a comma-separated list of rules where each rule is also surrounded by a pair of curly brackets { }. No extra white spaces are allowed. These rules tell your phone how to handle an inbound call, such as sending it to the Phone (and ringing the attached phone(s)), sending it to the Auto Attendant for further routing (interactively with the caller), or making another call on a specific trunk to bridge with this call.

The general format is:

```
InboundCallRoute := rule OR {rule},{rule},...
```

The curly brackets may be omitted if there is only one rule in the route. The OR operator is NOT part of the parameter syntax. It is used here to separate alternative values only.

A rule has the following format:

Inbound Call Route Structure

Inbound Call Route Structure				
{rule},{rule}, {rule}, {rule}				
rule =				
{peer-list		:	terminal-list}	
peer-list =				
{(peering),(peering), (peering),		:	(terminal), (terminal)}	
peering =		terminal =		
{(caller-list	>	callee-list)	:	PHx OR AAx OR LIx(arg) OR SPx(arg) OR PPx(arg) }
caller-list =		callee-list =	arg =	
{caller caller caller	>	callee callee}	cid	> target
caller =		caller =	cid =	target =
number OR (digit-map) OR ? or @ (? = anonymous, @ = any #)		number OR (digit-map) OR @	(digit-map) OR spoofed-caller -number (the outbound CallerID)	(digit-map) OR number-to-call



- The `terminal-list` can be empty, which means to block this call. The preceding `:` cannot be omitted. As many as 4 terminals may be specified in the list. The listed terminals are called/rung by your phone simultaneously. This operation is called forking the call. A terminal may be a trunk or an endpoint.
- Abbreviated terminal names are case-insensitive.
- The `number` and `number-to-call` are literal strings, such as 14089991234.
- The `digit-map` is just any proper digit map, such as (1xxx|xx.). Make sure to include the enclosing parentheses
- The `spoofed-caller-number` is a literal string, such as 14081112233, to be used as the caller number for making a new call on the specified trunk
- (Mlabel) is a named digit map, where `label` is the abbreviated name of any terminal that has a digit map defined: SP1-SP6, PP, PH, or AA.
- `$1` is an internal variable containing the value of the caller number of this inbound call, after any digit map transformation in the matched caller object of the matched peering object in the peering-list.
- `$2` is an internal variable containing the called number of this inbound call, after any digit map transformation in the matched `callee` object of the matched peering object in the peering-list.

More notes on peering-list and peering objects:

- The `peering-list` is optional in **InboundCallRoute**. If `peering-list` is empty, the succeeding `:` can be omitted also. An empty peering-list implies a single peering object whose caller object list matches any caller number. That is, the following **InboundCallRoute** values are all equivalent:
 - ^ ph
 - ^ {ph}
 - ^ {:ph}
 - ^ {?|@>@:ph}
- The `callee-list` in a peering object can be empty. It implies the callee object @, meaning any called number. The preceding `>` can be omitted if `callee-list` is empty.
- The `caller-list` in a peering object can be empty. It implies the caller-list @|?, meaning any caller number including `anonymous`. The succeeding `>` cannot be omitted if `caller-list` is empty but not the `callee-list`.

More notes on the `arg`, `cid`, and `target` objects:

- The `cid` object inside an `arg` object is optional. If omitted, it implies no caller-ID spoofing when making the call on the specified trunk. The succeeding `>` can be omitted if `cid` is omitted.
- The `target` object inside an `arg` object is optional. If omitted, it implies the `target $2`, which means to call the original called number after applying any necessary digit map transformation implied by the rule. The preceding `>` cannot be omitted if `target` is omitted but `cid` is not.
- The `arg` object is optional. If omitted, it implies the `arg` with the `target $2` and no `cid`. If `arg` is omitted, the succeeding parentheses () can be omitted also.

An inbound call matches a rule if its caller-number/callee-number matches one of the peering objects of the rule. Peering objects are tested in the order left to right, and the first matched peering object wins. Rules are also checked in the order left to right, and the first matched rule wins. Therefore, it is important that you place the more specific rules first in the **InboundCallRoute** if multiple rules can potentially match the same inbound call.

InboundCallRoute examples:

- `ph OR {ph} OR {:ph} OR {@|?>@:ph}` (all equivalent)
It says: Ring the phone (only) for all incoming calls. This is the default **InboundCallRoute** for all trunks.
- `{14081223330|15103313456:ph,aa},{(1800xx.|1888xx.)|?:},{ph}`
It says: Ring the phone and AA for calls coming from 1 408 122 3330 or 1 510 331 3456, block all 800, 888, and anonymous calls, and ring just the phone for all other calls.
- `{(x.4081113333|x.4152224444):aa},{ph}`
It says: Ring the AA for calls coming from any number that ends with 408 111 3333 or 415 222 4444, and ring the phone for all other calls. Be sure to include the enclosing parentheses in this example since "x." is a digit map specific syntax.
- `{200123456:aa},{sp1(14083335678)}`
It says: Ring the AA for calls coming from 200123456. For all any other call, bridge it by calling 1 408 333 5678 using SP1 Service.

Basic Outgoing Call Handling

When the trunk receives a number to call, it makes the call if the trunk is enabled, running, and under full capacity. Otherwise, the trunk fails to send the outgoing call.

Although there's a **DigitMap** parameter available per service, it's however not used by the trunk to validate the number to call. The validation is done at a higher level before the call attempt is routed to the trunk for execution. The per-line **DigitMap** is used as a reference in other **DigitMap** parameters (usually in **Phone Settings::DigitMap**) and in trunk selection from within a trunk group. The latter is more relevant to the trunk itself: When a trunk group is determined as the destination of an outgoing call, your phone selects the trunk from the group by taking into account whether the number to call is valid against the rules in the **DigitMap** of the trunk.

Outbound Call Route Configuration

Every endpoint has an **OutboundCallRoute** parameter in your phone's configuration. It tells your phone where to send the call when the endpoint attempts to make a call. Endpoints may call each other or an outside number using one of the trunks. The **OutboundCallRoute** syntaxes are almost identical to those of the **InboundCallRoute** parameter. The differences are mainly in the implied value when an optional field is omitted, there are no caller objects, and one and only one terminal object per terminal-list is present in an **OutboundCallRoute** parameter. Forking is not supported when routing outbound calls.

The general format is:

OutboundCallRoute := rule **OR** {rule},{rule},...



The curly brackets may be omitted if there is only one rule in the route. The **OR** operator is NOT part of the parameter syntax. It is used here to separate alternative values only.

A rule has the following format:

Outbound Call Route Structure

Outbound Call Route Structure		
<code>{rule}, {rule}, {rule}, {rule}</code>		
<code>rule =</code>		
<code>{callee-list</code>	<code>:</code>	<code>terminal}</code>
<code>callee-list =</code>		<code>terminal =</code>
<code>{callee callee</code>	<code>:</code>	<code>PHx, AAx, SPx, LIx, OR PPx}</code>
<code>callee =</code>		
<code>number OR (digit-map) OR @</code>		<code>(@ = any number)</code>



- A terminal may be a trunk or another endpoint.
- Abbreviated terminal names are case-insensitive.
- The `number` and `number-to-call` are literal strings, such as 14089991234.
- The `digit-map` is just any proper digit map, such as (1xxx|xx.). Make sure to include the enclosing parentheses.
- The `spoofed-caller-number` is a literal string, such as 14081112233, to be used as the caller number for making a new call on the specified trunk.
- (Mlabel) is a named digit map where label is the abbreviated name of any terminal that has a digit map defined: SP1-SP6, PP, PH, or AA.
- \$2 is an internal variable containing the called number of this outbound call, after any digit map transformation in the matched callee object.
- The `callee-list` can be empty, which implies the single callee object @, which means any called number. The succeeding ':' can be omitted also when callee-list is empty.

More notes on the *arg*, *cid*, and *target* objects:

- The *cid* object inside an *arg* object is optional. If omitted, it implies no caller-ID spoofing when making the call on the specified trunk. The succeeding '>' can be omitted if *cid* is omitted.
- The *target* object inside an *arg* object is optional. If omitted, it implies the *target* \$2, which means to call the original called number after applying any necessary digit map transformation implied by the rule. The preceding '>' cannot be omitted if *target* is omitted but not the *cid*.
- The *arg* object is optional. If omitted, it implies the *arg* with the *target* \$2 and no *cid*.

An outbound call matches a rule if its called number matches one of the callee objects of the rule. Callee objects are tested in the order left to right, and the first matched callee wins. Rules are also checked in the order left to right, and the first matched rule wins. Therefore, it is important that you place the more specific rules first in the **OutboundCallRoute** parameter if multiple rules can potentially match the same outbound call.

Note that every endpoint also has a digit map defined. The user-dialed number is completely processed with the endpoint's digit map first before it is passed to the **OutboundCallRoute** parameter for routing decision. Therefore the number used for matching call routing rules has already incurred the transformations, if any, implied by the digit map. Remember this fact when crafting your own **OutboundCallRoute** parameter.

OutboundCallRoute examples:

- `sp1 OR {SP1} OR {:SP1} OR {@:Sp1}` (all equivalent).

This rule says: Make all calls using SP1 Service, without any caller-id spoofing or digit transformation.

- `{**0:aa}, {***:aa2}, {(Mpli):pli}, {(<**1:>(Msp1)):sp1}, {(<**2:>(Msp2)):sp2}, {(<**9:>(Mpp)):pp}`

This is the default **Phone Settings::OutboundCallRoute** parameter. It says:

- Dial ****0** to invoke AA.
- Dial ******* to invoke the local phone configuration IVR (also known as AA2).
- `(Mpli)` and `pli` are substituted with the abbreviated name of the **PrimaryLine** value.
- Use SP1 Service to call all numbers that start with ****1** and subsequent digits matching SP1 Service's **DigitMap** parameter. Remove the ****1** prefix from the resulting number before making the call.
- Use SP2 Service to call all numbers that start with ****2** and subsequent digits matching SP2 Service's **DigitMap** parameter. Remove the ****2** prefix from the resulting number before making the call.
- Use the OBITALK Service to call all numbers that start with ****9** and subsequent digits matching OBITALK Services **DigitMap** parameter. Remove the ****9** prefix from the resulting number before making the call.

Voice Gateways and Trunk Groups

Voice Gateways

A gateway in this context is another phone that lets incoming OBITALK callers to call further on one or more of its trunks (such as SP1, SP2, or PP). The caller can call the gateway first with a normal OBITALK call, get the AA, and then dial the target number. For authentication, the AA may ask you to enter a PIN before establishing the second call. This way of dialing is known as 2-stage dialing.

On the other hand, a gateway can be configured on the originating phone such that the caller can dial the target number directly without going through the AA. This way of dialing is known as direct dialing or 1-stage dialing. Since it is not possible to enter a PIN in the case of direct dialing, a user/password pair can be configured for the gateway also so that the device can authenticate with the gateway automatically using the HTTP digest method. The HTTP digest authentication is optional. You do not need to provide a user/password if the gateway does not require authentication for direct dialing.

The phone allows you to specify as many as eight gateways. Each gateway is addressed using its factory-assigned OBi number. A gateway is conceptually a trunk with its own digit map. You can refer to a gateway and its associated digit map with the short trunk name `VG n` and `(Mvgn)` respectively, for $n = 1, 2, 3, \dots, 8$. `VG n` and `(Mvgn)` can be used in call routing rules and digit maps just like other real trunks.

As an example, you can add the rule `{(1xxx xxx xxxx):vg2}` in **Phone Settings::OutboundCallRoute** to let the device dial out using VGs when the caller dials any 11-digit number starting with 1. On the gateway side, you can add the corresponding rule `{>(1 xxx xxx xxxx):sp1}` in the **OBITALK Service::InboundCallRoute** to make the call on its SP1 trunk. You can change the last rule to `{(290 333 100|200 444 101)>(1 xxx xxx xxxx):sp1}` if you want to limit the gateway to allow just the two stated caller numbers to make such calls.

A gateway may also be configured with a SIP URL as the access number to be accessed by the device over one of the SP trunks. For example, one can set the gateway access number as SP1 (some-sip-server.mydomain.com), or SP2 (192.168.15.111:5062), and so forth. Note that when using an SP trunk to access a (SIP) gateway, the device:

- Won't use the outbound proxy, ICE, or STUN regardless the settings on the SP trunk.
- Uses only the device's local address as the SIP Contact, and ignores any netted address discovered by the device.
- Uses the gateway's SIP URL to form the FROM header of the outbound INVITE.
- Uses the gateway's **AuthUserID** and **AuthPassword** for authentication.
- Applies the symmetric RTP concept.

Trunk Groups

As the name implies, a trunk group is a group of trunks. If a call is routed to a trunk group, your phone picks one of the available trunks from the group to make the call. Availability of trunk is based on:

- Whether the trunk's digit map allows the number to call, AND
- Whether the trunk has capacity to make one more call

As many as four trunk groups can be configured on your phone. Each trunk group is conceptually another trunk with its own digit map. A trunk group and its associated digit map are referenced using the short name TG_n and (Mtgn) respectively, where $n = 1, 2, 3, 4$. They can be referenced in other digit maps and call routing rules so that calls may be routed to a particular trunk group.

Only trunks can be added to a trunk group. These include: PP, SP1 – SP6, VG1, VG2, ..., VG8, TG1, TG2, ... TG4.



A TG may include another TG (that is, TG can be recursive). However, you must make sure this does not result in infinite recursion.

Trunks, Endpoints, and Terminals

Your phone is also a Voice Service Bridge (VSB) that supports multiple voice services. It can bridge calls across any of the supported services. A *call bridge* is a voice connection connecting two calls on the same or different voice services. Your phone allows four concurrent independent call bridges. The following matrix shows the possible call bridge connections.

Supported 2-Way Call Bridges

	SP1 Service	SP2 Service	SP3 Service	SP4 Service	SP5 Service	SP6 Service	OBiTALK Service
SP1 Service	yes	yes	yes	yes	yes	yes	yes
SP2 Service	yes	yes	yes	yes	yes	yes	yes
SP3 Service	yes	yes	yes	yes	yes	yes	yes

Supported 2-Way Call Bridges

	SP1 Service	SP2 Service	SP3 Service	SP4 Service	SP5 Service	SP6 Service	OBiTALK Service
SP4 Service	yes	yes	yes	yes	yes	yes	yes
SP5 Service	yes	yes	yes	yes	yes	yes	yes
SP6 Service	yes	yes	yes	yes	yes	yes	yes
OBiTALK Service	yes	yes	yes	yes	yes	yes	yes

Each supported service is also referred to as a trunk, a traditional telco term for a physical wire or wires that deliver phone services to homes or businesses. Each trunk is represented with 2-letter abbreviation and a 1-based instance identifier:

- SP1 = the SP1 Voice Service (with ITSP A, B, C, D, E or F)
- SP2 = the SP2 Voice Service (with ITSP A, B, C, D, E or F)
- SP3 = the SP3 Voice Service (with ITSP A, B, C, D, E or F)
- SP4 = the SP4 Voice Service (with ITSP A, B, C, D, E or F)
- SP5 = the SP5 Voice Service (with ITSP A, B, C, D, E or F)
- SP6 = the SP6 Voice Service (with ITSP A, B, C, D, E or F)
- PP1 = the OBiTALK Service

The instance identifier may be omitted if it is equal to 1. Hence, PP is equivalent to PP1, and so forth. These short-hand notations are used heavily in configuring your phone, as found in call routes, call forward numbers, and speed dials parameters. Unless stated otherwise, the abbreviated trunk names are case-insensitive.

The Phone (PH) and the AA are the entities in your phone where calls can terminate (start or end there), as opposed to the trunks, which rely on the corresponding service providers to terminate the call. In this document, the phone and the AA are referred to as endpoints. Like the trunks, each endpoint is represented by a 2-letter abbreviation and a 1-based instance identifier:

- PH = The Phone
- AA = The Auto Attendant

Unless stated otherwise, abbreviated endpoint names are case-insensitive. A trunk or an endpoint is also referred to as a *terminal* in this document.

The following matrix shows the possible call connections between the endpoints and the trunks:

Supported Endpoint Calls

	Any Trunk	PH1	AA
Any Trunk	n/a	yes	yes
PH	yes	no	yes
AA	yes	yes	no

Third-Party Servers

This section provides information on configuring phones and features with third-party servers.

BroadSoft

This section shows you how to configure Polycom OBi Edition devices with BroadSoft Server options.

Built-In BroadSoft Phone Applications

The **Net Services** and the **Net Directory** built-in phone apps rely on the BroadSoft/BroadWorks switch.

Built-In BroadSoft Phone Applications

Display Caption (Default)	Action URL	Summary
Net Services	Phone://netsrv	A collection of settings of features the service provider offers. The settings are organized on a per SP Service basis. This App is only applicable with a BroadSoft softswitch.
Net Dir	Phone://netdir	A directory service offered by a service provider, either BroadSoft directory or LDAP.

Net Services App

BroadSoft offers a set of user features that you can access through a few APIs that correspond to an SP service on the phone. These features are listed under **Network Services per SP** service on the phone. The settings of these features are available under the **Net Services** App on the Main menu.

The top level of the App shows a list of the SP services that have network features enabled and the status of the enabled features:

- DND On or Off
- Call Forward All On or Off
- ACD Agent State
- Anonymous Call On or Off

The softkey set on this screen is configurable with these options:

- BCI
- DND
- CFA

- ACD

You can select the SP service to view or change all the available settings for the SP service.

The ordering and the list of items to include in the second-level menu are configurable with the **NetServicesMenu1** parameter.

Net Dir App




The usage depends on the app provider. There are two options:

- BroadSoft Directory
- LDAP

Call Recording Controls

(SIP/BroadSoft.) Your phone supports the call recording functions available with a BroadSoft application server, by providing the controls for call recording during a call. This feature can be enabled on a per SP service basis by enabling the option **SPn Service – Network Provided Services::CallRecording**. When the feature is enabled, the phone shows one of the following call recording states in the call items of the Calls App:

Call Recording Controls

Call Recording State	Description	Available Softkey Options	Icon
off	Recording has been turned off	Rec.Start	
on	Recording has been turned on	Rec.Stop, Rec.Pause	
paused	Recording has been turned on but paused at the moment	Rec.Resume	
na	Feature is not enabled		

The softkey options for recording controls options are available only when the call is in the Connected or the Holding state.

BroadSoft Call Center Features

This suite of features supports call center applications with a BroadSoft softswitch.

Disposition Code

A disposition code can be entered by an agent for the current call that is still ongoing or for the last call that has just ended. For the first case, the agent selects the **Dispose Code** softkey that is available when the call is in the connected state. The agent then enters the code and submits it while talking to the caller. For the latter case, the agent can press the feature key that has been assigned the `Disposition Code` function right after the call, then enter and submit the code.

To use this feature on the phone, you must enable the option **SPn Service – Network Provided Services::DispositionCode**. The option only applies to calls on the same SP service.

Customer Originated Call Trace

The agent can start a call trace during the call by pressing the **Trace** softkey when the call is in the Connected state. The actual call trace function is executed entirely on the softswitch after invocation from the phone. To use this function on the phone, you must enable the option **SPn Service – Network Provided Services::CallTrace**. The option only applies to calls on the same SP service.

To start a call trace outside the context of a call, the administrator can define a speed dial feature key with the following parameters:

Function = Speed Dial

Number = customer-originated-trace

Server = The SP Service to invoke the Call Trace operation with, such as SP3

Escalation

The agent can escalate the current call to a specific supervisor or the predefined default supervisor by selecting the **Escalate** softkey that is available when the call is in the Connected State. After pressing the key, the agent has a chance to enter a specific supervisor's extension, or skip that to use the default supervisor's extension. To use this feature on the phone, you must enable the option **SPn Service – Network Provided Services::Escalation**. The option only applies to calls on the same SP service.

Call Center Information

When the softswitch sends an incoming call to the phone, it may include some basic information about the call center where the call is coming from. Such information, if available, is displayed by the phone with the Ring Alert message for the incoming call. The following information can be displayed:

- Call Center name
- Call Center user ID
- Average waiting time
- Number of calls in the queue

This behavior as described above does not have any configuration.

In addition, the phone can proactively subscribe to the status of the call center that a SP service belongs to. This feature is enabled with the option **SPn Service – Network Provided Services::CallCenter**. The call

center status information can be viewed at any time on the phone by going through the Net Services App (from the phone's main menu).



BroadSoft Guest Login/Logout (Hoteling)

(SIP only.) This feature is also known as Hoteling (consult BroadSoft documentation on how to administer this feature on the server). The phone may be set up to be used temporarily by a guest, such as a visiting employee or temp worker in a hot desk environment. To use this feature on the phone, the option **SPn Service – Network Provided Services::Hoteling** must be enabled. If enabled, the phone starts a subscription to the x-broadworks-hoteling event package in the context of a SP service, right after the first successful registration with the SIP Proxy Server. The expires value of this subscription can be set with the **ITSP Profile – SIP – Feature Configuration::X_BWHotelingSubscribeExpires** parameter.

There are two ways to access the guest login/logout function:

- Within the **Net Services App**, select the SP service, and then select the Hoteling option
- Define a feature key with the function `Hoteling` (but no more than one hoteling feature key per SP service). The LED and the icon of the key reflect the current guest login state, as shown in the following table:

Hoteling Settings

Guest Login State	Description	Icon	LED
Guest Logged In	A guest has successfully login. The login extension is shown on the screen		Solid green
Guest Logged Out	No guest login		Off
Protocol error	Problem reaching the server or subscribing to the service		Off

When trying to login, the guest is prompted to enter the guest extension first. After submitting a valid guest extension, the guest is prompted to enter the corresponding password. Once the password is accepted by the server, the phone screen shows the guest extension in the **Call Appearance** and **Line Monitor** keys that are bound to the SP service. The guest logs out by pressing the **Hoteling** feature key again once, or the server logs out the guest remotely. The screen updates automatically according to the updated guest login state.

BroadSoft AS-Feature-Event Features

This is a collection of network-provided (that is, softswitch-provided) features available on a BroadSoft Application Server. You can view and change the settings from the phone UI. Note that these network-provided features are configured and executed in the context of a single SP service. To allow any of the network-provided features listed below to be viewable and changeable from the phone, you must enable the option **SPn Service – Calling Fetures::X_ASFeatureEventSubscribe** and must also enable the individual network-provided feature whichever you want to allow users to access from the phone.



The features themselves are executed entirely on the server and the settings of the features are stored on the server. The phone displays the values of the settings as stored on the server (not the ones entered and submitted by user, which may or may not be acceptable by the server).

The as-feature is based on the SIP subscribe/notify framework. The expires value of the subscription dialog (initiated by the phone per SP service with the feature enabled) can be set using the ***ITSP Profile X–Feature Configuration::X_ASFeatureEventSubscribeExpires*** parameter. On each subscribe request sent by the phone, the server returns a NOTIFY that refreshes the current settings of all the softswitch-provided features listed below (whichever is enabled or made available on the server). When a setting is changed, the server also updates the phone with a NOTIFY that specifies the latest settings of just the affected features.

Some of the network-provided features can be accessed from the phone via special feature key functions (such as Do Not Disturb) or special soft keys (such as Call Forward All), while all enabled network-provided features can be accessed via the Net Services App of the Main menu.

Call Forward All

CallForwardAll is synonymous to CallForwardUnconditional in this document. This feature, if enabled, lets the softswitch forward all calls to the configured forwarding number unconditionally. The functionality provided by this feature is similar to that of the CallForwardUnconditional feature provided natively by the phone (per line). The administrator is advised to disable the native version when using the network-provided version to avoid ambiguity. To make the setting of this network-provided service viewable and changeable from the phone UI, you must enable the option ***SPn Service – Network Provided Services::CallForwardAlways***. In addition to going through the Net Services app, the following methods of access are available:

- Feature Key: Define a feature key with the function Call Forward. Set the Service parameter of the key to the SP service that provides this feature.
- Softkey: The **Call Forward** softkey on the home screen of the phone can be mapped to the network provided CallForwardAll service on a specific SP service, by setting the ***Phone Settings – User Preferences Settings::CallForwardUnconditionalFeatureProvider*** parameter to equal to that SP service.
- User Preferences – Call Forward Setting: As a side effect of enabling the softkey option in the last item, the Call Forward Setting under User Preferences app on the UI is also pointing to the same network provided **CallForwardAlways** feature.

You can enable or disable **CallForwardAlways** as well as setting a forwarding number. The settings are submitted and stored on the server.

Call Forward Busy

To make the setting of this network-provided service viewable and changeable from the phone, you must enable the option ***SPn Service – Network Provided Services::CallForwardBusy***. The functionality provided by this feature is similar to that of the CallForwardOnBusy feature that is available natively on the phone (per line). To use the version provided by the softswitch, the administrator is advised to disable the native version to avoid ambiguity.

The only way of access from the phone is by going through the Net Services app. The phone allows you to get and change the setting of this feature on the server. The setting is submitted to the server with a subscribe request and stored on the server.

Call Forward No Answer

To make the setting of this network-provided service viewable and changeable from the phone, you must enable the option ***SPn Service – Network Provided Services::CallForwardNoAnswer***. This feature is similar to the CallForwardOnNoAnswer feature that is available natively on the phone (per line). To use the version that is provided by the softswitch, the administrator is advised to disable the native version to avoid ambiguity.

The only way of access from the phone is by going through the Net Services app. You can enable or disable the feature, set the forwarding number, and specify the number of rings before forwarding the call. These settings are submitted to the server with a subscribe request and stored on the server.

Do Not Disturb

To make the setting of this network provided service viewable and changeable from the phone, you must enable the option ***SPn Service – Network Provided Services::DoNotDisturb***. The functionality provided by this feature is similar to that of the DoNotDisturb feature that is available natively on the phone (per line). To use the version that is provided by the softswitch, the administrator is advised to disable the native version to avoid ambiguity.

In addition to going through the Net Services app, the setting can also be accessed via:

- Feature Key: Define a feature key with the function `Do Not Disturb`. Set the Service parameter of the key to the SP service that provides this feature.

You can enable or disable this feature from the UI. The setting is submitted to the server with a subscribe request and stored on the server.

ACD Agent State

ACD stands for Automated Call Distribution and is the primary way a call center distributes calls among a number of agents. Normally the ACD controller only sends a new call to an agent who is in the “Available” state. An agent typically “Signs On” when they arrive at work and then “Signs Off” when done for the day (or taking a very long break). Agents may at any time tag themselves as “Unavailable” when taking a break, or “Wrapping Up” when filing paperwork for the last call before taking another call. Through as-event subscription, your phone allows an agent to sign on, sign off, or change their availability states from the phone UI. To make this feature available on the phone, you must enable the option ***SPn Service – Network Provided Services::ACDAgent***.

You can invoke this feature in two ways:

- Press the feature key with the function ACD Agent
- Within the Net Services App, select the SP service, and then select the **ACD Sign On/Off** item

With the ACD feature key, the agent can sign on by pressing the key once.

You can sign on or off and change your state from the phone UI. You can set your state to one of the following values:

- Available (to take new calls)
- Unavailable (to take new calls)
- Signed On
- Signed Off
- Wrapping Up (the last call)

While Signed Off, press the key once to sign on and become Available. While Available, press the key once to become Unavailable. You must also enter one of the valid unavailable reason codes (such as 11) that are defined by the Call Center admin. While Unavailable or Wrapping Up, you can press the key once to become Available again.



You cannot change your state to Signed Off or Wrapping Up directly by pressing the feature key. To change to these states, you must use the corresponding feature key menu item from the UI (invoked by pressing and holding down the feature key), or some other means provided by the softswitch, such as a web portal for agents.

Security Classification

To make the Security Classification setting viewable and changeable from the phone, you must enable the option **SPn Service – Network Provided Services::SecurityClass**. With that the phone lets you view or set the security levels for your extension. The currently available security levels you are allowed to choose are presented to you on the screen. In addition to invoking this function by going through the Net Services App, the administrator can also define a feature key with the function `Security Class` as a shortcut to launch this function.

Executive Call Filter

To make the setting of the Executive Call Filter option viewable and changeable from the phone, you must enable the option **SPn Service – Network Provided Services::Executive**. In addition to invoking the function through the Net Services App, the administrator can define a feature key with the function `Exec Filter On/Off` as a shortcut to turn this setting on or off. The LED color also reflects the current on or off status.

Executive Assistant

By enabling the option **SPn Service – Network Provided Services::ExecutiveAssistant**, the phone makes the following settings of this feature available to user:

- From a list of Executives currently associated with the assistant, turn the call filtering feature on or off for any of the executives.
- Enable or disable the **Divert** option and set or modify the phone number to divert to.

In addition to invoking the function through the Net Services App, the administrator can define a feature key with the function `Exec Assistant` as a shortcut to launch this option. The LED color also reflects the current **Divert** on or off status.

Call Recording Settings

The phone can extract the call recording settings from the as-event notifications to help determine if and which call recording controls to present to you during a call, provided the **SPn Service – Network Provided Services::CallRecording** is also enabled.

BroadSoft XSI Features

This is a collection of features that is provided with a BroadSoft XSI Application Server. The phone makes XSI Features available per SP/SIP service. This allows the configuration of as many as six independent sets of XSI services per phone, one per SP service. To execute any of the features listed in this section, the following parameters are required:

BroadSoft XSI Features

Parameter Group	Parameter	Description
<i>ITSP Profile X – SIP</i>	X_XsiServer	The XSI server hostname or IP address. Phone attempts to resolve the hostname as DNS A Record only. DNS SRV lookup isn't supported here.
<i>ITSP Profile X – SIP</i>	X_XsiServerPort	The server port. If not specified (or 0), the default port is used (80 for HTTP or 443 for HTTPS).
<i>ITSP Profile X – SIP</i>	X_XsiServerScheme	Must be HTTP or HTTPS.
SPn Service – SIP Credentials	X_XsiUserName	The user name to authenticate to the XSI server with. If not specified (blank), the phone forms the user name as: <code>{sip-userid}@{sip-domain}</code> where <code>{sip-userid}</code> is the SIP Account User ID that is used for SIP Registration on the same SP service, and <code>{sip-domain}</code> is the domain name that is used for SIP Registration on the same SP service.
SPn Service – SIP Credentials	X_XsiPassword	The password to authenticate to the XSI server with. If not specified (blank), the same password for SIP authentication on the same SP service is used.

Some of the XSI features can be accessed from the phone by launching dedicated apps (such as Network Directories) or via special feature key functions (such as Do Not Disturb) or special soft keys (such as Call Forward All). In addition, all enabled XSI services can also be accessed under the Net Services app of the Main menu.



- **CallForwardAll**, **CallForwardBusy**, **CallForwardNoAnswer**, and **DoNotDisturb** can also be accessed using the **AS-Feature-Sync** method. If **AS-Feature-Sync** is enabled, then it is used instead of XSI to access these features.
- Except for network directories and call logs, the XSI server needs to notify the phone when settings have changed on the server. This notification is done using an event channel over HTTP. The method for sending HTTP messages is to use Comet.

Network Directories

Network Directories are directories hosted by a server somewhere in the network.

BroadSoft Hosted PBX Platform

With the BroadSoft BroadWorks platform, the phone supports the following types of network directories:

- Group
- Group Common
- Enterprise
- Enterprise Common
- Personal

Please consult your BroadSoft documentation on how to setup and manage these directories on the server side.

To access this service from the phone, you must enable the option **SPn Service – Network Provided Services::Directory**. You can invoke the Network Directory service of a specific SP service from the phone by launching the Net Dir app from the Main menu of the phone UI. Which SP service's Network Directories service is invoked is controlled by the **Phone Settings – Network Directory::VoiceService** parameter. In order for the Net Dir item to show on the phone's Main menu, both **Phone Settings – Network Directory::Enable** and **SPn Service – Network Provided Services::Directory** of the corresponding SP service must be enabled. You can access the Network Directories on other SP services through the Net Services App.

It should be noted that all the directory data are stored entirely on the server and are downloaded once when the network directories function is invoked on the phone. After initial invocation, you can refresh the data by pressing the **Refresh** or **Refresh All** softkey. A Search function lets you enter a search string of a name pattern (such as OBi*) and submit it to the server. The server then returns a list of entries matching the search criteria to display on the phone screen.

Network Directory Softkey Options

The network directory results screen provides four soft keys:

- **Call**: Call the highlighted entry.
- **Search**: Enter one or more search fields and press **OK** or **Enter** to start the search.
- **Prev Page**: Display the previous page of results.
- **Next Page**: Display the next page of results.

You can disable the Search option by setting the **SPn Service – Network Directory Setup::EnableSearch** to `false`. This replaces the **Search** softkey with the **Refresh** softkey, which reloads the results from the URL.

Other PBX Platforms

A service provider can host a Group directory similar to the format used in BroadWorks on the network. The URL to access this directory is configured in the: **SPn Service – Network Directory Setup::URL** parameter. For example:

URL = <https://mypbx.com/group-dir.php?user=jsmith>

The returned document should be an XML (properly escaped) as shown in the example below:

```
<?xml version='1.0' encoding='utf-8' ?>
<Group>
  <startIndex>1</startIndex>
  <numberOfRecords>2</numberOfRecords>
  <totalAvailableRecords>2</totalAvailableRecords>
  <groupDirectory>
    <directoryDetails>
      <name>Samuel Samson</name>
      <firstName>Samuel</firstName>
      <lastName>Samson</lastName>
      <userId>ssamson</userId>
      <number>14089991234</number>
      <extension>104</extension>
      <department>Marketing</department>
      <groupId>Management</groupId>
      <impId>ssamson@abcd-wallpaper.com</impId>
      <additionalDetails>
        <emailAddress>sam.samson@abcd-wallpaper.com</emailAddress>
        <mobile>14089998899</mobile>
      </additionalDetails>
    </directoryDetails>
    <directoryDetails>
      <name>John J. Smith</name>
      <firstName>John</firstName>
      <lastName>Smith</lastName>
      <userId>john</userId>
      <number>14087771123</number>
      <extension>103</extension>
      <department>Sales</department>
      <groupId>Staff</groupId>
      <impId>jjs@abcd-wallpaper.com</impId>
      <additionalDetails>
        <emailAddress>john.smith@abcd-wallpaper.com</emailAddress>
        <mobile>16509991802</mobile>
      </additionalDetails>
    </directoryDetails>
  </groupDirectory>
</Group>
```

In addition, you can append the following optional URL parameters to the URL:

- **results** Number of results to return by the server. This is merely a suggestion to the server which may return a different number of results. The phone won't reject the result if the number of results returned is different from the request and makes a best effort to display all the available results.
- **start** The 1-based index of the first result to return by the server. The implied start index is 1 if not specified.

Example: <https://mypbx.com/group-dir.php?user=jsmith&start=30&results=20>

You may also include user name and password in the URL for authentication to the user, such as:

<https://jsmith:!4xuIKK12y@mypbx.com/group-dir.php?user=jsmith>

Replace the Built-In Phone Book with a Network Directory

You can replace the built-in phone book with a Network Directory, such that when you press the **phbk** softkey or select **Contacts** from the Main menu, your phone launches the corresponding network directory instead of the built-in (local) phone book. Below is the configuration for this:

User Preferences – Phone Book Settings::ActionURL = `phone://netdir`

Phone Settings – Network Directory::VoiceService = `SPn`

(where n = 1, 2, 3, ..)

Network Call Logs

The network call logs consist of four logs: All, Missed, Received, and Outgoing. The log data are stored entirely on the server and are downloaded to the phone when you invoke this function from the phone. Please consult BroadSoft on how to manage these call logs on the server side.

To make this function available on the phone, you must enable the option **SPn Service – Network Provided Services::CallLogs**. There is no specialized app, feature key functions or softkey options to launch network call logs. You can only invoke this function by going through the Net Services app.

If the Buddy List is also enabled and available under the same SP service, the phone also displays the presence icon in the network directory if it can be found in the buddy list.

BroadWorks Anywhere

You can view and change the following settings of this feature from the phone UI:

- Turn the **Alert all locations for Click-To-Dial Calls** option on or off.
- Turn the **Alert all locations for Group Paging Calls** option on or off.
- Enable or disable a location.
- Add a location.
- Remove a location (Note: The last location cannot be removed via XSI).
- Edit a location's **Number** or **Description** attributes.

To make this function available on the phone, you must enable the option **SPn Service – Network Provided Services::BroadWorksAnywhere**. There is no specialized app, feature key functions or softkey options to launch this function. You can only invoke this function by going through the Net Services App.

Remote Office

You can view and change the following settings of this feature from the phone UI:

- Enable or disable this feature.
- Change the Remote Phone Number (Note: The number cannot be removed via XSI).

To make this function available on the phone, you must enable the option **SPn Service – Network Provided Services::RemoteOffice**. There is no specialized app, feature key functions or softkey options to launch this function. You can only invoke this function by going through the Net Services App.

Simultaneous Ring

You can view and change the following settings of this feature from the phone UI:

- Enable or disable this feature by turning the **Active** option on or off.
- Turn the **Do not ring my Simultaneous Ring Numbers if I'm already on a call** option on or off.
- Add a location.
- Remove a location (Note: The last location cannot be removed via XSI).
- Change the **Phone Number** attribute of a location.
- Turn the **Answer confirmation required** option on or off for each location.

To make this function available on the phone, you must enable the option **SPn Service – Network Provided Services::SimultaneousRing**. There is no specialized app, feature key functions, or softkey options to launch this function. You can only invoke this function by going through the Net Services App.

Call Forward Always

To make this function available on the phone, you must enable the option **SPn Service – Network Provided Services::CallForwardAlways**. This feature is similar to the **CallForwardUnconditional** feature that is available natively on the phone (per service). To use the version that is provided by the softswitch, the administrator should disable the corresponding service on the phone to avoid ambiguity.

In addition to going through the Net Services app, the following methods of access are available:

- Feature Key: Define a feature key with the function Call Forward. Set the Service parameter of the key to the SP service that provides this feature.
- Softkey: The Call Forward softkey on the home screen of the phone can be mapped to the network provided **CallForwardAll** service on a specific SP service, by setting the **Phone Settings – User Preferences Settings::CallForwardUnconditionalFeatureProvider** parameter to equal to that SP service.
- User Preferences – Call Forward Setting: As a side effect of enabling the softkey option in the last item, the Call Forward Setting under the User Preferences app on the UI points to the same network-provided **CallForwardAlways** feature.

You can enable or disable **CallForwardAlways** as well as set a forwarding number. The settings are stored on the server.

Call Forward Busy

To make this function available on the phone, you must enable the option **SPn Service – Network Provided Services::CallForwardBusy**. This feature is similar to the **CallForwardOnBusy** feature that is available natively on the phone (per service). To use the version provided by the softswitch, disable the corresponding service on the phone to avoid ambiguity.

Use the Net Services app to access the feature on the server. The phone lets you access and set this feature and store feature settings on the server.

Call Forward No Answer

To make this function available on the phone, you must enable the option ***SPn Service – Network Provided Services::CallForwardNoAnswer***. This feature is similar to the CallForwardOnNoAnswer feature that is available natively on the phone (per service). To use the version that provided by the softswitch, disable the corresponding service on the phone to avoid ambiguity.

Use the Net Services app to access the feature on the server. You can enable and disable the feature, set the forwarding number, set the number of rings before forwarding the call and store feature settings on the server.

Anonymous Call

To make this function available on the phone, enable the option ***SPn Service – Network Provided Services::AnonymousCall***. This feature is similar to the AnonymousCall feature that is available natively on the phone (per service). To use the version provided by the softswitch, disable the corresponding service on the phone.

Use the Net Services app to access the feature and define a feature key with the function `Block Caller ID`. Set the Service parameter of the key to the Service Provider service that provides this feature.

Users can enable or disable this feature from the UI, and the server stores the feature settings.

Do Not Disturb

To make this function available on the phone, enable the option ***SPn Service – Network Provided Services::DoNotDisturb***. This feature is similar to the Do Not Disturb feature available natively on the phone (per line). To use the version provided by the softswitch, disable the corresponding feature on the phone to avoid ambiguity.

Use the Net Services app to access the feature and define a feature key with the function `Do Not Disturb`. Set the Service parameter of the key to the SP service that provides this feature to users who can enable or disable this feature from the UI. The server stores the feature settings.

Device Web Page and Configuration Parameter Reference

This reference section lists all the configuration and status parameters. Configuration parameters can be modified. Status parameters are read-only.

Status Parameters

The Status parameters web pages show read-only values for certain parameters on your phone. They include these pages:

- [System Status Web Page](#)
- [Call Status Web Page](#)
- [Call History Web Page](#)
- [SP Services Stats Web Page](#)

System Status Web Page

System Status is available on the phone built-in portal and on the OBiTALK device management portal, under the web page with the same title.

System Status Settings

Parameter	Description	Example Value
<i>WAN Status (DeviceInfo.Network.Status.WAN.)</i>		
AddressingType	Method currently used by the phone to get an IP address assignment.	DHCP
IPAddress	IP address currently assigned to the phone when using static IP addressing.	192.168.15.165
SubnetMask	Subnet mask to use when using static IP addressing.	255.255.255.0
DefaultGateway	Gateway to use when using static IP addressing.	192.168.15.1
DNSServer1	URL for domain name server 1 when using static IP addressing.	8.4.4.4
DNSServer2	URL for domain name server 2 when using static IP addressing.	4.2.2.2
MACAddress	MAC address installed on the phone.	9CADEF90004E

System Status Settings

Parameter	Description	Example Value
LLDP-MEDStatus	Enables LLDP media endpoint discovery for improved network connections.	Enabled
802_1XStatus	802_1X authentication status.	HELD
SWPortLinkStatus	Status of the switch port (the port connecting the phone to the Internet).	Link:Y; Spd:100F; Auto:N
PCPortLinkStatus	Status of the PC port (the port connecting the phone to an auxiliary PC).	Link:N; Spd:100F; Auto:Y
Stats	Ethernet statistics.	P/E: Rx=82276/0; Tx=5538/0

Wi-Fi Status (DeviceInfo.Network.Status.Wi-Fi.)

AddressingType	Method currently used by the phone to get an IP address assignment. Choose from DHCP or Static.	DHCP
IPAddress	IP address currently assigned to the phone when using static IP addressing.	192.168.15.165
SubnetMask	Subnet mask to use when using static IP addressing.	255.255.255.0
DefaultGateway	Gateway to use when using static IP addressing.	192.168.15.1
DNSServer1	URL for domain name server 1 when using static IP addressing.	8.4.4.4
DNSServer2	URL for domain name server 2 when using static IP addressing.	4.2.2.2
MACAddress	MAC address installed on the device.	9CADEF90004E

Product Information (DeviceInfo.)

ModelName	Your phone's model name.	VVX 450
MACAddress	Your phone's MAC address.	9CADEF90004E
SerialNumber	Your phone's serial number.	88H01NA00ZXV
OBiNumber	Your phone's OBi number, a value that uniquely identifies your phone to Polycom and to other OBi devices.	552 860 300
HardwareVersion	Your phone's hardware version.	1.1
SoftwareVersion	Your phone's installed software version. This value changes with a firmware update or downgrade.	6.3.0.15058
SystemTime	Shows the current time on the system.	15:32:35 01/29/2019, Wednesday

System Status Settings

Parameter	Description	Example Value
UpTime	With last Reboot Reason in parentheses. See the Reboot Reason Codes table following this table for the reboot reason codes.	20 Days 5:04:13 (2)
CertificateStatus	Indicates if a device certificate is installed on the phone.	Installed
CustomizationStatus	Indicates if this phone is a customized unit.	Generic

SPn Service Status (VoiceService.1.VoiceProfile.1.Line.n), n = 1 – 6

Status	Registration status of this service. If there are problems with the registration or authentication, the SIP 4xx – 6xx error code and error message are displayed here. This is very useful information for troubleshooting issues with SIP-based services.	Registered (server=192.168.15.118; expire in 39s)
PrimaryProxyServer	IP address of the current Primary Proxy Server if proxy server redundancy is enabled on this service.	10.100.123.234
SecondaryProxyServer	IP address of the current Secondary Proxy Server if proxy server redundancy and secondary registration are both enabled on this service.	10.100.234.123
CallState	Describes the state of an active call on this service.	0 Active Calls

OBiTALK Service Status (VoiceService.1.X_P2P.1.Stats.)

Status	Connection status with the OBiTALK network.	Normal (User Mode)
CallState	Describes the state of an active call on OBiTALK.	0 Active Calls

Reboot Reason Codes

Reboot Reason Codes

Reason Code	Description
0	Reboot on power cycle.
1	Operating system reboot.
2	Reboot after firmware update via provisioning or phone (**6).
3	Reboot after new profile invoked.
4	Reboot after parameter value change or firmware change, and invoked via device web page.
5	Reboot after factory reset.
6	New Profile Invoked AND profile URL changed.
7	Reboot from SIP NOTIFY (reserved).
8	Reboot from phone IVR.

Reboot Reason Codes

Reason Code	Description
9	Reboot from webpage — No change in parameter value or firmware.
10	Reboot during OBiTALK signup.
11	Reboot during OBiTALK signup.
12	Reboot after DHCP server offers IP, GW-IP, and/or Netmask different from what the phone is currently using.
13	Reboot on data networking link reestablishment.
16	Reboot after DHCP RENEW NAK received.
18	Reboot after WAN IP setting(s) changed.
30	Reboot from Phone UI — No change in parameter value.
31	Reboot from Phone UI — After parameter value change.

Call Status Web Page

The Call Status page shows a number of running call statistics and state parameters for each active call currently in progress. Call Status of each call is only available during the lifetime of the call. It is removed as soon as the call is ended.

The following information and statistics are shown for each call:

Call Status Descriptions

Status	Description
Terminal ID	A short name to identify each call terminal: Phone, SP1, SP2, SP3, SP4, SP5, SP6, OBiTALK.
State	Call State.
Peer Name	Call Peer's Name.
Peer Number	Call Peer's Number.
Start Time	Starting time of the call.
Duration	Duration of the call.
Peer RTP Address	The peer address and port where RTP packets are sent to.
Local RTP Address	The local address and port where RTP packets are sent from.
RTP Transport	The transport used for RTP (UDP, TCP, or SSL).
Audio Codec	The audio encoder and decoder being used for this call.
RTP Packetization (ms)	The transmitted and received packet sizes in milliseconds.
RTP Packet Count	Total number of RTP packets transmitted and received thus far.

Call Status Descriptions

Status	Description
RTP Byte Count	Total number of RTP bytes transmitted and received thus far.
Peer Clock Differential Rate	Clock difference between this phone and the peer in ppm (parts per million).
Packets in Jitter Buffer	Number of packets in the Jitter Buffer.
Packets Out-of-Order	Number of received RTP packets that are out of order.
Packets (10ms) Interpolated	Number of frames interpolated.
Packets Lost	Number of incoming RTP packets assumed lost.
Packet Loss Rate	Amount of incoming RTP packets assumed lost rate in percent.
Packet Drop Rate	Amount of incoming RTP packets dropped in percent.
Jitter Buffer Length	Size of the current jitter buffer in milliseconds.
Received Interarrival Jitter	Average measured network jitter in the received direction in milliseconds.
DTMF Digits Received	Number of DTMF digits received.
Jitter Buffer Underruns	Amount of jitter buffer underruns during the call.
Jitter Buffer Overruns	Amount of jitter buffer overruns during the call.
Sequence number discontinuities	Number of times a discontinuity is encountered in the sequence number of the incoming RTP stream.
Round-Trip Propagation Delay	A measurement of the round trip propagation delay of RTP packets.
End System Delay	Contribution to the overall end-to-end system delay by this end-point in milliseconds as defined in RFC3611.
Peer End System Delay	Contribution to the overall end-to-end system delay by the peer end-point in milliseconds as defined in RFC3611.
MOS-LQ	The estimated mean opinion score for listening quality.
MOS-SQ	The estimated mean opinion score for conversational quality.

For each entry on the call status page, the following buttons are available:

- **Remove:** This button is available for all calls. Pressing this button ends that call.
- **Record:** Pressing this button lets you record the current conversation in an audio (.au) file.

Call History Web Page

This page shows the calls in the call history, in order of most recent to oldest.

The following buttons are available:

- **Remove All:** Pressing this button erases the entire call history.
- **Save All:** Pressing this button saves the call history to the `callhistory.xml` file.

The **Call History** page isn't available on the OBiTALK.com website.

SP Services Stats Web Page

This page shows the following information for each SP n service, where $n = 1 - 6$.

SP Services Settings

Parameter	Description	Default Setting
Reset Statistics (VoiceService.1.VoiceProfile.1.Line.n.Stats.), $n = 1 - 6$		
ResetStatistics	Check this option and press Submit to reset the statistics for this SP Service.	NA
RTP Statistics (VoiceService.1.VoiceProfile.1.Line.n.Stats.), $n = 1 - 6$		
PacketsSent	Total RTP packets sent on this line.	NA
PacketsReceived	Total RTP packets received on this line.	NA
BytesSent	RTP payload bytes sent for this line.	NA
BytesReceived	RTP payload bytes received for this line.	NA
PacketsLost	Number of RTP packets lost on this line.	NA
Overruns	Number of times a jitter buffer overrun was received on this line.	NA
Underruns	Number of times a jitter buffer underrun was received on this line.	NA

OBiWiFi Configuration Parameters

The OBiWiFi Configuration parameters web pages show Wi-Fi parameters on your phone. They include these pages:

- [WiFi Settings Web Page](#)
- [WAN Settings Web Page](#)

WiFi Settings Web Page

This page shows all the Wi-Fi setup parameters and status:

Wi-Fi Settings

Parameter	Description	Default Setting
Basic Settings (DeviceInfo.WiFi.Basic.)		
Enable	Enables OBiWiFi feature. You must have an OBiWiFi dongle attached to the phone to use this feature.	True
PreferredAccessPoint	Indicates which access point to use when more than one remembered access point are in range. Select from the list: <ul style="list-style-type: none"> None Access Point 1 Access Point 2 ... Access Point 20. This value is automatically populated with the last access point that the phone user chose to connect explicitly from the native phone web page.	None
ShowAccessPointPassword	Check this box and press Submit to show all the access point passwords in (unmasked) plain text (no reboot required). The passwords are masked again following a reboot of the device.	False
ScanResultWaitDuration	Specifies the WiFi Scan time, from 3 to 30 seconds. This parameter can also be changed from the Phone interface under the Settings/WiFi Settings menu.	6
Internet Settings (DeviceInfo.WiFi.)		
AddressingType	The method to assign an IP address to this interface. Choice of: <ul style="list-style-type: none"> DHCP Static. 	DHCP
IPAddress	The IP address to use if AddressingType = <i>Static</i> .	None
SubnetMask	The subnet mask to use if AddressingType = <i>Static</i> .	None
DefaultGateway	The default gateway to use if AddressingType = <i>Static</i> .	None
DNSServer1	An additional DNS Server to use in addition to the ones received from DHCP.	None
DNSServer2	An additional DNS Server to use in addition to the ones received from DHCP.	None

Wi-Fi Settings

Parameter	Description	Default Setting
802_1XMode	Port-based network access control provides an authentication mechanism to attach to a LAN. Choice of: <ul style="list-style-type: none"> • Disable • MDS • TLS • TTLS-MSCHAPv2 • PEAP-MSCHAPv2 	Disable
802_1XIdentity	User name for 802.1x authentication.	None
802_1XPassword	Password for 802.1x authentication.	None
802_1XAnonymousID	If blank, the anonymous identity isn't used in authentication.	None
802_1XTLSecurityProfile	The TLS platform profile to use for device certification.	1
Access Point <i>n</i> (DeviceInfo.WiFi.AP.n.), <i>n</i> = 1, 2, ..., 20		
SSID	SSID of the access point.	None
Password	Password or pass-phrase based on the authentication method used by the access point. For WPA, the pass-phrase should be no more than 64 characters. For WEP, the password should be in one of the four formats: 10 HEX digits, 26 HEX digits, 5 ASCII characters, or 13 ASCII characters. The HEX digits can be upper or lower case.	None
SecurityEnabled	This read-only parameter indicates if the access point has security enabled or not.	No

WiFi Scan Web Page

The **WiFi Scan** page offers a familiar user interface to let you scan for access points in your vicinity. You can click on the page one of the available access point to connect to. If the access point requires authentication but the phone does not have valid credentials, a page prompts you to enter a password or pass-phrase and press **Connect** to continue.

If your access point does not show up as a listed device on this page, for example, its SSID isn't broadcast. You can enter its SSID and security credentials manually by clicking the **Add a Network** link. The **Manage Networks** link takes you back to the **Wi-Fi Settings** page. The **Scan For Networks** link reloads this page so you can rescan for access points in your vicinity.

System Management Parameters

The System Management parameters web pages show network parameters on your phone. They include these pages:

- [WAN Settings Web Page](#)
- [Auto Provisioning Web Page](#)

- [Device Admin Web Page](#)
- [Device Update Web Page](#)

WAN Settings Web Page

The **WAN Settings** page lists the Ethernet settings for your phone.

WAN Settings

Parameter	Description	Default Setting
<i>Internet Settings (DeviceInfo.WAN.)</i>		
AddressingType	The method used for assigning IP address, subnet mask, default gateway, and so forth, to the device. Choice of: <ul style="list-style-type: none"> • DHCP: IP address, subnet mask, and default gateway are assigned by DHCP Server. • Static: IP address, subnet mask, and default gateway are taken from the manually configured values. 	DHCP
IPAddress	The IP address assigned to the phone when AddressingType is set to <i>Static</i> .	None
SubnetMask	The subnet mask to use when AddressingType is set to <i>Static</i> .	None
DefaultGateway	The default gateway IP address assigned to the phone when AddressingType is set to <i>Static</i> .	None
DNSServer1	IP address of the first DNS server to use, in addition to the ones obtained from the DHCP server when DHCP is also enabled. If AddressingType is set to <i>Static</i> , the phone only uses DNSServer1 and DNSServer2 for DNS lookup. It tries as many as 5 DNS servers when attempting to resolve a domain name. DNSServer1 and DNSServer2 are tried first, whichever is specified, and then the ones obtained from the DHCP server if available.	None
DNSServer2	IP address of the second DNS server to use, in addition to the ones obtained from the DHCP server when DHCP is also enabled. If AddressingType is set to <i>Static</i> , the phone only uses DNSServer1 and DNSServer2 for DNS lookup. It tries as many as 5 DNS servers when attempting to resolve a domain name. DNSServer1 and DNSServer2 are tried first, whichever is specified, and then the ones obtained from the DHCP server if available.	None
VLANEnable	Enables VLAN operation. When enabled, your phone allows only packets with the VLANID to pass, and inserts an 802.1Q header according to VLANID and VLANPriority to all outbound packets.	False
VLANID	Valid range is 0 – 4094 (4095 is reserved). 0 means VLAN is disabled and egress packets aren't tagged by the device. This setting applies to all packets sent by the device.	0

WAN Settings

Parameter	Description	Default Setting
VLANPriority	Valid choices are 0 – 7. This setting applies to all packets sent by the device.	0
VLANDiscovery	Enables taking VLAN setting from DHCP options. Choice of: <ul style="list-style-type: none"> • Disabled • Fixed • Custom 	Disabled
VLANDiscoveryOption	Specifies which DHCP option to use for VLAN discovery.	129
802_1XMode	Authentication is enabled or disabled with the selected mode. Choice of: <ul style="list-style-type: none"> • Disable • MD5 • TLS • TTLS-MSCHAPv2 • PEAP-MSCHAPv2 	Disable
802_1XIdentity	Identity.	None
802_1XPassword	Password for EAP-MD5, EAP-TTLS Private key, and EAP-TTLS/MSCHAPv2 modes.	None
802_1XAnonymousID	Anonymous ID. If empty, anonymous ID isn't used in authentication.	None
802_1XTLSSecurityProfile	Security profile for 802.1x authentication. Choices are 1 or 2.	1
LLDP-MED	Enables LLDP-MED discovery.	True
LLDP-MEDExclusivePeriod	Number of seconds for LLDP-MED discovering Network Policy exclusively before the IP is established according to AddressType . Choice of: <ul style="list-style-type: none"> • 0 • 1 • 2 • 5 • 10 • 20 • 30 • 60 	5
LLDP-MEDAssetID	Customizable AssetID to be included in Inventory Management TLV. The default is the OBi number.	\$DSN

WAN Settings

Parameter	Description	Default Setting
OpenSSLCiphers	OpenSSLCiphers provides support for all SSL/TLS connections. An empty value tells the phone to use the DEFAULT ciphers. A valid value must start with DEFAULT: or HIGH:.	Empty

Switch Port 1

Name	SW	
Speed	<p>Link speed selection mode for the port that connects your phone to the internet. This port must be connected for your phone to work. Choice of:</p> <ul style="list-style-type: none"> • Auto • 100 Full • 100 Half • 10 Full • 10 Half • 1000 Full • Disabled <p>Polycom recommends using only the <code>Auto</code> setting for this parameter.</p>	Auto

Switch Port 2

Name	PC	
Speed	<p>Link speed selection mode for the port that connects your phone to your PC. This port can be left disconnected. Choice of:</p> <ul style="list-style-type: none"> • Auto • 100 Full • 100 Half • 10 Full • 10 Half • 1000 Full • Disabled <p>Polycom recommends using only the <code>Auto</code> or <code>Disabled</code> settings for this parameter.</p>	Auto

Local Time (DevicefoInfo.Time.)

CurrentLocalTime	Current local date and time of the phone (read-only).	
CurrentNTPServer1	Hostname or IP address of the current first NTP server.	None
CurrentNTPServer2	Hostname or IP address of the current second NTP server.	None

Time Service Settings (DevicefoInfo.Time.)

NTPServer1	Hostname or IP address of the first NTP server.	ntp.polycom.com
-------------------	-------------------------------------------------	-----------------

WAN Settings

Parameter	Description	Default Setting
NTPServer2	Hostname or IP address of the second NTP server.	None
LocalTimeZone	<p>Local time zone. Choice of:</p> <ul style="list-style-type: none"> • GMT-12:00 (Int'l Dateline West) • GMT-11:00 (Samoa) • GMT-10:00 (Hawaii) • GMT-09:00 (Alaska) • GMT-08:00 (Pacific Time) • GMT-07:00 (Mountain Time) • GMT-06:00 (Central Time) • GMT-05:00 (Eastern Time) • GMT-04:00 (Atlantic Time) • GMT-03:30 (Newfoundland) • GMT-03:00 (Buenos Aires, Greenland) • GMT-02:00 (Mid-Atlantic) • GMT-01:00 • GMT+00:00 (London, Lisbon) • GMT+01:00 (Rome, Paris, Madrid) • GMT+02:00 (Athens, Cairo) • GMT+03:00 (Moscow, Baghdad) • GMT+04:00 (Abu Dhabi) • GMT+04:30 (Kabul) • GMT+05:00 (Islamabad, Karachi) • GMT+05:30 (New Delhi) • GMT+05:45 (Kathmandu) • GMT+06:00 • GMT+07:00 (Bangkok, Jakarta) • GMT+08:00 (Beijing, HK, Singapore) • GMT+09:00 (Tokyo, Seoul) • GMT+10:00 (Sydney, Guam) • GMT+11:00 (Solomon Is.) • GMT+12:00 (Fiji, Auckland) 	GMT-08:00
DaylightSavingTimeEnable	Enables Daylight Saving Time on the phone.	True
DaylightSavingTimeStart	<p>Daylight Saving Time Start Date. Format: month/day/weekday/hh:mm:ss, where month=1-12, day=±(1-31), weekday=0,1-7 (0=special, 1=Monday, 7=Sunday), hh=0-23, mm=0-59, ss=0-59.</p> <p>If weekday=0, daylight saving starts on the given month/day. Otherwise, it starts on the weekday on or after the given month/day if day > 0, or on the weekday on or before the last day of the given month+day+1. Note that day = -1 is equivalent to the last day of the month.</p> <p>:ss can be omitted if its value is 0. :mm:ss can be omitted if mm and ss are both 0.</p>	3/8/7/2

WAN Settings

Parameter	Description	Default Setting
DaylightSavingTimeEnd	Daylight Saving Time End Date. Same format as Start Date.	11/1/7/2
DaylightSavingTimeDiff	Amount of time to add to current time during Daylight Saving Time. Format: [-]hh:mm:ss. :ss can be omitted if it is 0. :mm:ss can be omitted if both are 0.	1

DHCP Client Settings (X_DHCP)

Hostname	Device Hostname to be sent to server.	\$DM
ExtraOptions	Comma-separated list of extra DHCP options to be requested. Choice of: <ul style="list-style-type: none"> • 42 • 66 • 150 • 159 • 160 • 161 	66, 42

DNS Control (X_DNSControl)

DNSQueryOrder	Controls the order in querying DNS Servers. Choice of: <ul style="list-style-type: none"> • DNS Server1, DNS Server 2, DHCP Offered DNS Servers • DHCP Offered DNS Servers, DNS Server1, DNS Server 2 • DNS Server1, DNS Server 2 	DNS Server1, DNS Server 2, DHCP Offered DNS Servers
DNSQueryDelay	Controls the delay (in seconds) before trying the next DNS server when resolving a domain name. Choice of: 0, 1, 2, 3, 4, and 5.	2

WAN Settings

Parameter	Description	Default Setting
Local DNS Records (X_LocalDNSRec.)		
N where N = 1 – 32	<p>One of 32 Local DNS Records (numbered 1 – 32). Each record is a mini script of either of the following formats:</p> <ul style="list-style-type: none"> Name=A, A, A, . . . Name=R, R, R, . . . <p>where <i>Name</i> represents the domain name to be resolved locally, and has the format <i>prefix+domain</i> (such as <i>machine.sip+obihai.com</i>). Everything after '+' is considered as the domain to be appended to the host field in each <i>R</i> on the right hand side. '+' is optional. If missing, the full domain must be used in every <i>R</i>.</p> <p><i>A</i> represents an <i>A</i> record, which is just an IP address, such as 192.168.12.17.</p> <p><i>R</i> represents an <i>SRV</i> record and has the format: <code>{host:port,pri,wt}</code> where</p> <ul style="list-style-type: none"> <i>host</i> is a hostname with or without domain part (such as <i>xyz</i>, <i>xyz.abc.com</i>). A dot (.) at the end of host indicates it is a complete hostname that does not require the domain to be appended. <i>port</i> is a port number (such as 5060). <i>port</i> is optional. The default to use is based on the protocol (5060 for SIP, 80 for HTTP, and so forth). <i>pri</i> is the priority. Valid value is 0 (highest) – 65535 (lowest). <i>wt</i> is the weight. Valid value is 0 (lowest) – 65535 (highest). <i>wt</i> is optional. 1 is the default if not specified. <i>pri</i> is optional only if <i>wt</i> isn't specified. 1 is the default if not specified. <p>The enclosing curly brackets { } are also optional if there is only one <i>R</i>, or if no comma appears inside the <i>R</i>.</p> <p>Examples:</p> <pre>_sip._udp+obihai.com=abc,xyz,pqr:5080,{mm m,2},{super.abc.com.} abc.obihai.com=192.168.15.118,192.168.15. 108</pre> <p>Note: If the <i>A</i> record of a given hostname cannot be found in any of the local DNS records, the phone attempts to resolve it using external DNS queries. Any change applied to a local DNS record needs a reboot to take effect.</p>	None

Auto Provisioning Web Page

The **Auto Provisioning** web page shows all the parameters related to remote provisioning of the device, as shown in the following table. Provisioning is an important topic for deployment by service providers. Please see the [Polycom OBi Edition Deployment Guide](#) for details on phone provisioning.

Auto Provisioning Settings

Parameter	Description	Default Setting
<i>System Info(X_DeviceManagement.SystemInfo.)</i>		
X_EnableTR69	Enables TR69 provisioning.	False
X_Enable3CXPNP	Enables 3CX Plug-and-Play support. This option lets the phone broadcast a SIP Subscribe to the Event: ua-profile on each reboot. With this enabled, the phone also listens to SIP Notify of the same Event to extract from it the provisioning URL to request configuration, but only if the current ITSP Provisioning – ConfigURL parameter is still set to the default value (default flag on the device native web page is checked).	False
<i>Auto Firmware Update (X_DeviceManagement.FirmwareUpdate.)</i>		
Method	<p>Current operational method of auto firmware updating. Choice of:</p> <ul style="list-style-type: none"> • Disabled = Do not check for firmware upgrade from FirmwareURL. • System Start = Check for firmware upgrade from FirmwareURL just once on system start • Periodically = Check for firmware upgrade from FirmwareURL on system start, and then periodically at the interval specified in the Interval parameter • Time Of Day = Download from ConfigURL once a day at the time specified in the TimeofDay parameter. <p>Note: First firmware upgrade check on system start is performed after a random delay of 0 to 30 seconds.</p>	Disabled
Interval	When Method is set to Periodically , this is the number of seconds between each checking of firmware upgrade check from FirmwareURL . If value is 0, the phone checks once only on system start, the equivalent to setting Method to System Start .	0
TimeofDay	<p>The time during the day at which to execute ConfigURL is the Method is set to Time Of Day. It must be specified in the format: hh:mm[+rr] where:</p> <ul style="list-style-type: none"> • hh: 0 - 23 • mm: 0 - 59 • +rr: (optional) maximum range of random delay in minutes from 0 - 360. If not specified, the default is 30 minutes. Setting rr = 0 removes the random delay. 	00:00+30
RandomDelayRange	The random time delay range (in seconds) inserted before the first attempt only.	30
FirmwareURL	URL of firmware package. URL must include scheme. Supported schemes are http:// and tftp://.	None
TLSSecurityProfile	Security profile when using HTTPS. Choices are 1 or 2.	1

Auto Provisioning Settings

Parameter	Description	Default Setting
DnsLookupType	The type of DNS record to query. Choice of: <ul style="list-style-type: none"> A Record Only SRV Record Only Try Both 	A Record Only
DnsSrvPrefix	Prepends service prefix to domain name when querying DNS. Choice of: <ul style="list-style-type: none"> No Prefix With Prefix Try Both 	No Prefix
Username	Optional user name for authentication if URL scheme is http:// or https://.	None
Password	Optional password for authentication if URL scheme is http:// or https://.	None
RangeDownload	If HTTP/HTTPS is specified in FirmwareURL , your phone tries to download a portion of the image that needs to be updated using HTTP range header, instead of the entire image, unless the server doesn't support the range header.	False
Suspend	Suspends firmware update until canceled.	False
Precheck	Performs firmware precheck.	True

ITSP Provisioning (X_DeviceManagement.ITSPProvisioning.)

Method	Current operational method of Provisioning. Choice of: <ul style="list-style-type: none"> Disabled = Do not download from ConfigURL System Start = Download from ConfigURL just once on system start Periodically = Download from ConfigURL on system start, and then periodically at the interval specified in the Interval parameter Time Of Day = Download from ConfigURL once a day at the time specified in the TimeOfDay parameter <p>Note: First download on system start is performed after a random delay of 30 to 90 seconds. If there is a firmware update scheduled at the beginning. Or a random delay of 10 to 70 seconds.</p>	System Start
Interval	When Method is set to <i>Periodically</i> , this is the number of seconds between download from ConfigURL . If value is 0, the phone downloads once only on system start, the equivalent to setting Method to <i>System Start</i> .	0

Auto Provisioning Settings

Parameter	Description	Default Setting
TimeofDay	The time during the day at which to execute ConfigURL is the Method is set to <code>Time Of Day</code> . It must be specified in the format: <code>hh:mm[+rrr]</code> where: <ul style="list-style-type: none"> • <code>hh</code>: 0 – 23 • <code>mm</code>: 0 – 59 • <code>+rrr</code>: (optional) maximum range of random delay in minutes from 0 – 360. If not specified, the default is 30 minutes. Setting <code>rr = 0</code> removes the random delay. 	00:00+30
ConfigURL	URL of config file.	tftp://\$DHCP OPT66 /\$MAC.xml Note: Default value is blank for OBiTALK provisioning.
DnsLookupType	The type of DNS record to query. Choice of: <ul style="list-style-type: none"> • A Record Only • SRV Record Only • Try Both 	A Record Only
DnsSrvPrefix	Prepends service prefix to domain name when querying DNS. Choice of: <ul style="list-style-type: none"> • No Prefix • With Prefix • Try Both 	No Prefix
Override	Defines the local settings that can be overridden by this parameter. Choice of: <ul style="list-style-type: none"> • All • All except user settings 	All
GPRM0 to GPRM7	Non-volatile generic parameters that can be referenced in other parameters, such as ConfigURL .	None
TPRM0 to TPRM3	Temporary variables used in scripts for ConfigURL . Please refer to the Polycom OBi Edition Deployment Guide for examples on how to use these variables.	None

OBiTALK Provisioning (X_DeviceManagement.Provisioning.)

Auto Provisioning Settings

Parameter	Description	Default Setting
Method	<p>Current operational method of provisioning. Choice of:</p> <ul style="list-style-type: none"> Disabled = Do not download from ConfigURL System Start = Download from ConfigURL just once on system start Periodically = Download from ConfigURL on system start, and then periodically at the interval specified in the Interval parameter Time Of Day = Download from ConfigURL once a day at the time specified in the TimeofDay parameter <p>Note: First download on system start is performed after a random delay of 30 – 90s. If there is a firmware update scheduled at the beginning. Or a random delay of 10- 70s.</p>	Disabled
Interval	<p>When Method is set to Periodically, this is the number of seconds between download from ConfigURL. If value is 0, the phone downloads once only on system start, the equivalent to setting Method to System Start.</p>	0
TimeofDay	<p>The time during the day at which to execute ConfigURL is the Method is set to Time Of Day. It must be specified in the format: hh:mm[+rr] where:</p> <ul style="list-style-type: none"> hh: 0 – 23 mm: 0 – 59 +rr: (optional) maximum range of random delay in minutes from 0 – 360. If not specified, the default is 30 minutes. Setting rr = 0 removes the random delay. 	00:00+30
ConfigURL	URL of config file.	None
DnsLookupType	<p>The type of DNS record to query. Choice of:</p> <ul style="list-style-type: none"> A Record Only SRV Record Only Try Both 	A Record Only
DnsSrvPrefix	<p>Prepends service prefix to domain name when querying DNS. Choice of:</p> <ul style="list-style-type: none"> No Prefix With Prefix Try Both 	No Prefix
Override	<p>Defines the local settings that can be overridden by this parameter. Choice of:</p> <ul style="list-style-type: none"> All All except user settings 	All
GPRM0 to GPRM7	Non-volatile generic parameters that can be referenced in other parameters, such as ConfigURL .	None

Auto Provisioning Settings

Parameter	Description	Default Setting
TPRM0 to TPRM3	Temporary variables used in scripts for ConfigURL . Please refer to the Polycom OBi Edition Deployment Guide for examples on how to use these variables	None

Phone Customization Data Package (X_DeviceManagement.ITSPCustomizationUpdate)

Note that this parameter group name does not end with a period.

Method	<p>Current operational method of provisioning. Choice of:</p> <ul style="list-style-type: none"> Disabled = Do not download from ConfigURL System Start = Download from ConfigURL just once on system start Periodically = Download from ConfigURL on system start, and then periodically at the interval specified in the Interval parameter Time Of Day = Download from ConfigURL once a day at the time specified in the TimeofDay parameter <p>Note: The first download on system start is performed after a random delay of 30 to 90 seconds if there is a firmware update scheduled at the beginning, else a random delay of 10 to 70 seconds.</p>	Disabled
Interval	When Method is set to <code>Periodically</code> , this is the number of seconds between download from ConfigURL . If value is 0, the phone downloads once only on system start, the equivalent to setting Method to <code>System Start</code> .	0
TimeofDay	<p>The time during the day at which to execute ConfigURL is the Method is set to <code>Time Of Day</code>. It must be specified in the format: <code>hh:mm[+rr]</code> where:</p> <ul style="list-style-type: none"> hh: 0 – 23 mm: 0 – 59 +rr: (optional) maximum range of random delay in minutes from 0 – 360. If not specified, the default is 30 minutes. Setting <code>rr = 0</code> removes the random delay. 	00:00+30
DownloadURL	URL of file to be downloaded.	None
MD5Checksum	MD5 checksum of the downloaded file.	None
Incremental	When disabled, old data is erased before new update is installed.	False
DnsLookupType	<p>The type of DNS record to query. Choice of:</p> <ul style="list-style-type: none"> A Record Only SRV Record Only Try Both 	A Record Only

Auto Provisioning Settings

Parameter	Description	Default Setting
DnsSrvPrefix	Prepends service prefix to domain name when querying DNS. Choice of: <ul style="list-style-type: none"> No Prefix With Prefix Try Both 	No Prefix
Username	Optional user name for authentication if URL scheme is http:// or https://	None
Password	Optional password for authentication if URL scheme is http:// or https://.	None
User Defined Macro n (X_DeviceManagement.X_UserDefinedMacro.n.), n = 0, 1, 2, 3 (\$UDM0 – \$UDM3)		
Value	Any plain text, or reference to another parameter's full TR-104 name prepended by a '\$'	None
ExpandIn	A comma-separated list of parameters that are allowed to use this macro expansion. Each parameter must be specified using its full TR-104 name. Specify ANY to allow it in all parameters.	None
SyntaxCheckResult	This read-only status value regards the syntax of the UDM. Pass means that this UDM is valid. Otherwise, it shows the syntax error detected by the phone either in the Value or ExpandIn parameters of the UDM.	Disabled
Statistics Reporting (X_DeviceManagement.Monitoring.)		
SyncCQM	Enables sending anonymous call quality reports to PDMS-SP in JASON format. Caller-ID information is excluded.	Disabled

\$MACRO Expansion Supported by the Phone

Supported Macros

Macro Name	Description	Where It Can Be Used
MAC	Phone MAC address in upper case, such as 9CADEF000000	ANY
MACC	Phone MAC address in upper case with colons, such as 9C:AD:EF:00:00:00	ANY
mac	Phone MAC address lower case, such as 9cadef000000	ANY
macc	Phone MAC address lower case, with colons, such as 9c:ad:ef:00:00:00	ANY

Supported Macros

Macro Name	Description	Where It Can Be Used
FWV	Firmware version, such as 1.0.3.1626	ANY
HWV	Hardware version, such as 2.8	ANY
IPA	Phone current IP address, such as 192.168.15.100	ANY
DM	Phone model name, such as VVX 350	ANY
DMN	Phone model number, such as 350	ANY
OBN	Phone OBi number, such as 200 123 456	ANY
DSN	Phone serial number, such as 88B01NA00000	ANY
GPRM n $n=0-7$	Value of Auto Provisioning::GPRMn	Auto Provisioning::ConfigURL , Auto Firmware Update::FirmwareURL
TPRM n $n=0-3$	Value of Auto Provisioning::TPRMn	Auto Provisioning::ConfigURL , Auto Firmware Update::FirmwareURL
UDM n , $n=0-3$	Value of User-Defined Macro n::Value	Value of User-Defined Macro n::ExpandIn
DHCPOPT66	Value of DHCP option 66 assigned by server	ANY
DHCPOPT150	Value of DHCP option 150 assigned by server	ANY
DHCPOPT159	Value of DHCP option 159 assigned by server	ANY
DHCPOPT160	Value of DHCP option 160 assigned by server	ANY
DHCPOPT161	Value of DHCP option 161 assigned by server	ANY

Device Admin Web Page

The **Device Admin** web page includes the following configuration parameters.

Device Admin Settings

Parameter	Description	Default Setting
Web Server (<i>X_DeviceManagement.WebServer.</i>)		
Port	Web server port number.	80
AdminPassword	Administrator password, case sensitive.	admin

Device Admin Settings

Parameter	Description	Default Setting
UserPassword	User password, case sensitive.	user
LCDScreenShot	Enables creating LCD screenshot from the option in Device Update page.	False
UserMenuDeviceAdmin		True
UserMenuDeviceUpdate		True
Syslog (X_DeviceManagement.Syslog.)		
Server	IP address of the Syslog Server where the phone sends syslog debug messages to. If the value is blank, syslog is disabled.	None
Port	Syslog Server Port Number.	514
Level	Syslog Message Level.	7
TAG	An arbitrary string to add to the beginning of each syslog message.	None
ReportingEnable	Buffers syslog locally and uploads it periodically to a server according to ReportingInterval and ReportingURL .	False
ReportingInterval	Periodic buffered syslog upload interval in seconds. Range is 30 seconds to 120 seconds.	60
ReportingURL	URL to upload locally buffered syslog.	None
ReportingUTCTimeStamp	Include UTC time stamp (TR69 ISO 8601) in each message.	True
LevelDspObi	Log level for OBi/DSP interface. Choice of: <ul style="list-style-type: none"> • Debug • Event 1 • Event 2 • Event 3 • Minor Error • Major Error • Fatal Error 	Event 3
LevelAvm	Log level for AVM. Choice of: <ul style="list-style-type: none"> • Debug • Event 1 • Event 2 • Event 3 • Minor Error • Major Error • Fatal Error 	Minor Error

Device Admin Settings

Parameter	Description	Default Setting
LevelSps	Log level for SPS. Choice of: <ul style="list-style-type: none"> • Debug • Event 1 • Event 2 • Event 3 • Minor Error • Major Error • Fatal Error 	Minor Error
LevelDrvCodec	Log level for DrvCodec. Choice of: <ul style="list-style-type: none"> • Debug • Event 1 • Event 2 • Event 3 • Minor Error • Major Error • Fatal Error 	Minor Error
LevelUsb	Log level for USB. Choice of: <ul style="list-style-type: none"> • Debug • Event 1 • Event 2 • Event 3 • Minor Error • Major Error • Fatal Error 	Minor Error

HTTP Client (X_DeviceManagement.HTTPClient.)

UserAgent	The value of User-Agent request-header field.	\$DM
TimeOut	HTTP request timeout setting in seconds. This value shall be greater than 60 seconds.	600
ProxyServer	Host name or IP address of the HTTP proxy server.	None
ProxyServerPort	Destination port to connect to the HTTP proxy server. Range = [0:65535]. Do not choose a port at random.	80
BypassProxyServerForLocalAddresses	Enables the option to bypass the HTTP Proxy server when requests are sent to addresses in the same subnet.	False
ProxyAuthUsername	User name for proxy authentication.	None
ProxyAuthPassword	Password for proxy authentication.	None

External Port Security (X_DeviceManagement.X_PortSecurity.)

Device Admin Settings

Parameter	Description	Default Setting
USB	Locks the USB port so that the phone does not react to any USB device connected to it.	False
PCPort	Locks the PC port so that the phone does not allow any network traffic in and out of that port.	False

Remote PCAP Server (X_DeviceManagement.X_RPCAPD.)

Enable	Enables PCAP (Packet Capture) Server function on the phone.	False
Ports	PCAP Server port number.	2002
Client	List of clients allowed to connect to this server. An empty list means everyone is allowed.	None

Speed Test (X_DeviceManagement.X_SpeedTest.)

Status	Status of capturing.	Test Result Not Available
Duration	Duration of test for each direction in seconds. The value must be within range of 3 to 30 seconds.	10
On	Enables test.	False

Platform CA n (X_DeviceManagement.PlatformCACert.n.), n = 1, 2

DownloadURL	URL to download certificate.	None
MD5Checksum	MD5 checksum of the certificate file to be downloaded. Failure to provide this causes the phone to try to download the same file on every reboot or restart.	None
CommonName	The Common Name set in the installed certificate. Read-only status field.	None
FingerPrint	SHA1 fingerprint of the installed certificates.	None
Obsolete	When set to true, the certificate is deleted from the phone. Also, the certificate downloading process is ignored.	False

Custom DeviceCert n (X_DeviceManagement.CustomDeviceCert.n.), n = 1, 2

DownloadURL	URL to download certificate.	None
MD5Checksum	MD5 checksum of the downloaded certificate.	None
CommonName	The Common Name set in the installed certificate. Read-only status field.	None
FingerPrint	SHA1 fingerprint of the installed certificates.	None
Obsolete	When set to true, the device is deleted from the phone. Also, the certificate downloading process is ignored.	False

TLSPlatform Profile n (X_DeviceManagement.TLSPlatform.n.), n = 1, 2

Device Admin Settings

Parameter	Description	Default Setting
CipherSuite	The cipher suite to use in a TLS profile (the encryption algorithms to support in establishing a TLS connection according to the TLS profile specification configured on the phone).	None
CACertList	The CA Certificate List to use in a TLS profile. Choice of: <ul style="list-style-type: none"> • Default • Default+P1 • Default+P2 • All • Platform1 • Platform2 • Platform1+2 	Default
DeviceCert	The Device Certificate List to use in a TLS profile. Choice of: <ul style="list-style-type: none"> • Polycom • Custom1 • Custom2 	Polycom

Device Update Web Page

The Device Update web page offers the following functions:

- [Firmware Update](#)
- [Screenshot](#)
- [Screenshot](#)
- [Screenshot](#)
- [Extracting PCAP Capture Result](#)
- [Backup Configuration](#)
- [Restore Configuration](#)
- [Reset Configuration](#)

Firmware Update

You can upgrade the firmware for your phone from the native phone web page. The firmware file with which you want to upgrade the phone must be stored locally on a computer from which you can access with a web browser.

Follow these steps to upgrade:

- 1 Select the **System Management – Device Update** menu on the side panel of the web page.

- 2 Specify the path of the firmware file by clicking the **Select file to upgrade firmware** box or pressing the **Browse** button in the **Firmware Update** section of the page. This presents a file browser window where you can select the firmware file.
- 3 Upon selection of the firmware file, press the **Update** button to start the upgrade process.

The process takes about 30 seconds to complete.



You **MUST NOT** disconnect the power from the phone during this procedure. If the new firmware is upgraded successfully, the phone reboots automatically to start running the new firmware. Otherwise, the page shows an error message explaining why upgrade has failed.

Possible Error Messages on Firmware Update Failure

The following table lists the possible error messages encountered when a firmware upgrade fails.

Error Messages for Firmware Update Failure

Error Message	Description	Suggested Solution
Firmware Package Checksum Error	A corrupted firmware package file has been used for the update.	Check the file and / or re-download the firmware package and try again.
System Is Busy	The phone is busy because one of the phone services is in an active call or device provisioning is in progress.	Try to update again later.
Firmware Is Not Modified	The phone is already running the same firmware as the one selected for update.	No need to upgrade.

Screenshot

Click this **Snap** button to capture a screenshot of the current desktop image on your phone. This saves a file named `snapshot.bmp` to your PC. The default save location is your downloads folder, unless otherwise specified.

Extracting PCAP Capture Result

If PCAP is enabled on your phone, click this **Extract** button to extract the PCAP capture results. This saves a file named `trace.pcap` to your PC. The default save location is your downloads folder, unless otherwise specified.

To enable PCAP, set the `X_DeviceManagement.X_RPCAPD.Enable` parameter to `True`. Before extracting PCAP capture data, set this parameter to `False`.

Backup Configuration

The current configuration of the phone can be backed up and stored as a file in XML format at a user specified location. The default name of the file is `backupxxxxxxxxxxxx.xml`, where `xxxxxxxxxxxx` represents the MAC address of your phone.

When backing up a device's configuration, you can select one the following options before clicking **Backup**.

Backup Options

Option	Description	Default Setting
Incl. Running Status	If checked, the values of all status parameters are included in backup file. Otherwise, status parameters are excluded from the backup.	No
Incl. Default Value	If checked, the default values of parameters are included in the backup file. Otherwise, default values are excluded from the backup.	No
Use OBi Version	If not checked, the backup file uses XML tags that are compliant with the TR-104 standard. Otherwise, the backup file is stored in an OBi proprietary format where the XML tags aren't compliant with TR-104, but the file size is smaller and the file is more readable.	No

When the file browser window opens, you can change the filename and choose the location to save the backup file.



Different web browsers may handle this differently. If the operation is blocked due to the security setting of the web browser, you should change the security setting temporarily to allow this operation to complete.

Restore Configuration

When restoring the configuration to a previous backup copy, you need to specify the backup file you want to restore to by clicking **Browse** in the **Restore Configuration** section of the web page. Then, select the **Restore** button to start the process. The phone reboots automatically after the restoration is complete.



All passwords and PINs are excluded from the backup file. Hence, they aren't available to restore. Call history is excluded from the backup, but can be saved as an XML formatted file separately from the Call History web page.

Reset Configuration

The **Reset Configuration** function resets the phone to its factory default condition. Call history and various statistical information is removed at the same time. Resetting the device configuration should be used with extreme caution as the operation cannot be undone.

Reset Configuration Options

Option	Description	Default Setting
User Data	When checked, clicking Reset restores all your user data to the factory defaults.	Yes
Voice Configuration	When checked, clicking Reset restores all your voice configuration parameters to the factory defaults.	Yes

To do this, click **Reset** in the **Reset Configuration** section. A confirmation window pops up. On confirmation, the phone resets its configuration to its factory default condition. The phone reboots automatically when factory reset completes.

Service Providers Parameters

The Service Providers parameters web pages show parameters for the provisioned service providers on your phone. They include these sets of pages for each of the six ITSP Profiles A through F :

- [General Web Page](#)
- [SIP Web Page](#)
- [RTP Web Page](#)

ITSP Profile X (X = A, B, C, D, E, F)

ITSP profiles represent profiles for the service providers. Voice service profiles, described in the next set of web pages, represent the profiles that bind your phone to the service providers.

General Web Page

The following configuration parameters are available on this page.

General Settings

Parameter	Description	Default Setting
<i>General ITSP Settings (VoiceService.1.VoiceProfile.k.), k = 1 – 6 for X = A – F, respectively</i>		
Name	Human-readable string to identify the profile instance. Maximum length = 127 characters.	None
SignalingProtocol	Choose among the following list of signaling protocols for this ITSP: <ul style="list-style-type: none"> • SIP 	SIP
DTMFMethod	Method to pass DTMF digits to peer device. Choice of: <ul style="list-style-type: none"> • Inband - DTMF tone are sent as inband audio signal • RFC2833 - DTMF tone events are relayed per RFC2833 • SIPInfo - DTMF tones are relayed with SIP INFO request • Auto - Method to use based on call setup negotiation (either Inband or RFC2833 can be negotiated) 	Auto
InbandDTMFVolume	DTMF tone volume when sending inband DTMF. Valid values are –24dB to 0dB in 3dB steps.	–15dB
X_UseFixedDurationRFC2833DTMF	When relaying DTMF digit events on this trunk using RFC2833, the RFC2833 RTP packets normally keep streaming for as long as the digit is pressed. With this option set to True, the phone sends only one RTP digit event packet with a fixed duration of 150 ms regardless how long the digit has been pressed.	False
DigitMap	A digit map to restrict the numbers that be dialed or called with this service. See the Call Routing and Digit Map Configuration sections for a description of digit map syntaxes. Maximum Length = 511 characters.	(1xxxxxxxxxx <1>[2-9]xxxxx xxxx 011xx. xx.)

General Settings

Parameter	Description	Default Setting
STUNEnable	Enables phone to send a STUN binding request for its RTP port prior to every call.	False
STUNServer	IP address of domain name of the STUN Server to use.	None
X_STUNServerPort	UDP listen port of the STUN Server.	3478
X_ICEEnable	Enables phone to use ICE algorithm to find the best peer RTP address to forward RTP traffic for every call.	False
X_EarlyICEEnable	Enables starting ICE upon 18x response when possible for outbound calls.	False
X_EarlyICEEnableIn	Enables sending 183 with SDP and start ICE on INVITE when possible for inbound calls.	False
X_ICEExpires	ICE Expires in milliseconds for outgoing calls.	2000
X_ICEExpiresIn	ICE Expires in milliseconds for incoming calls.	10
X_SymmetricRTPEnable	Enables phone to apply symmetric RTP behavior on every call. That is, send RTP to peer at the address where incoming RTP packets are received from.	False
<i>Service Provider Info (VoiceService.1.VoiceProfile.k.ServiceProviderInfo.), k = 1 – 6 for X = A – F, respectively</i>		
Name	Human-readable string identifying this service provider. Maximum length = 127 characters.	None
URL	Website of this service provider. Maximum length = 127 characters.	None
ContactPhoneNumber	Phone number to contact this service provider. Maximum length = 31 characters.	None
EmailAddress	Email address to contact this service provider. Maximum length = 127 characters.	None

SIP Web Page

The following configuration parameters are available on this page.

SIP Settings

Parameter	Description	Default Setting
<i>SIP (VoiceService.1.VoiceProfile.k.SIP), k = 1 – 6 for X = A – F, respectively</i>		
ProxyServer	Host name or IP address of the SIP proxy server.	None
ProxyServerPort	Destination port to connect to the SIP server.	5060

SIP Settings

Parameter	Description	Default Setting
ProxyServerTransport	Transport protocol to connect to SIP server. Choice of: <ul style="list-style-type: none"> • UDP • TCP • TLS 	UDP
RegistrarServer	Hostname or IP address of the SIP registrar. If a value is specified, the phone sends REGISTER to the given server. Otherwise, REGISTER is sent to ProxyServer	None
RegistrarServerPort	Destination port to connect to SIP registrar.	5060
X_XsiServer	Host name or IP address of the BroadSoft XSI Application Server.	False
X_XsiServerPort	Listening port of the BroadSoft XSI Application Server. If not specified or 0, the default port is used (80 for HPTP and 443 for HTTPS). Range = [0 : 65535].	0
X_XsiServerScheme	Scheme to access the BroadSoft XSI Application Server. Choice of: <ul style="list-style-type: none"> • HTTP • HTTPS 	HTTP
UserAgentDomain	CPE domain string. If empty, the phone uses ProxyServer as its own domain to form its AOR (Address Of Record) or Public Address when constructing SIP messages (for example, in the FROM header of outbound SIP Requests). Note: If SPx Service::URI is specified, additional rules applied in forming the AOR. See description of the URI parameter for more details and examples.	False
OutboundProxy	Host name or IP address of the outbound proxy. Outbound proxying is disabled when this parameter is blank.	None
OutboundProxyPort	Destination port to be used in connecting to the outbound proxy.	5060
X_OutboundProxyTransport	A different SIP transport can be used by the OutboundProxy . Choice of: <ul style="list-style-type: none"> • UDP • TCP • TLS • Follow ProxyServerTransport 	Follow ProxyServerTransport
X_UserAgentContactFollowProxyServerTransport	If enabled, the user agent should use a Contact and Via transport that agrees with ProxyServerTransport .	False
X_BypassOutboundProxyInCall	Bypasses the outbound proxy after initial call setup.	False
RegistrationPeriod	Nominal interval between device register in seconds.	60

SIP Settings

Parameter	Description	Default Setting
X_RegistrationMargin	Specifies the margin to renew SIP registration before it expires. The default is to renew at half-time before the next expiration if the expires value is less than 1200 seconds, or 600 seconds before expiration otherwise. You can specify here the number of seconds before expiration to renew registration explicitly (such as 10 or 20), or as a fraction of the current register expires value (such as 0.1 or 0.25).	None
TimerT1	Value of SIP timer T1 in ms.	500
TimerT2	Value of SIP timer T2 in ms.	4000
TimerT4	Value of SIP timer T4 in ms.	5000
TimerA	Value of SIP timer A in ms.	500
TimerB	Value of SIP timer B in ms.	32000
TimerD	Value of SIP timer D in ms.	32000
TimerE	Value of SIP timer E in ms.	500
TimerF	Value of SIP timer F in ms.	32000
TimerG	Value of SIP timer G in ms.	500
TimerH	Value of SIP timer H in ms.	32000
TimerI	Value of SIP timer I in ms.	5000
TimerJ	Value of SIP timer J in ms.	32000
TimerK	Value of SIP timer K in ms.	5000
InviteExpires	Invite request Expires header value in seconds.	60
ReInviteExpires	Re-invite Expires header value in seconds.	10
RegistersMinExpires	Register Min-Expires header value in seconds (not used at the moment).	15
RegisterRetryInterval	Register retry interval in seconds.	30
X_RegisterRetryResponseCodes	A set of rules to control how many seconds to wait before retrying register after a specific failure response. The rules are specified with a digit map string. The value after the letter <i>w</i> designates the number of seconds to wait. A range can be specified with two numbers with a – in between such that a random delay within that range is used.	(<40[17]:w120> <40[34]:w120> <99[01]:w120-200> [4-9]xx)
X_RegisterIncludeInstance	Enables or Disables inclusion of instance parameter in the Contact of REGISTER requests.	True
DSCPMark	Diffserv code outgoing SIP packets. Valid range: [0-63].	26

SIP Settings

Parameter	Description	Default Setting
X_SpoofCallerID	Allows outbound Caller ID spoofing. If set to <code>True</code> , the phone attempts to set the caller ID name and user ID field in the FROM header to that of a remote caller in the case of a bridged call (from another trunk, such as PSTN Line or another SP Service). Otherwise, the phone always provides its own account information to form the FROM header. Note that most service providers don't allow originating a call if the FROM header field doesn't match the account credentials. Enable this option only if you are sure that the service provider allows it.	False
X_UseRefer	Enables using SIP REFER for call transfer. If disabled, the phone bridges the call instead when performing a call transfer, which consumes some resources on the phone.	True
X_ReferAOR	Enables using the target's AOR in Refer-To header of SIP REFER. If disabled, the target's Contact is used instead.	True
X_HoldReferee	Holds the Referee before a blind transfer if the call isn't placed on hold. This allows reconnecting with the Referee if the blind call transfer fails.	False
X_Use302ToCallForward	Enables the 302 response to INVITE for call forward. If disabled, the phone bridges the call legs instead when forwarding a call, and consumes some resources on the device.	True
X_UserAgentName	If a value is specified, the phone includes a User-Agent header in all SIP Requests or a Server header in all SIP responses that contains exactly the given value.	OBIHAI / \${DM} - \${FWV}
X_ProcessDateHeader	Enables the phone to decode the DATE header sent by the ITSP in a 200 response to its REGISTER. The DATE header specifies the current GMT time and the phone can use this to adjust its local time and date without relying on NTP.	True
X_InsertRemotePartyID	Enables the phone to include a Remote-Party-ID header in its outbound SIP INVITE to indicate to the ITSP the caller's preferred privacy setting (either full or none).	True
X_EnforcePAssertedIdentity	Takes Caller ID from P-Asserted-Identity header only.	False
X_InsertPPreferredIdentity	Inserts P-Preferred-Identity header in all outbound INVITE.	False
X_InsertPAccessNetworkInfo	Inserts P-Access-Network-Info header in REGISTER and INVITE requests.	False
X_InsertPrivacyHdr	Inserts a Privacy:id header in INVITE for anonymous calls.	False

SIP Settings

Parameter	Description	Default Setting
X_UseAnonymousFROM	Enables using <code>sip:anonymous@localhost</code> in FROM header of SIP INVITE when attempting to make an anonymous call.	False
X_SessionRefresh	Enables session refresh signaling (with SIP Re-INVITE) during a connected call. This allows the phone to detect if the connection with the peer is broken abnormally so it can release the call. Disable this option if the ITSP does not support Re-INVITE sent from the client device.	False
X_SessionTimer	Enables standard session timer behavior based on RFC4028.	False
X_SessionExpires	Session Expires before value. If session refresh is enabled, the phone refreshes half-time before the session expires.	20
X_AccessList	A comma-separated list of IP addresses such that the phone only accepts SIP requests coming from one of the given addresses. If the list is empty, the phone accepts SIP requests from any IP address.	False
X_InsertRTPStats	Enables the phone to include a X-RTP-Stat header in a BYE request or 200 response to BYE request at the end of an established call. This header contains a summary of RTP statistics collected during the call.	True
X_MWISubscribe	Enables the phone to SUBSCRIBE to the message-summary event package to support MWI and VMWI service. Note that the phone handles NOTIFY of this event package regardless of whether MWISubscribe is enabled.	False
X_MWISubscribeURI	Blank implies to use the same URL as REGISTER for the TO and FROM header as well as the Request-URI Otherwise, if the URI does not contain '@', it is user as the user ID field in TO/FROM header as well as the Request-URI, which are otherwise same as REGISTER If the URI contains '@', it is used in the TO and FROM header as the phone forms the Request-URI of SUBSCRIBE the same way as the TO header, with an additional port number.	False
X_MWISubscribeExpires	Periodic interval to renew SUBSCRIBE.	3600
X_RegSubscribe	Enables subscription to the reg event package.	False
X_RegSubscribeExpires	Reg event subscription expires value.	3761
X_NoNonceAuth	Enables sending Authorization header without being challenged first.	False

SIP Settings

Parameter	Description	Default Setting
X_ProxyServerRedundancy	Enables proxy redundancy feature on the phone. To use this feature, phone registration must be enabled and the SIP Registration Server or Outbound Proxy Server must be configured as a domain name.	False
X_SecondaryRegistration	Enables phone to register with a secondary server in addition to the primary server. X_ProxyServerRedundancy must be enabled for this parameter to take effect	False
X_CheckPrimaryFallbackInterval	Interval in seconds at which the phone should check the primary fallback list of candidate servers.	60
X_CheckSecondaryFallbackInterval	Interval in seconds at which the phone should check the secondary fallback list of candidate servers.	60
X_ProxyFailoverResponseCodes	A list of failure response codes specified in the form of a digit map string to trigger proxy failover. If only one digit map is specified, it applies to REGISTER and INVITE requests. If two digit maps are provided (separated by a comma), the first one applies to REGISTER and the second to INVITE.	([5-9]xx)
X_InviteFailoverWaitRegisterTimer	Maximum time (in milliseconds) to wait for successful register failover to retry INVITE after a failure.	32000
X_ProxyRequire	If this parameter isn't blank, the phone includes a Proxy-Require header stating the value of this parameter in all SIP requests sent to the ITSP.	None
X_MaxForward	Value for the Max-Forward header in all SIP requests sent by the phone.	70
X_AcceptLanguage	If this parameter isn't blank, the phone includes an Accept-Language header stating the value of this parameter in all SIP requests sent to the ITSP.	None
X_DnsSrv	Enables DNS SRV Lookup for the Proxy Server or the Outbound Proxy Server.	False
X_DnsSrvAutoPrefix	Lets the phone automatically prepend a standard prefix to the domain name when querying DNS Server to resolve the ProxyServer or OutboundProxy name as a SRV record. The standard prefix is: <ul style="list-style-type: none"> • <code>_sip._udp.</code> for SIP over UDP • <code>_sip._tcp.</code> For SIP over TCP • <code>_sip._tls.</code> for SIP over TLS 	False
X_DnsNAPTR	Enables DNS/NAPTR lookup for the Proxy Server or for the Outbound Proxy Server.	False
X_Support100rel	Turns on support for RFC3262 (reliable provisional SIP responses). If enabled, the phone announces this support in a SIP Supported header, and requires a caller to use this option if the caller also supports this feature.	False

SIP Settings

Parameter	Description	Default Setting
X_UserEqPhone	Enables inserting <code>user=phone</code> parameter in INVITE Request-URI.	False
X_UseTelURI	Enables using <code>tel:</code> in outbound SIP Request-URI and TO-URL.	False
X_DiscoverPublicAddress	Enables the phone to use the public IP address and port it has discovered as its SIP Contact address.	False
X_UsePublicAddressInVia	Use the public IP address and port discovered by the phone as the address in VIA header.	False
X_PublicIPAddress	A static public IPv4 address, if specified, is used by the phone to form its SIP Contact address.	None
X_UseRport	Enables the phone to insert a blank <code>rport</code> parameter in the VIA header for outbound SIP messages. This option should be turned off if you are using port forwarding on the external router to route inbound SIP messages to the phone.	True
X_UseCompactHeader	Enables using compact format SIP headers.	False
X_IncludeMessageHash	Includes a MD5 hash of all the SIP headers in an X-MD5-Hash header. A hash of the SDP is also included in an x-md5-hash SDP attribute.	False
X_EchoServer	A server that can echo back SIP messages to the device.	None
X_EchoServerPort	The echo server listening port.	5060
X_EnableRFC2543CallHold	Enables the phone to recognize call hold signaling as used in RFC2543.	False
X_VerifyServerDomain	Enables verification of server domain against its certificate on a SSL/TLS connection.	False
X_RejectKeyResponseCode	SIP Response Code and Phrase to inbound INVITE when you press the Reject key.	None
X_SupportOutbound	Enables support for SIP Outbound (RFC 5626).	False
X_SupportPath	Enables support for the SIP header path.	False
X_SupportServiceRoute	Enables support for the SIP header Service-Route.	False
X_SupportPAssociatedURI	Enables support for the SIP header P-Associated-URI.	False
X_Sticky18x	Enables the behavior of ignoring further 18x responses without SDP upon receiving the first 18x with SDP to INVITE in the context of playing a local ringback tone.	False

SIP Settings

Parameter	Description	Default Setting
<i>Feature Configuration (VoiceService.1.VoiceProfile.k.SIP), k = 1 – 6 for X = A – F, respectively</i>		
X_CallParkMethod	Select the method to use to park a call. Choice of: <ul style="list-style-type: none"> Feature Code REFER 	Feature Code
X_DirectedCallPickup Method	Select the method to use for directed call pickup. Choice of: <ul style="list-style-type: none"> Feature Code INVITE+Replaces 	Feature Code
X_AutoAnswerMethod	Method to signal to the called phone to auto-answer the call. Choice of: <ul style="list-style-type: none"> Call-Info: Inserts answer-after=0 parameter in a Call-Info header in the INVITE request. Alert-Info: Inserts info=alert-autoanswer;delay=0 parameter in an Alert-Info header in the INVITE request. 	Call-Info
X_ShareLineMethod	Select the signaling method for share line operation. Choice of: <ul style="list-style-type: none"> call-info dialog;sla dialog;ma 	call-info
X_ShareLineStateAgent	A State Agent user ID that the phone initiates a subscription to, if specified.	False
X_DialogStateUpdate Filter	Comma-separated list of unwanted states to not notify the state agent.	ringing,early,proceeding
X_DialogSubscribeExpires	BLA Dialog Subscription Renewal interval in seconds. Set to 0 to disable subscription renewal.	3600
X_IgnoreDialogInfoVersion	Ignore the version attribute in dialog-info tag to work around implementation that does not update version.	False
X_BLFSubscribeExpires	BLF Subscription Expires value, in seconds. Set to 0 to disable subscription renewal.	3600
X_CallInfoSubscribeExpires	CallInfo (SCA) Subscription Renewal interval in seconds. Set to 0 to disable subscription renewal.	3600
X_BWCallParkSubscribeExpires	x-broadworks-callpark Subscription Renewal interval in seconds. Set to 0 to disable subscription renewal.	3600
X_BWCallCenterSubscribeExpires	x-broadworks-callcenter Subscription Renewal interval in seconds. Set to 0 to disable subscription renewal.	3600
X_BWHotelingSubscribeExpires	x-broadworks-hoteling Subscription Renewal interval in seconds. Set X_KeepAliveMsgType to 0 to disable subscription renewal.	3600
X_ASFeatureEventSubscribeExpires	(BroadSoft) as-feature-event Subscription Renewal interval in seconds. Set to 0 to disable subscription renewal.	3600

SIP Settings

Parameter	Description	Default Setting
X_LineSeizeSubscribe Expires	Line-seize event subscription renewal interval in seconds.	15
<i>Feature Codes (VoiceService.1.VoiceProfile.k.X_FeatureCode.), k = 1 – 6 for X = A – F, respectively</i>		
DirectedCallPickup	Code to invoke Directed Call Pickup feature on the ITSP.	*97
CallPickup	Code to retrieve a parked call from the ITSP.	*88
BargeIn	Code to barge-in an active call on another extension.	*33
Whisper	Code to whisper (or coach) another extension on an active call.	*48
Monitor	Code to monitor (or coach) another extension on an active call.	*47
Park	Code to park a call on the ITSP.	*68

RTP Web Page

The following configuration parameters are available on this page.

RTP Settings

Parameter	Description	Default Setting
<i>RTP (VoiceService.1.VoiceProfile.k.RTP), k = 1 – 6 for X = A – F, respectively</i>		
LocalPortMin	Base of port range for tx/rx RTP with this SP.	16600
LocalPortMax	Top of port range for tx/rx RTP with this SP.	16798
KeepAliveInterval	Interval in seconds between sending keep alive packet on an RTP channel that is currently in idle (due to call hold for instance). RTP keepalive is disabled if the value of this parameter is set to 0.	0
DSCPMark	Diffserv code for outgoing RTP packets with this SP.	0
X_RefreshSession	Allow incoming RTP packets to refresh session.	False
X_SymmetricMedia	If incoming payload type changes unannounced, after 10 packets with the new payload type, decoding switches to the new format. If symmetric media is enabled, outgoing packets are also in the new format.	False
<i>RTCP (VoiceService.1.VoiceProfile.2.RTP.RTCP), k = 1 – 6 for X = A – F, respectively</i>		
Enable	Enables RTCP operation.	False
TxRepeatInterval	Interval in milliseconds between RTCP transmissions.	10000

RTP Settings

Parameter	Description	Default Setting
LocalCName	The local CNAME to use in RTCP message. By default, the phone uses <code>account-userid@local-ip-address</code> as the local CNAME.	None
X_RTCPMux	Enables RTCP-Mux operation (send and receive RTCP on the same local port as the corresponding RTP).	False
X_VqPublishEnable	Enables Voice Quality (VQ) report sent to the proxy server using Publish method.	False
X_VqPublishUrl	A user name or URL to send VQ report using Publish method.	False
X_VqPublishInterval	Interval in seconds between VQ reports. 0 or {blank} disables periodic reports.	0
<i>Jitter Buffer (VoiceService.1.VoiceProfile.k.RTP.JIB.), k = 1 – 6 for X = A – F, respectively</i>		
Adaptive	Enables jitter buffer adaptation.	True
MaximumSize	Maximum jitter buffer size in ms.	250
SetPoint	Initial play out delay in ms.	60
Target	Target play out delay in ms.	20
AdaptationSlope	Maximum adaptation slope in samples per 10 ms.	16

Voice Services

The Voice Services parameters web pages show parameters for the voice services that are bound to the service providers for your phone. They include these sets of pages:

- [SPn Service Web Page \(n = 1, 2, 3, 4, 5, 6\)](#)
- [OBiTALK Service Web Page](#)
- [Auto Attendant Web Page](#)
- [Gateways and Trunk Groups Web Page](#)

SPn Service Web Page (n = 1, 2, 3, 4, 5, 6)

The following configuration parameters are available on this page.

SPn Services Settings

Parameter	Description	Default Setting
<i>SPn Service (VoiceService.1.VoiceProfile.1.Line.n.), n = 1 – 6</i>		
Enable	Enables this line.	True
X_DisplayLabel	A label that represents this service on the phone screen. Special characters must include a closing character. For example, an open curly bracket must also include a closing curly bracket.	None
X_DisplayNumber	A number that represents this service on the phone screen. Special characters must include a closing character. For example, an open curly bracket must also include a closing curly bracket.	None
X_ServProvProfile	Select a Service Provider profile for this service. Choices are A, B, C, D, E, or F.	A
X_RingProfile	Selects a Ring Profile to ring the phone for incoming calls on this service that are routed to it. The ringing pattern is taken from the given profile. Choices are A or B.	A
X_CodecProfile	Selects a Codec Profile for all calls on this service. Choices are A or B.	A
X_InboundCallRoute	Routing rule for directing incoming calls on this service. The default rule is to send all incoming calls to the phone. See the Call Routing and Digit Map Configuration sections for a description of the syntaxes to specify this parameter.	ph
X_RegisterEnable	Enables registration for this line. If set to True, the phone sends periodic SIP REGISTER to the service provider according to the settings in the ITSP Profile. Otherwise, the phone doesn't send any SIP REGISTER for the service.	True
X_AcceptSipFromRegistrar Only	Accept SIP packets coming from the current registrar IP address only.	False
X_NoRegNoCall	Enables this option to disallow incoming and outgoing calls if registration with the service provider isn't successful.	False
X_KeepAliveEnable	Enables sending keep alive message. If set to True, the phone sends periodic keep-alive messages to the destination specified in the X_KeepAliveServer and X_KeepAliveServerPort parameters, at the interval specified in the X_KeepAliveExpires parameter. The content of this message is the ASCII string <code>keep-alive\r\n</code> .	False

SPn Services Settings

Parameter	Description	Default Setting
X_KeepAliveExpires	Keep alive period in seconds.	15
X_KeepAliveExpiresFollow Server	Follows the keep-alive expires value from server in the VIA-header keep parameter.	True
X_KeepAliveMsgType	The type of keep alive messages to send out periodically if keep-alive is enabled. Choice of: <ul style="list-style-type: none"> <code>keep-alive</code>: The string <code>keep-alive</code> <code>empty</code>: A blank line <code>stun</code>: A standard STUN binding request. The phone uses the binding response to form its contact address for REGISTRATION <code>custom</code>: use the value of X_CustomKeepAliveMsg <code>options</code>: A SIP OPTIONS message <code>notify</code>: A SIP NOTIFY message 	<code>keep-alive</code>
X_CustomKeepAliveMsg	Defines the custom message to be used when X_KeepAliveMsgType is <code>custom</code> . The value has the following format: <code>mtd=NOTIFY;event=<whatever>;user=<anyone></code> Where <ul style="list-style-type: none"> <code>NOTIFY</code> can be replaced by any other SIP method, such as <code>PING</code> <code>event</code> is optional and is only applicable if method is <code>NOTIFY</code>. If <code>event</code> isn't specified, the 'keep-alive' event is used with <code>NOTIFY</code> <code>user</code> is optional. If not specified, the request-uri won't have a user ID, and the TO header field uses the same user ID as the FROM header, which is the local account user ID. If <code>user</code> is specified, it's used as the user ID in the Request-URI and TO header. SIP messages for keep-alive are sent only once without retransmission. Responses to the SIP messages are ignored by the phone.	None
X_UserAgentPort	UDP port where the phone sends and listens for SIP messages.	5060
X_UserAgentPorts	A comma-separated list of as many as 10 alternative user agent ports to use when there is no response received from the SIP Registrar.	None
DirectoryNumber	Directory number associated with this service.	None
X_DefaultRing	Default ring pattern number to ring the phone for incoming calls on this trunk that are routed to the phone according to the InboundCallRoute of this service. The ring pattern is taken from the selected Ring Profile. Valid choices are 1–10.	1
X_RingtoneFile	Default ringtone to use for incoming calls on this trunk.	None

SPn Services Settings

Parameter	Description	Default Setting
X_SatelliteMode	Enables satellite mode on this trunk. In this mode, you must explicitly sign on (using * code) to receive phone calls on this trunk. The SIP REGISTER sent by the phone to the ITSP on this trunk indicates if you want to sign on (and therefore takes over the incoming calls for this account). This feature is only applicable if the service is provided by an OBiPLUS system.	False
X_AcceptResync	Controls whether to accept a SIP NOTIFY request with event=resync to trigger a reboot of the phone so it can download new firmware or configuration upon boot up. Choice of: <ul style="list-style-type: none"> no (don't accept resync trigger) yes with authentication (accept after challenging the sender) yes without authentication (accept without challenging the sender) 	yes without authentication
Debug Options (VoiceService.1.VoiceProfile.1.Line.n.), n = 1 – 6		
X_SipDebugOption	Enables sending of SIP signaling debug information to the syslog server (if one is configured on the device). Choice of: <ul style="list-style-type: none"> Disable (do not send SIP signaling debug information) Log All Messages (log all messages) Log All Except REGISTER Messages 	Disable
X_SipDebugExclusion	Comma-separated list of SIP methods to exclude from the debug log.	None
Credentials (VoiceService.1.VoiceProfile.1.Line.n.SIP), n = 1 – 6		
AuthUserName	The User ID to authenticate to a SIP UAS (User Agent Server) when an outbound SIP request sent by the phone is challenged by the UAS with a 401 or 407 response.	None
AuthPassword	The password (corresponding to AuthUserName) to authenticate to a SIP UAS (User Agent Server) when an outbound SIP request sent by the phone is challenged by the UAS with a 401 or 407 response.	None
DTLSPeerCertAltName	Alternate name for verifying peer's DTLS certificate.	None

SPn Services Settings

Parameter	Description	Default Setting
URI	<p>This parameter affects the way the AOR is formed by the phone in outbound SIP Requests. The AOR has the format: <code>user@domain</code>.</p> <p>If the value of URI is empty, the phone gets the user portion of its AOR from the AuthUserName, and the domain portion the value of ITSP Profile's UserAgentDomain if it isn't empty, or that of the ProxyServer otherwise.</p> <p>If the value of URI isn't empty and does not contain "@", it is used as the user portion of the AOR while the domain portion is formed the usual way.</p> <p>If the value of URI contains @, it is interpreted as a full AOR and phone takes it as the AOR as is.</p> <p>Some examples:</p> <ul style="list-style-type: none"> Let ProxyServer = <code>sip.myitsp.com</code>, AuthUserName = <code>4089991123</code>, URI=[empty], UserAgentDomain=[empty], then AOR = <code>4089991123@sip.myitsp.com</code> Change UserAgentDomain to <code>users.myitsp.com</code>, then AOR = <code>4089991123@users.myitsp.com</code> Change URI to <code>bobdylan</code>, then AOR = <code>bobdylan@users.myitsp.com</code> Change URI to <code>bobdylan@superusers.myitsp.com</code>, then AOR = <code>bobdylan@superusers.myitsp.com</code> <p>Note: In all cases, the phone uses AuthUserName and AuthUserPassword to compute authorization if challenged by a 401 or 407 response.</p>	None
X_MyExtension	An extension assigned to this line for inter-office calling and feature invocation.	None
X_XsiUserName	User name to authenticate a BroadSoft XSI Application Server. If not specified, <code>sip-userid@sip-proxy</code> is used.	None
X_XsiPassword	Password to authenticate a BroadSoft XSI Application Server. If not specified, the SIP password is used.	None
X_XmppDomain	Domain name of a BroadSoft XMPP Service.	None
X_XmppDomainUseSRV	Enables using SRV record to resolve XMPP domain.	False
X_XmppUserName	User name to authenticate to the XMPP Server.	None
X_XmppAuthWithUserName	Use the XmppUserName as-is for authentication without appending the @domain.	False
X_XmppPassword	Password to authenticate a BroadSoft XMPP Application Server.	None

SPn Services Settings

Parameter	Description	Default Setting
X_ContactUserID	An alternative user ID to be used in Contact header. Enter <code>Random</code> to let the phone generate a random one.	None
X_EnforceRequestUserID	Enforce incoming INVITE request user ID to match AuthUserName or X_ContactUserID .	False
Share Line Features (VoiceService.1.VoiceProfile.1.Line.n.ShareLineFeatures.), n = 1 – 6		
X_ShareLine	Check this option if this is a shared line.	False
X_ShareLineUserID	The (third-party) user ID to register for this line, if different from the account user ID.	None
X_ShareLineBargeIn	Check this if the share line supports barge-in function.	False
X_ShareLineWhisper	Check this if the share line supports whisper function.	False
X_ShareLineMonitor	Check this if the share line supports monitor function.	False
X_ShareLinePrivateHold	Check this if the share line supports private hold function.	False
X_ShareLineCallPickup	Check this if the share line supports call pickup function.	False
BLF Features (VoiceService.1.VoiceProfile.1.Line.n.BLFFeatures.), n = 1 – 6		
X_BifBargeIn	Check this if the BLF feature supports barge-in function.	False
X_BifCallPickup	Check this if the BLF feature supports call pickup function.	False
X_BifDirectedCallPickup	Check this if the BLF feature supports directed call pickup function.	False
X_BifWhisper	Check this if the BLF feature supports whisper (or coach) function.	False
X_BifMonitor	Check this if the BLF feature supports monitor function.	False
Calling Features (VoiceService.1.VoiceProfile.1.Line.n.CallingFeatures.), n = 1 – 6		
CallerIDName	Display name to identify the subscriber. The display name field is usually inserted in a FROM header in outbound SIP requests (such as INVITE) for the purpose of displaying a Caller ID Name on the recipient's device.	None
MaxSessions	The maximum number of simultaneous calls that can be established on this service.	4
CallForwardUnconditional Enable	Enables call forwarding of all calls unconditionally by the phone. If CallForwardUnconditionalNumber is blank, this parameter is treated as if it has been set to False. Note: You can set this parameter from the phone using a Star Code.	False

SPn Services Settings

Parameter	Description	Default Setting
CallForwardUnconditionalNumber	Directory number to forward all incoming calls on this service unconditionally. Maximum Length is 127 characters. Note: You can set this parameter from the phone using a Star Code.	None
CallForwardOnBusyEnable	Enables call forwarding of all incoming calls when the phone is busy. If CallForwardOnBusyNumber is blank, this parameter is treated as if it has been set to False. The phone is considered busy if one of the following conditions holds: <ul style="list-style-type: none"> This service already reaches the limit of simultaneous calls as specified in MaxSessions DND (Do Not Disturb) Service is enabled on this service if the call is routed to the phone where the phone is in a busy state (such as ringing, dialing, playing reorder, or already having 2 calls in progress) Note: You can set this parameter from the phone using a Star Code.	False
CallForwardOnBusyNumber	Directory number to forward all incoming calls on this service when the phone is busy. Maximum Length is 127 characters. Note: You can set this parameter from the phone using a Star Code.	None
CallForwardOnNoAnswerEnable	Enables call forwarding of all incoming calls when the call isn't answered after a period as specified in CallForwardOnNoAnswerRingCount . If CallForwardOnNoAnswerNumber is blank, this parameter is treated as if it has been set to False. Note: You can set this parameter from the phone using a Star Code.	False
CallForwardOnNoAnswerNumber	Directory number to forward all incoming calls when the call isn't answered after a period specified in CallForwardNoAnswerRingCount . Note: You can set this parameter from the phone using a Star Code.	None
CallForwardOnNoAnswerRingCount	Number of rings to be considered by the phone as no answer to an incoming call. Note: 1 ring is approximately 6 seconds.	2
X_BlockedCallers	A comma-separated list of as many as 10 caller numbers to block from calling this service.	None
X_MailboxID	The mailbox ID to subscribe MWI with.	None
X_CheckVoiceMailNumber	The number to call to check voicemail.	None
X_CheckVoiceMailWithCallKey	Enables checking voicemail by pressing and holding down a call key that is bound to this SP service, provided X_CheckVoiceMailNumber is specified.	False

SPn Services Settings

Parameter	Description	Default Setting
MWIEnable	Enables MWI Signal (stutter dial tone) generation. If enabled, any SP voice service enabled on the phone that has MWI Service enabled triggers the generation of stutter dial tone if there are new voicemails for the subscriber on the service provider's voicemail system.	True
X_VMWIEnableMask	Enables Visual Message Waiting Indication for this service on the phone. It is specified as a bit mask, such that bit 0 = Phone 1, bit 1 = phone 2, and so forth.	255
X_MessageCenterEnable	Shows this Message Status under Messages menu of phone UI.	True
X_CallFeaturePreferences Menu	Shows Call Feature Preferences under Preferences menu of phone UI.	False
MessageWaiting	This state indicates if there are any new messages for this subscriber on the service provider's voicemail system.	False
MessageCount	Displays count of new messages, in format new/old (urgent – new/urgent – old).	None
AnonymousCallBlockEnable	Enables blocking of Anonymous Calls on this service. Anonymous calls are rejected with a SIP 486 (Busy) response and Call Forward On Busy service isn't applied. Note: You can set this parameter from the phone using a Star Code.	False
AnonymousCallEnable	Enables masking of Caller-ID information for all outgoing calls. If enabled, the called party sees the call as coming from an anonymous caller. Note: You can set this parameter from the phone using a Star Code.	False
DoNotDisturbEnable	Enables Do Not Disturb Service. If enabled, all incoming calls on this service are treated as if the phone is busy. Note: You can set this parameter from the phone using a Star Code.	False
X_BridgedOutboundCallMaxDuration	Limit on the call duration in seconds for all outbound calls that are bridged from the same or another trunk. A blank or 0 value implies the call duration isn't limited.	None
X_AcceptDialogSubscription	Enables the phone to accept SUBSCRIBE to this trunk's dialog event package.	False

SPn Services Settings

Parameter	Description	Default Setting
X_S RTP	Enables SRTP. Choose one of: <ul style="list-style-type: none"> <code>Disable SRTP</code> = Do not use SRTP for all calls. The call fails if the peer insists on using SRTP only. <code>Use SRTP Only</code> = Require all calls to use SRTP. The call fails if the peer does not support SRTP. <code>Use SRTP When Possible</code> = Use SRTP for a call if the peer supports SRTP. Otherwise, fall back to use regular unencrypted SRTP. 	<code>Disable SRTP</code>
X_ASFeatureEventSubscribe	Enables subscription to the as-feature-event package.	False
X_ShowDiversionHistory	Enables showing the diversion history of a call on the Calls screen.	True
X_RTPRecordURL	URL to upload RTP recording.	None
X_UrlDialingIPAddress	If enabled, dialing or calling an IP address bypasses the outbound proxy.	True
X_UrlDialingDomainName	If enabled, dialing or calling a domain name bypasses the outbound proxy. For example, if you have a conference bridge number SP1(bridge@some-domain.com), you may need to set this option to False if you still need the call to this bridge to go through the outbound proxy.	True
X_IncomingCallMustUseBoundCallKey	Controls whether incoming call must be assigned to a bound call key only or can be assigned to a floating (unbound) call key if bound call keys aren't available. --OR-- If enabled, the phone must assign an incoming to a Line Key with Function = Call Appearance and Service the same as the one the call is from. Otherwise, the phone may also assign the call to a Line Key with Function = Call Appearance and Service = {blank} as a fall-back.	False
X_ConferenceBridge	The number of an external conference bridge to use with calls on this service. Note: If the number also specifies the service to use, such as SP1(bridge@xyz-domain.com), the phone calls the number as it is on the given service. Otherwise, the phone applies its digit map and outbound call route setting to determine which service to use for the call.	<code>cbridge</code>

SPn Services Settings

Parameter	Description	Default Setting
X_CallEndedActionURL	<p>Specifies the ActionURL to invoke when a call ends. The value is a valid HTTP/HTTPS URL that returns an HTTP 2xx response with a valid Obihai Phone XML App in the body.</p> <p>Note: ActionURL automatically includes the called SIP CALL-ID and GMT-timedate as URL-query parameters. You can use the following macros:</p> <ul style="list-style-type: none"> • <code>\$peername</code> = peer caller ID name • <code>\$peernumber</code> = peer caller ID number • <code>\$dur</code> = duration of call • <code>\$refid</code> = the Reference ID, used to control the call in an Phone XML App • <code>\$dir</code> = direction of the call (inbound or outbound) • <code>\$service</code> = the service or line the call is on (e.g., SP1 or SP6) <p>For example: <code>http://abc.com?name=\$peerName&number=\$peernumber&ref=\$refid</code></p>	None
X_CallConnectedActionURL	<p>Specifies the ActionURL to invoke when a call is answered. The value is a valid HTTP/HTTPS URL that returns an HTTP 2xx response with a valid Obihai Phone XML App in the body.</p> <p>Note: ActionURL automatically includes the called SIP CALL-ID and GMT-timedate as URL-query parameters. You can use the following macros:</p> <ul style="list-style-type: none"> • <code>\$peername</code> = peer caller ID name • <code>\$peernumber</code> = peer caller ID number • <code>\$dur</code> = duration of call • <code>\$refid</code> = the Reference ID, used to control the call in an Phone XML App • <code>\$dir</code> = direction of the call (inbound or outbound) • <code>\$service</code> = the service or line the call is on (e.g., SP1 or SP6) <p>For example: <code>http://abc.com?name=\$peerName&number=\$peernumber&ref=\$refid</code></p>	None

SPn Services Settings

Parameter	Description	Default Setting
X_IncomingCallRingingActionURL	<p>Specifies the ActionURL to invoke when an incoming call starts ringing.</p> <p>The value is a valid HTTP/HTTPS URL that returns an HTTP 2xx response with a valid Obihai Phone XML App in the body.</p> <p>Note: ActionURL automatically includes the called SIP CALL-ID and GMT-timedate as URL-query parameters. You can use the following macros:</p> <ul style="list-style-type: none"> • <code>\$peername</code> = peer caller ID name • <code>\$peernumber</code> = peer caller ID number • <code>\$dur</code> = duration of call • <code>\$refid</code> = the Reference ID, used to control the call in an Phone XML App • <code>\$dir</code> = direction of the call (inbound or outbound) • <code>\$service</code> = the service or line the call is on (e.g., SP1 or SP6) <p>For example: <code>http://abc.com?name=\$peerName&number=\$peernumber&ref=\$refid</code></p>	None

Network Provided Services (VoiceService.1.VoiceProfile.1.Line.n.X_NetServices.), n = 1 – 6

ACDAgent	ACD Agent sign or off. Requires BroadSoft/Feature-Sync .	False
ACDAgentUnavailableCodes	ACD Agent Unavailable Codes. Requires BroadSoft/Feature-Sync .	False
AnonymousCall	Allows Anonymous Call service, also known as Block Caller ID. Requires BroadSoft/Feature-Sync or BroadSoft/XSI .	False
AnonymousCallRejection	Allows Anonymous Call Rejection service. Requires BroadSoft/Feature-Sync or BroadSoft/XSI .	False
BroadWorksAnywhere	Allows Broadworks Anywhere service. Requires BroadSoft/XSI .	False
BuddyList	Buddy List Service. Requires BroadSoft or Generic XMPP Service.	False
CallCenter	Call Center. Requires BroadSoft.	False
CallForwardAlways	Call Forward Always. Requires BroadSoft/Feature-Sync or BroadSoft/XSI .	False
CallForwardBusy	Call Forward on Busy. Requires BroadSoft/Feature-Sync or BroadSoft/XSI .	False
CallForwardNoAnswer	Call Forward on No Answer. Requires BroadSoft/Feature-Sync or BroadSoft/XSI .	False
CallLogs	Call Logs Service. Requires BroadSoft.	False

SPn Services Settings

Parameter	Description	Default Setting
CallPark	Park a call (softkey) against current extension with a Feature Access Code.	False
CallParkStatus	Get notified if a call is parked against current extension. Requires BroadSoft.	False
CallPickup	Retrieve a call (softkey) parked against current extension with a Feature Access Code.	False
CallRecording	Call Recording Control. Requires BroadSoft.	False
CallTrace	Customer Originated Call Trace. Requires BroadSoft.	False
CallWaiting	Call Waiting. Requires BroadSoft.	
Directory	Directory Service. Requires BroadSoft or similar directory.	False
DispositionCode	Disposition Code. Requires BroadSoft.	False
DoNotDisturb	Enable subscription of Do Not Disturb (XSI) Event. Requires BroadSoft/Feature-Sync or BroadSoft/XSI .	False
Escalation	Escalation (to a supervisor). Requires BroadSoft.	False
Executive	Executive Call Filtering On/Off. Requires BroadSoft.	False
ExecutiveAssistant	Executive Assistant. Requires BroadSoft.	False
Hoteling	Hoteling or Guest User Sign On or Off. Requires BroadSoft.	False
RemoteOffice	Remote Office. Requires BroadSoft/XSI .	False
SecurityClass	Security Classification. Requires BroadSoft/Feature-Sync .	False
SimultaneousRing	Simultaneous Ring. Requires BroadSoft/XSI .	False
UserProfile	User Profile. Requires BroadSoft/XSI .	False
VisualVoiceMail	Visual Voice Mail. Requires BroadSoft/XSI .	False
ConferenceBridgeControl	View who is on the bridge and apply some control on selected participants. Requires BroadSoft/XSI .	False
Network Directory Setup (VoiceService.1.VoiceProfile.1.Line.n.X_NetServices.Dir.), n = 1 – 6		
URL	URL to download a Group directory. This parameter must be {blank} when using BroadWorks Directories where the URL to download the directory is formed internally based on BroadSoft XSI API.	None
EnableSearch	Enables the Search softkey option on the directory pages.	True
ResultsPerPage	Number of results per page to suggest to the server. Default is 0, which means unspecified. Note that this is just a suggestion to the server, which may return a different number of results per page. The phone always tries to display all the results returned by the server.	0

SPn Services Settings

Parameter	Description	Default Setting
EnableGroup	Enables the Group subdirectory.	True
EnableGroupCommon	Enables the GroupCommon subdirectory.	True
EnableEnterprise	Enables the Enterprise subdirectory.	True
EnableEnterpriseCommon	Enables the EnterpriseCommon subdirectory.	True
EnablePersonal	Enables the Personal subdirectory.	True
<i>Buddy List Setup (VoiceService.1.VoiceProfile.1.Line.n.X_NetServices.BuddyList.), n = 1 – 6</i>		
ServerType	Value can be <code>BroadSoft</code> or <code>Generic</code>	<code>BroadSoft</code>
Priority	Priority of the contact. Must be in the range of –127 to 128, inclusive.	33
VerifyServerCert	Boolean. Turns the verification of server certificate in a SSL connection with the XMPP server on or off.	False
QueryVcard	Boolean. Turns the reading of VCARD on or off for each item in the roster list. Turn on this option if you wish to get a picture of each buddy.	False
MyPresence	The (XMPP) Presence value of this line. Choice of: <ul style="list-style-type: none"> • <code>online</code> • <code>away</code> • <code>xa</code> • <code>dnd</code> • <code>offline</code> 	<code>online</code>
MyStatus	The (XMPP) Status value of this line. Can be any human readable string.	None
<i>Action URL Setup (VoiceService.1.VoiceProfile.1.Line.n.X_NetServices.ActionURL.), n = 1 – 6</i>		
SyncURL	The ActionURL the phone should execute every time when it starts up.	None
OBiStatusNotifyURL	The URL to send <obi-status> XML on start up and for status update. Default is SIP/NOTIFY. To use HTTP/POST, specify <code>http://</code> or <code>https://</code> at the beginning of the URL.	None

SPn Services Settings

Parameter	Description	Default Setting
OBiStatusFormat	Choose the format to report obi-status with. Value can be either <code>obi-status</code> or <code>dialog-info</code> . The former one is an OBiPhone proprietary XML, the latter a <code><dialog-info></code> XML based on RFC4235.	<code>obi-status</code>
OBiStatusFilter	<p>Specifies which status changes to report to the given URL. The value is a string that is a comma-separated list of filter groups. There are three filter groups:</p> <ul style="list-style-type: none"> • Calls: For status related to current calls on the phone, such as Call State, Caller-ID, and so forth. • Services: For status related to a voice service, such as up or down • Features: For status related to the setting of a feature, such as DND on or off <p>Each filter group takes a comma-separated list of arguments to specify which status within each group to report. For the Calls group, the arguments can be one or more of the following: <code>sp1, sp2, sp3, sp4, sp5, sp6, pp</code>.</p> <p>Each argument indicates the service whose call status are to be reported. For the Services group, the arguments can be one or more of the following: <code>sp1, sp2, sp3, sp4, sp5, sp6, pp</code>.</p> <p>Each argument indicates the service whose statuses (except call status) are to be reported. For the Features group, each argument specifies a feature whose setting changes are to be reported. The following is a list of valid feature arguments:</p> <ul style="list-style-type: none"> • <code>aans</code> (Auto Answer Intercom setting) • <code>aud</code> (Audio Path setting) • <code>bac</code> (Block Anonymous Call setting) • <code>bci</code> (Block Caller ID setting) • <code>cwa</code> (Call Waiting setting) • <code>dnd</code> (Do Not Disturb setting) • <code>dnr</code> (Do Not Ring setting) • <code>miss</code> (Number of new missed calls) • <code>mwi</code> (New Messages Waiting status) • <code>ncalls</code> (Number of calls) • <code>pg1</code> (Page Group 1 setting) • <code>pg2</code> (Page Group 2 setting) • <code>ring</code> (Ring status) <p>Note: If OBiStatusFormat = <code>dialog-info</code>, only the Calls filter group is used. The other filter groups are ignored.</p> <p>For example: <code>Calls (sp1) , Services (sp1, sp2) , Features (dnd)</code></p>	<code>Calls (sp1, sp2, sp3, sp4, sp5, sp6)</code>

OBiTALK Service Web Page

The following configuration parameters are available on this page.

OBiTALK Service Settings

Parameter	Description	Default Setting
OBiTALK Service Settings (VoiceService.1.X_P2P.1.)		
Enable	Enables the OBiTALK Service (the built-in free voice service that comes with every OBi Edition phone).	True
DisplayLabel	A label that represents this service on the phone screen. Special characters must include a closing character. For example, an open curly bracket must also include a closing curly bracket.	None
DisplayNumber	A number that represents this service on the phone screen. Special characters must include a closing character. For example, an open curly bracket must also include a closing curly bracket.	None
LocalPort	The UDP or TCP port used by the phone to send and listen for OBiTALK messages.	10000
TryMultiplePorts	Enables the phone to try a few random UDP ports until it can successfully join the OBiTALK network.	True
Transport	The transport protocol to use for OBiTALK traffic. Choice of: <ul style="list-style-type: none"> • UDP • UDP/TCP • UDP/TLS • TLS 	UDP/TCP
ServerAddress	OBiTalk Server IP Address Should be empty in normal operation.	None
LastRegistrarAddress	IP address and port number of the last registrar address used.	None
TLSserverPort	TLS server port.	443
DisplayName	Display name to identify the subscriber, for the purpose of displaying a Caller ID Name on the recipient's phone.	None
DigitMap	Digit map to restrict numbers that can be dialed or called with this service. See the Call Routing and Digit Map Configuration sections for a description of the syntaxes to specify a digit map.	(<ob>xxxxxxxxx obxxxxxxxxx)

OBiTALK Service Settings

Parameter	Description	Default Setting
InboundCallRoute	Routing rule for directing incoming calls on this service. The default rule is to send all incoming calls to the phone. See the Call Routing and Digit Map Configuration sections for a description of the syntaxes to specify this parameter.	ph
RingProfile	Select a Ring Profile to ring the phone with when an incoming call is routed to the phone. Choices are A or B.	A
CodecProfile	Select a Codec Profile to be used for all calls on this service. Choices are A or B.	A
DefaultRing	Default ring pattern number to ring the phone for incoming calls on this trunk that are routed to the phone according to the InboundCallRoute of this service. The ring pattern is taken from the selected Ring Profile. Valid choices are 1–10.	2
RingtoneFile	Default Ringtone for OBiTALK Service.	None
CallOnHoldRing	Pattern to ring phone when holding a call on this trunk that has been connected to the phone. Typically this is a very short distinctive ring pattern that serves as a reminder to you that a call is being on hold. The ring pattern is taken from the selected Ring Profile. Choice of: <ul style="list-style-type: none"> • NO Ring • 1-10. 	8
RepeatDialRing	The ring pattern number to use to ring the phone when a repeat dial operation on this trunk is successful as the called party is either ringing or answered.	4
DTMFMethod	Method to pass DTMF digits to peer phone. Choice of: <ul style="list-style-type: none"> • Inband: DTMF tone are sent as inband audio signal • RFC2833 : DTMF tone events are relayed per RFC2833 • SIPInfo: DTMF tones are relayed with SIP INFO request • Auto: Method to use based on call setup negotiation (either Inband or RFC2833 can be negotiated) 	Auto
UseFixedDurationRFC2833DTMF	When relaying DTMF digit events on this trunk using RFC2833, the RFC2833 RTP packets normally keep streaming for as long as the digit is pressed. With this option set to True, the phone sends only one RTP digit event packet with a fixed duration of 150 ms regardless how long the digit has been pressed.	False
SymmetricMedia	If incoming payload type changes unannounced, after 10 packets with the new payload type, decoding switches to the new format. If symmetric media is enabled, outgoing packets are also in the new format.	True

Calling Features (VoiceService.1.X_P2P.1.CallingFeatures.)

OBiTALK Service Settings

Parameter	Description	Default Setting
CallForwardUnconditionalEnable	Enables call forwarding of all calls unconditionally by the phone. If CallForwardUnconditionalNumber is blank, this parameter is treated as if it has been set to False. Note: You can set this parameter from the phone using a Star Code.	False
CallForwardUnconditionalNumber	Directory number to forward all incoming calls on this service unconditionally. Maximum length is 127 characters. Note: You can set this parameter from the phone using a Star Code.	None
CallForwardOnBusyEnable	Enables call forwarding of all incoming calls when the phone is busy. If CallForwardOnBusyNumber is blank, this parameter is treated as if it has been set to False. The phone is considered busy if one of the following conditions holds: <ul style="list-style-type: none"> This service already reaches the limit of simultaneous calls as specified in MaxSessions. DND (Do Not Disturb) Service is enabled on this service If the call is routed to the phone where the phone is in a busy state (such as ringing, dialing, playing reorder, or already having 2 calls in progress) Note: You can set this parameter from the phone using a Star Code.	False
CallForwardOnBusyNumber	Directory number to forward all incoming calls on this service when the phone is busy. Maximum Length is 127 characters. Note: You can set this parameter from the phone using a Star Code.	None
CallForwardOnNoAnswerEnable	Enable call forwarding of all incoming calls when the call isn't answered after a period as specified in CallForwardOnNoAnswerRingCount . If CallForwardOnNoAnswerNumber is blank, this parameter is treated as if it has been set to False. Note: You can set this parameter from the phone using a Star Code.	False
CallForwardOnNoAnswerNumber	Directory number to forward all incoming calls when the call isn't answered after a period specified in CallForwardOnNoAnswerRingCount Note: You can set this parameter from the phone using a Star Code.	None
CallForwardOnNoAnswerRingCount	Number of rings to be considered by the phone as no answer to an incoming call. Note: 1 ring is approximately 6 seconds.	2
BlockedCallers	A comma-separated list of as many as 10 caller numbers to block from calling this service.	None

OBI-TALK Service Settings

Parameter	Description	Default Setting
MaxSessions	The maximum number of simultaneous calls that can be established on this service.	2
AnonymousCallBlockEnable	Enables blocking Anonymous Calls on this service. Anonymous calls are rejected with a SIP 486 (Busy) response and Call Forward On Busy service isn't applied. Note: You can set this parameter from the phone using a Star Code.	False
AnonymousCallEnable	Enables Anonymous Call.	False
DoNotDisturbEnable	Enables Do Not Disturb.	False
CallStatusAggregator	An OBi number that serves as call status aggregator for OBi phones.	None
RTPRecordURL	URL to upload RTP recording.	None
<i>Inbound Direct Dialing Authentication (VoiceService.1.X_P2P.1.VoiceGateway.)</i>		
AuthMethod	<p>The OBI-TALK protocol allows incoming calls to indicate a target number that is different from this phone's OBi number. In this case, the phone attempts to establish and bridge the call to the target number according to the rules configured in the trunk's InboundCallRoute parameter. Hence, the phone acts as a gateway and the method is referred to direct dialing or 1-stage dialing (versus 2-stage dialing via the Auto Attendant). Since the caller isn't able to enter a PIN in such cases, an automated method based on signaling protocol must be used to authenticate the caller if authentication is required. The phone offers the following choices for this purpose:</p> <ul style="list-style-type: none"> • None = Disable authentication • HTTP Digest = Use HTTP Digest with user ID and password pairs. Note that at least one of AuthPasswordx (x=1,2,3,4) must be specified, otherwise authentication is disabled. 	HTTP Digest
AuthUserID1	One of 4 user IDs for authenticating direct dialing callers.	None
AuthPassword1	One of 4 passwords for authenticating direct dialing callers.	None
AuthUserID2	One of 4 user IDs for authenticating direct dialing callers.	None
AuthPassword2	One of 4 passwords for authenticating direct dialing callers.	None
AuthUserID3	One of 4 user IDs for authenticating direct dialing callers.	None
AuthPassword3	One of 4 passwords for authenticating direct dialing callers.	None
AuthUserID4	One of 4 user IDs for authenticating direct dialing callers.	None
AuthPassword4	One of 4 passwords for authenticating direct dialing callers.	None

OBiTALK Service Settings

Parameter	Description	Default Setting
<i>Jitter Buffer (VoiceService.1.X_P2P.1.JIB.)</i>		
Adaptive	Enables jitter buffer adaptation.	True
MaximumSize	Maximum jitter buffer size in ms.	250
SetPoint	Initial play out delay in ms.	60
Target	Target play out delay in ms.	20
AdaptationSlope	Maximum adaptation slope in samples per 10 ms.	16

Auto Attendant Web Page

The following configuration parameters are available on this web page.

Auto Attendant Settings

Parameter	Description	Default Setting
<i>Auto Attendant (VoiceService.1.X_AA.1.)</i>		
Enable	Enables AA. When enabled, the AA answers an incoming call that has been routed to it after the period specified in AnswerDelay . If disabled, the AA won't attempt to answer any incoming call.	True
DigitMap	Once the AA answers an incoming call, it presents the caller with an option to make a further call using one of the available voice services on the device. This Digit map serves to restrict the numbers that can be dialed or called via this AA option. See the Call Routing and Digit Map Configuration sections for a description of the syntaxes to specify a digit map.	<code>([1-9]x?(Mpli) [1-9] [1-9] [0-9] <00:\$1> [0-8] **1 (Msp1) **2 (Msp2) **3 (Msp3) **4 (Msp4) **9 (Mpp) (Mpli))</code>
OutboundCallRoute	After the caller dials a number that is acceptable by the AA (according to its DigitMap) to make a further call, the phone uses this outbound call routing rule to determine which service to make this call with. See the Call Routing and Digit Map Configuration sections for a description of the syntaxes to specify this parameter.	<code>{ ([1-9]x?(Mpli)) :pp}, { 0:ph}, { (<**1:> (Msp1)) :sp1}, { (<**2:> (Msp2)) :sp2}, { (<**3:> (Msp3)) :sp3}, { (<**4:> (Msp4)) :sp4}, { (<**9:> (Mpp)) :pp}, { (Mpli) :pli}</code>

Auto Attendant Settings

Parameter	Description	Default Setting
PrimaryLine	<p>The primary line is the service that does not require any access code prefix (such as **1 or **9) when dialing. It is the default service to be used for making the call when no explicit access code prefix is entered. This parameter indicates to the phone which voice service is considered as the primary line when dialing out via the Auto Attendant. Choice of:</p> <ul style="list-style-type: none"> • SP1 Service (code = sp1) • SP2 Service (code = sp2) • SP3 Service (code = sp3) • SP4 Service (code = sp4) • SP5 Service (code = sp5) • SP6 Service (code = sp6) • OBiTALK Service (code = pp1) • Trunk Group 1 (code = tg1) • Trunk Group 2 (code = tg2) <p>The phone processes the parameter by substituting occurrences of pli and (Mpli) in DigitMap and OutboundCallRoute with the corresponding code and (Mcode).</p>	SP1 Service
AnswerDelay	Time in milliseconds that the AA waits before answering an incoming call that has been routed to it.	4000
CallbackAnswerDelay	Delay in ms the AA answers after ring when the AA should call the caller back if the caller hangs up before it answers.	10000
NumberOnNoInput	If the caller does not enter any option from the top level menu after the menu has been announced 3 times, the AA directs the caller to the number specified in this parameter. If this number isn't specified, the AA terminates the current call.	0 Note: According to the default DigitMap and OutboundCallRoute parameters, calling 0 means ringing the phone.
UsePIN	Enables using a PIN to authenticate callers when they select the option to make a further call. If PIN1 , PIN2 , PIN3 , and PIN4 are all empty, the phone treats it as if UsePIN is set to No. Otherwise, the caller must enter one of the non-empty PINs to proceed.	False
PIN1	PIN code to make a call (must be all digits). Maximum length = 15 digits.	None
PIN2	PIN code to make a call (must be all digits). Maximum length = 15 digits.	None

Auto Attendant Settings

Parameter	Description	Default Setting
PIN3	PIN code to make a call (must be all digits). Maximum length = 15 digits.	None
PIN4	PIN code to make a call (must be all digits). Maximum length = 15 digits.	None

Gateways and Trunk Groups Web Page

The following configuration parameters are available on this page.

Gateways and Trunk Groups Settings

Parameter	Description	Default Setting
Voice Gateway <i>n</i> (VoiceService.1.X_VoiceGateway.n), n = 1 – 8		
Enable	Enables this voice gateway.	True
Name	An arbitrary user-friendly name to identify this gateway (optional).	None
AccessNumber	The gateway's OBiTALK number, including trunk information, such as: PP (ob200112334) or PP (ob300331456). If the value is blank, the phone treats this voice gateway as disabled. This can also be set to a SIP URL, such as: SP1 (sip.mycompany.com:5060), or SP2 (192.168.15.113).	None
DigitMap	DigitMap for this voice gateway. It can be referenced as (Mvgn).	(xx.)
AuthUserID	A user ID to authenticate with the voice gateway.	None
AuthPassword	A password to authenticate with the voice gateway.	None
Trunk Group <i>n</i> (VoiceService.1.X_TrunkGroup.n), n = 1 – 4		
Enable	Enables this trunk group.	True
Name	An arbitrary user-friendly name to identify this trunk group (optional).	None

Gateways and Trunk Groups Settings

Parameter	Description	Default Setting
TrunkList	A comma-separated list of names of trunks to include in this trunk group.	For TG1 and TG2, the default is: sp1, sp2, sp3, sp4 For other TGs, the default is (blank).
DigitMap	Digit map associated with this trunk group. It can be referenced as (Msp1) for TG1 and TG2.	For TG1 and TG2, the default is: (1xxxxxxxxxxx <1>[2-9]xxxxxxxxxx 011xx. xx.) For other TGs, the default is (xx.).

IP Phone Settings

The IP Phone Settings parameters web pages show parameters for customizing your phone. They include these pages:

- [Phone Settings Web Page](#)
- [Left Line Keys Web Page](#)
- [Right Line Keys Web Page \(VVX 450 only\)](#)
- [Programmable Keys Web Page](#)
- [Soft Keys Web Page](#)
- [LDAP Search Web Page](#)
- [Line Key Customization Web Page](#)
- [Screen Item Customization Web Page](#)

Phone Settings Web Page

The following configuration parameters are available on this page.

Phone Settings

Parameter	Description	Default Setting
Phone Settings (VoiceService.1.Phone.)		
DigitMap	This digit map restricts the numbers that can be dialed or called from the phone. See the Call Routing and Digit Map Configuration sections for a description of the syntaxes to specify a digit map.	([1-9]x?* (Mpli) [1-9]S9 [1-9] [0-9]S9 *** **0 **1 (Msp1) **2 (Msp2) **3 (Msp3) **4 (Msp4) **9 (Mpp) (Mpli))

Phone Settings

Parameter	Description	Default Setting
OutboundCallRoute	<p>After the caller dials a number that is acceptable according to the DigitMap, the phone uses this outbound call routing rule to determine which service to make this call with.</p> <p>See the Call Routing and Digit Map Configuration sections for a description of the syntaxes to specify this parameter.</p>	<pre>{ ([1-9]x?* (Mpli)):pp}, { **0:aa}, { **:aa2}, { (<**1:> (Msp1)):sp1}, { (<**2:> (Msp2)):sp2}, { (<**3:> (Msp3)):sp3}, { (<**4:> (Msp4)):sp4}, { (<**9:> (Mpp)):pp}, { (Mpli) :pli}</pre>
CallReturnDigitMaps	<p>Call Return enables you to call the last caller by dialing a star code (*69 by default). The phone implements this service by remembering the number of the last caller in memory. However, the stored information does not include any dialing prefix to tell the phone which voice service to use to call back the last caller. This list of digit maps serves the purpose of mapping a caller's number to one that includes the desired dialing prefix used exclusively for call return service.</p>	<pre>{pli: (xx.)}, {sp1: (<**1>xx.)}, {sp2: (<**2>xx.)}, {pp: (<**9>xx.)}</pre>
PrimaryLine	<p>The primary line is the service that does not require any access code prefix (such as **1 or **9) when dialing. It is the default service to be used for making the call when no explicit access code prefix is entered. This parameter indicates to the phone which voice service is considered as the primary line when dialing out from the phone. Choice of:</p> <ul style="list-style-type: none"> • SP1 Service (code = sp1) • SP2 Service (code = sp2) • SP3 Service (code = sp3) • SP4 Service (code = sp4) • SP5 Service (code = sp5) • SP6 Service (code = sp6) • OBiTALK Service (code = pp1) • Trunk Group 1 (code = tg1) • Trunk Group 2 (code = tg2) <p>The phone processes the parameter by substituting of the occurrences of pli and (Mpli) in DigitMap, OutboundCallRoute, and CallReturnDigitMaps with the corresponding code and (Mcode).</p>	SP1 Service

Phone XML App (VoiceService.1.Phone.Control.)

Phone Settings

Parameter	Description	Default Setting
AcceptPhoneXmlAppViaHTTP	Controls whether to accept XML App pushed to the phone via HTTP/Post. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
AcceptPhoneXmlAppViaOBiTALK	Controls whether to accept XML App pushed to the phone from OBiTALK or PDMS-SP using a SIP/NOTIFY. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
AcceptPhoneXmlAppViaSP1	Controls whether to accept XML App pushed to the phone via SP1 using a SIP/NOTIFY. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
AcceptPhoneXmlAppViaSP2	Controls whether to accept XML App pushed to the phone via SP2 using a SIP/NOTIFY. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
AcceptPhoneXmlAppViaSP3	Controls whether to accept XML App pushed to the phone via SP3 using a SIP/NOTIFY. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
AcceptPhoneXmlAppViaSP4	Controls whether to accept XML App pushed to the phone via SP4 using a SIP/NOTIFY. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
AcceptPhoneXmlAppViaSP5	Controls whether to accept XML App pushed to the phone via SP5 using a SIP/NOTIFY. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
AcceptPhoneXmlAppViaSP6	Controls whether to accept XML App pushed to the phone via SP6 using a SIP/NOTIFY. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no

Phone Settings

Parameter	Description	Default Setting
PhoneXmlAppAuthPassword	If authentication is enabled for the corresponding channel, this parameter specifies the authentication password for the pushed Phone XML App requests (HTTP/Post or SIP/NOTIFY) using MD5 digest.	None
AcceptResyncViaOBiTALK	Controls whether to accept NOTIFY/Resync or NOTIFY/Reboot from OBiTALK or PDMS-SP. Choice of: <ul style="list-style-type: none"> no yes with authentication yes without authentication 	no
ResyncViaOBiTALKAuthenticationPassword	If authentication is enabled, this parameter specifies the authentication password for the resync/reboot from OBiTALK or PDMS-SP using MD5 digest.	None

GUI Menus (VoiceService.1.Phone.GUI.Menus.)

MainMenu1	A comma-separated list of items to display on the phone's Main menu.	directories,calls,call-histories,all-preferences,settings,prod-info,messages,buddy
PreferencesMenu1	A comma-separated list of items to display on the phone's Preferences Menu. Note: The values of PreferencesMenu1 through PreferencesMenu6 are concatenated together to form a complete list of items to be displayed.	language;Language,timeFormat;TimeFormat,dateFormat;DateFormat,edialer;EnhancedDialer,skin;Skin,bgpic;BackgroundPicture,dring;DefaultRingtone,hpTime;AutoHome Page
PreferencesMenu2	A comma-separated list of items to display on the phone's Preferences Menu.	sdim,sdimDelay,ssvr;ScreenSaver,ssvrDelay;ScreenSaverDelay,ssvrType;ScreenSaverType,ssvrType2,ssvrLock;Require Passcode On Wake Up,ssvrPass;Wake Up Passcode,ssvrCustom;Screen Saver Show Custom Contents

Phone Settings

Parameter	Description	Default Setting
PreferencesMenu3	A comma-separated list of items to display on the phone's Preferences Menu.	brightness;Screen Brightness,dfont;Font,a ppCols;Home App Columns,packcalls;Pack Calls On Display,audioDevice;Pre ferred Audio Device,headsetDevice;Pr eferred Headset Device,ehs;Electronic Hook Switch,dnd;Do Not Disturb,dnr;Do Not Ring
PreferencesMenu4	A comma-separated list of items to display on the phone's Preferences Menu.	cfa,cfb,cfna,cwa;Call Waiting,bac;Block Anonymous Call,bci;Anonymous Call,aans;Auto Answer Page,pg1;Join Page Group 1,pg2;Join Page Group 2,ringerVol;Ringer Volume
PreferencesMenu5	A comma-separated list of items to display on the phone's Preferences Menu.	speakerVol;Speakerphone Volume,micGain;Speakerp hone Mic Gain,handsetVol;Handset Volume,handsetGain;Hand set Mic Gain,headsetRJ9Vol,RJ9 Headset Volume,headsetRJ9Gain;R J9 Headset Mic Gain,headsetRJ9Side;RJ9 Headset Sidetone
PreferencesMenu6	A comma-separated list of items to display on the phone's Preferences Menu.	headsetUSBVol;USB Headset Volume,headsetUSBGain;U SB Headset Mic Gain,headsetAEC;AEC for RJ9 Headset,headsetENS;Acou stic Fence for RJ9 Headset,handsetENS;Acou stic Fence for Handset,swapspkhs,swaps pkhsled

Phone Settings

Parameter	Description	Default Setting
SettingsMenu1	A comma-separated list of items to display on the phone's Settings Menu. Note: The values of SettingsMenu1 and SettingsMenu2 are concatenated together to form a complete list of items to be displayed.	net;Network,wifi;WiFi,obiline;OBiLine,flash;Storage,ringfile;Ringtones,pkeys;Programmable Keys
SettingsMenu2	A comma-separated list of items to display on the phone's Settings Menu.	lkeys;Line Keys,voice;Voice Services,sd99,clrcache,admin;Device Admin,login
ProductInfoMenu1	A comma-separated list of items to display on the phone's Product Info Menu. Note: The values of ProductInfoMenu1 and ProductInfoMenu2 are concatenated together to form a complete list of items to be displayed.	ipaddr,model;Model,obinumber;OBiNumber,mac;MAC Address,wfmac;WiFi MAC Address,serial;Serial Number
ProductInfoMenu2	A comma-separated list of items to display on the phone's Product Info Menu.	swver;Software Version,hwver;Hardware Version,ztinfo;Customization Status,uptime;Up Time
NetServicesMenu1	A comma-separated list of items to display on the phone's Net Services Menu, also known as Per Line Preferences Menu. Net Services is short for Network-Provided Services.	regst,ringtone,acd,bac,bci,bwanw,buddy,ccs,cfa,cfb,cfna,rec,dnd,dispcode,exec,xass,hotel,clog,dir,rmoft,secClass,simring
NetDirectoryMenu1	A comma-separated list of items to display on the phone's Network Directory (Net Dir for short) Menu.	name,firstName,lastName,ext,number,email,groupId,department,mobile,userId,impId,presence
DeviceAdministration Menu1	A comma-separated list of items to display on the phone's Device Administration Menu.	admin.port,admin.adminPassword,admin.userPassword,admin.syslogServer,prov.itsp.Method,prov.itsp.Interval,prov.itsp.ConfigURL,fwup.Method,fwup.Interval,fwup.FirmwareURL
FactoryResetRequire AdminPassword	Requires you to enter the Web Admin Password to factory reset the phone from the phone UI.	True

Calling Features (VoiceService.1.Phone.CallingFeatures.)

Phone Settings

Parameter	Description	Default Setting
MWIEnable	Enables MWI Signal (stutter dial tone) generation. If enabled, any SP voice service enabled on the phone that has MWI Service enabled triggers the generation of stutter dial tone if there are new voicemails for the subscriber on the service provider's voicemail system.	True
VMWIEnable	Enables VMWI Signal generation. If enabled, any SP voice service enabled on the phone that has VMWI Service enabled triggers the generation of VMWI signal if there are new voicemails for the subscriber on the service provider's voicemail system.	True
CallTransferEnable	<p>Enables Call Transfer. You can start Call Transfer by hanging up the phone in one of the following scenarios:</p> <ul style="list-style-type: none"> • One call on hold while a second outgoing call ringing • One call on hold while a second outgoing call connected • One call connected while a second outgoing call ringing • 3-way conference with both calls connected <p>Note: The OPUS codec doesn't support 3-way calling with both legs using OPUS.</p> <p>If Call Transfer is disabled, hanging up the phone in the above scenarios ends all the calls except for the one that is holding, which remains on hold (the first and second cases above).</p>	True
ConferenceCallEnable	<p>Enables 3-way Conference Call with local audio mixing. You can start a Conference Call by hook flashing the phone in one of the following scenarios:</p> <ul style="list-style-type: none"> • One call on hold while a second outgoing call ringing • One call on hold while a second outgoing call connected <p>The first case is an early conference. The second conferee is still ringing. The other two parties can converse while hearing ringback tone in the background until the third party answers. In either case, you can end the call with the second conferee by hook flashing another time and the call reverts to a 2-way call.</p> <p>If Conference Call service is disabled, then hook flashing the phone resumes the holding call but ends the second outgoing call in the first case, and swaps between the two calls in the second case, as in a call waiting situation.</p>	True

Phone Settings

Parameter	Description	Default Setting
UseExternalConferenceBridge	Uses the external conference bridge when starting a conference call.	False
CallWaitingEnable	Enables call waiting service. Call Waiting is the situation where a new incoming call is routed to the phone when there is already another call connected. If this service is enabled, the phone plays the call-waiting tone to alert you and generates the CWCID signal if CWCID is enabled. Then you can then swap between the two calls by hook flashing. If the service is disabled, the phone rejects the incoming call as busy. Note: You can set this parameter from the phone using a Star Code.	True
DoNotRingEnable	Enables the Do Not Ring phone feature.	False
CallParkRingEnable	Enables playing of a short ring burst at configurable regular intervals for a configurable total duration wherever there is parked call in any of the call park orbit being monitored with a feature key (using the Call Park Monitor function).	False
AnonymousCallBlockEnable	Enables the Anonymous Call Block feature.	False
AnonymousCallEnable	Enables the Anonymous Call feature.	False
CallForwardUnconditionalEnable	Enables the Call Forward All Calls feature.	False
CallForwardUnconditionalNumber	Target number to forward all callers.	None
CallForwardOnBusyEnable	Enables the Call Forward on Busy feature.	False
CallForwardOnBusyNumber	Target number to forward all callers on busy.	None
CallForwardOnNoAnswerEnable	Enables the Call Forward on No Answer feature.	False
CallForwardOnNoAnswerNumber	Target number to forward all callers when not answered after a few rings.	None
CallForwardOnNoAnswerRingCount	Number of rings to wait before activating call forward on no answer.	2
JoinPageGroup1	Join (if enabled) or Leave (if disabled) Multicast Page Group 1.	False

Phone Settings

Parameter	Description	Default Setting
JoinPageGroup2	Join (if enabled) or Leave (if disabled) Multicast Page Group 2.	False
AutoAnswerEnable	Enables auto-answering an incoming call for paging application.	True
ToneProfile	Selects a Tone Profile for call progress tone generation. Choices are A or B.	A
StarCodeProfile	Selects a Star Code Profile for interpreting the Star Codes you enter. Choice of: <ul style="list-style-type: none"> • None • A • B If the value is set to <code>None</code> , no star code is recognized by the phone.	A
LastDialedNumber	This read-only status parameter shows the last number dialed out on the phone.	
LastCallerNumber	This read-only status parameter shows the last number to ring the phone.	
AcceptMediaLoopback	Enables the phone to accept incoming media loopback calls.	True
MediaLoopbackAnswerDelay	Delay in milliseconds before the phone answers an incoming media loopback call.	0
MediaLoopbackMaxDuration	Maximum duration in seconds to allow for an inbound media loopback call. Set the value to blank or 0 to make it unlimited.	0
RepeatDialInterval	Interval in seconds between redial in a repeat dial operation.	30
RepeatDialExpires	Duration of time in seconds when a repeat dial operation remains active.	1800
MOHServiceNumber	A number to call and bridge to get some audio played when the phone user holds the call	None
PlaySITOnCallFailureCodes	Specify which INVITE failure response codes trigger the phone to play SIT tone to alert the phone user. It must be specified with a valid digit map format. If nothing valid is specified, the phone plays the normal reorder tone when the outbound call fails.	(404 9xx)

Phone Settings

Parameter	Description	Default Setting
InvalidNumberAnnouncementOption	Option to play an announcement of the error when user dials an invalid number. Choice of: <ul style="list-style-type: none"> SIT_ONLY DEFAULT_ONLY SIT_AND_DEFAULT CUSTOM_ONLY SIT_AND_CUSTOM 	SIT_ONLY
CustomInvalidNumberAnnouncement	The custom announcement to play when user dials an invalid number.	None
CallRemovalDelayOnPeerEndCall	Number of seconds to delay removal of the call from the screen when the far end ends the call. During that delay, reorder tone is played if the call was connected (active) when it was ended	0
CallHistoryUploadURL	URL to upload call history.	None
CallHistoryUploadFilter	A comma-separated list of services to include in the call history upload.	sp1, sp2, sp3, sp4, sp5, sp6, pp
MergeConfereeCalls	Enables grouping all the active conferee calls into a single call item and not showing each individual conferee call items on screen. The call state of this virtual call item is always Conferencing . Pressing the End soft key on this special call item ends all the conferee calls. Pressing the Split or NewCall soft key splits it up into individual conferee call items on screen while placing each conferee call on hold. Pressing the Join soft key lets you drop out of the conference while transferring one conferee to the other so they can continue with the conversation.	False
TransferOnHangUpConference	Enables transferring one conferee call to the other (in other words, Join the two conferees) when hanging up on a merged conference call. Otherwise, all the conferees are ended just like pressing the End soft key. Hanging up means either on-hook the handset while using the handset, turn off the speakerphone while using the speakerphone, or turn off the headset while using a headset.	False
Network Services (VoiceService.1.Phone.NetServices.)		
Enable	Makes this option available on the phone Main menu.	True
Name	A user-friendly name to identify this directory on the Phone screen.	None

Network Directory (VoiceService.1.Phone.NetDir.)

Phone Settings

Parameter	Description	Default Setting
Enable	Makes this option available on the phone Main menu.	True
Name	A user-friendly name to identify this directory on the Phone screen.	None
VoiceService	The Service that provides the directory. Choice of: <ul style="list-style-type: none"> • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • LDAP 	SP1

Buddy List (VoiceService.1.Phone.BuddyList.)

Enable	Makes this option available on the phone Main menu.	True
Name	A user-friendly name to identify this buddy list on the Phone screen.	None
VoiceService	The Service that provides the directory. Choice of: <ul style="list-style-type: none"> • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • LDAP 	SP1
MyStatusHistory	A comma-separated list of (XMPP) Status that you can select from the phone UI to be used as the current (XMPP) Status.	None

User Preferences Settings (VoiceService.1.Phone.SoftKeys.)

CallForwardUnconditionalFeatureProvider	Selects the provider of the Call Forward Unconditional feature under User Preferences. Choice of: <ul style="list-style-type: none"> • Phone • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	Phone
------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------

Phone Settings

Parameter	Description	Default Setting
CallWaitingFeatureProvider	<p>Selects the provider of the Call Waiting feature under User Preferences. Choice of:</p> <ul style="list-style-type: none"> • Phone • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	Phone
DoNotDisturbFeatureProvider	<p>Selects the provider of the Do Not Disturb feature under User Preferences. Choice of:</p> <ul style="list-style-type: none"> • Phone • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	Phone
AnonymousCallBlockFeatureProvider	<p>Selects the provider of the Block Anonymous Callers feature under User Preferences. Choice of:</p> <ul style="list-style-type: none"> • Phone • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	Phone
AnonymousCallFeatureProvider	<p>Selects the provider of the Anonymous Call feature under User Preferences. Choice of:</p> <ul style="list-style-type: none"> • Phone • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	Phone

Phone Settings

Parameter	Description	Default Setting
MessageStatusFeatureProvider	Selects the provider of the primary Message Status displayed on the phone. Choice of: <ul style="list-style-type: none"> • Phone • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 	Phone
ACDFeatureProvider	Selects the provider of the ACD Agent State feature to be shown on the phone's status bar area. Choice of: <ul style="list-style-type: none"> • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 	SP1

Timers (VoiceService.1.Phone.Timer.)

ReorderDelayTime	Delay in ms to start reorder tone after peer hangs up.	5500
DigitMapLongTimer	Default number of seconds when the digit map processor should timeout waiting for more digits for matching patterns with unspecified length.	10
DigitMapShortTimer	Default number of seconds when the digit map processor should timeout waiting for more digits when at least one pattern with specific length has matched.	2

Page Group n (VoiceService.1.Phone.PageGroup.n.), n = 1, 2

GroupName	A user-friendly name to identify the multicast page group.	None
MulticastAddress	The multicast address to use for the page group.	224.1.1.100
MulticastPort	The multicast port to use for the page group.	PG1: 65322 PG2: 65324
TTL	TTL of multicast packet (1-255).	2
ParticipantName	A human-readable name to identify this participant to the group.	None

Phone Settings

Parameter	Description	Default Setting
AudioCodec	The audio codec to use for the page group. Choice of: <ul style="list-style-type: none"> • G711U • G711A • G729 • iLBC • G722 	G711U
TxPacketSize	RTP transmission packet size in milliseconds.	20
RTCPTxInterval	RTCP transmission interval when talking in milliseconds. PG1 only.	0
SilenceSuppression	Enables silence suppression when talking. PG1 only.	False
PushToTalk	Enables push to talk mode. PG1 only.	False
PlayToneOnIncomingPage	Play a short Paging Tone on receiving a new incoming page. PG1 only.	True
PreferredAudioDevice	PG1 only. Choice of: <ul style="list-style-type: none"> • Speaker • Headset • System 	System
PageTimeout	Timeout in seconds to automatically end an outgoing page. Set to 0 to disable this automatic timeout mechanism. PG1 only.	0
EnableJoinLeaveGroupUserControl	Enables you option to join or leave the group by pressing the key quickly. PG1 only.	True

Left Line Keys Web Page

The following configuration parameters are available on this page.

Left Line Keys Settings

Parameter	Description	Default Setting
<i>Line Key n (VoiceService.1.Phone.LineKey.n), n = 1 – 2 for VVX 150, 1 – 16 for VVX 250, 1 – 24 for VVX 350, 1 – 24 for VVX 450</i>		
InvalidKeyHandling	Selects the method to handle an invalid key that has its Service field specified with a service that has not been configured. Applies unconditionally to all Line Keys, Left and Right. Choice of: <ul style="list-style-type: none"> • Unmodify • Pack • Disable 	Unmodify

Left Line Keys Settings

Parameter	Description	Default Setting
HiddenTabActivityIndicator	Enables indication of Busy Lamp Field or Call Park activities under a hidden tab by blinking the Tab Number. Applies unconditionally to all Line Keys, Left and Right.	False
Function	<p>Function assigned to this key. Can be set for each Left Line Key.</p> <p>Choice of:</p> <ul style="list-style-type: none"> • Function • Auto Answer • Intercom • ACD Sign On/Off • Blind Transfer • Block Anonymous Call • Block Caller ID • Busy Lamp Field • Call Appearance • Call Forward • Call Park Monitor • Call Waiting • Conference • Disposition Code • Do Not Disturb • Do Not Ring • Exec Filter On/Off • Exec Assistant • Hold • Hoteling • Line Monitor • Next Tab • Message Status • Page Group 1 • Page Group 2 • Presence Monitor • Security Class • Speed Dial • Transfer • Action URL • Add To Conference • Blind Transfer 2 • Left Next Tab 	Call Appearance

Left Line Keys Settings

Parameter	Description	Default Setting
Service	The Voice Service on which to apply the selected function. Choice of: <ul style="list-style-type: none"> • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	None
Number	The number, extension, or user ID of the entity associated with the assigned function.	None
Name	A display name for the entity associated with the assigned function.	None
Group	ID of the Custom Line Key Group to use to form the contents of the line key window.	None
Style	ID of the Custom Line Key Style to use to format the contents of the line key window.	None
MaxCalls	Maximum number of calls this key should handle if the function is Call Appearance.	1
Status	This read-only status parameter shows the last number to ring the phone.	

Right Line Keys Web Page (VVX 450 only)

The following configuration parameters are available on this page.

Right Line Keys Settings

Parameter	Description	Default Setting
Right Line Key <i>n</i> (VoiceService.1.Phone.LineKey.n.), <i>n</i> = 1 – 24		
Function	Function assigned to this key. Can be set for each Line Key. Choice of: <ul style="list-style-type: none"> • Auto Answer • Intercom • ACD Sign On/Off • Blind Transfer • Block Anonymous Call • Block Caller ID • Busy Lamp Field • Call Appearance • Call Forward • Call Park Monitor • Call Waiting • Conference • Disposition Code • Do Not Disturb • Do Not Ring • Exec Filter On/Off • Exec Assistant • Hold • Hoteling • Line Monitor • Next Tab • Message Status • Page Group 1 • Page Group 2 • Presence Monitor • Security Class • Speed Dial • Transfer • Action URL • Add To Conference • Blind Transfer 2 • Left Next Tab 	Call Appearance

Right Line Keys Settings

Parameter	Description	Default Setting
Service	The Voice Service on which to apply the selected function. Choice of: <ul style="list-style-type: none"> • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	None
Number	The number, extension, or user ID of the entity associated with the assigned function.	None
Name	A display name for the entity associated with the assigned function.	None
Group	ID of the Custom Line Key Group to use to form the contents of the line key window.	None
Style	ID of the Custom Line Key Style to use to format the contents of the line key window.	None
MaxCalls	Maximum number of calls this key should handle if the function is Call Appearance.	1
Status	This read-only status parameter shows the last number to ring the phone.	

Programmable Keys Web Page

The following configuration parameters are available on this page.

Programmable Keys Settings

Parameter	Description	Default Setting
<i>Key n (VoiceService.1.Phone.ProgKey.n.), n = 1 – 2 for VVX 150 and VVX 250, = 1 – 3 for VVX 350 and VVX 450</i>		
Function	Function assigned to this key. Can be set for each Programmable Key. Choice of: <ul style="list-style-type: none"> • Auto Answer • Intercom • ACD Sign On/Off • Blind Transfer • Block Anonymous Call • Block Caller ID • Busy Lamp Field • Call Appearance • Call Forward • Call Park Monitor • Call Waiting • Conference • Disposition Code • Do Not Disturb • Do Not Ring • Exec Filter On/Off • Exec Assistant • Hold • Hoteling • Line Monitor • Next Tab • Message Status • Page Group 1 • Page Group 2 • Presence Monitor • Security Class • Speed Dial • Transfer • Action URL • Add To Conference • Blind Transfer 2 • Left Next Tab 	Transfer

Programmable Keys Settings

Parameter	Description	Default Setting
Service	The Voice Service on which to apply the selected function. Choice of: <ul style="list-style-type: none"> • SP1 • SP2 • SP3 • SP4 • SP5 • SP6 • OBiTALK 	None
Number	The number, extension, or user ID of the entity associated with the assigned function.	None
Name	A display name for the entity associated with the assigned function.	None
Group	ID of the Custom Line Key Group to use to form the contents of the line key window.	None
Style	ID of the Custom Line Key Style to use to format the contents of the line key window.	None
Status	This read-only status parameter shows any applicable status for the assigned function.	

LED Settings Web Page

The following table lists the meanings for LED default settings. When more than one LED setting is listed, it means that the LED is cycling between those states.

For example, **R50, X50** means the LED is flashing red (cycling between red and off).

LED Meanings

LED Default Setting	Description
R	LED is red.
X	LED is turned off.
<i>n, nn, nnn, nnnn</i>	Where <i>n</i> represents a number from 0 to 9, any string of <i>n</i> represents that value in ms.

The following table lists the meanings for LED displays on line keys.

LED Settings

Parameter	Description	Default Setting
<i>Call State (VoiceService.1.Phone.LED.Call.)</i>		
Idle	No call.	X

LED Settings

Parameter	Description	Default Setting
Dialing	Dialing a number.	G
Trying	Trying an outgoing call before the called party rings.	G
PeerRingng	Called party is ringing.	G50, X50
Connected	Call connected.	G
Ringng	Incoming call ringing.	R50, X50
Holding	Holding the call.	R500, X500
Error	Call failed.	G500, X500
CallParked	A call is parked on the bound extension.	R100, X1000
ServiceDown	Service is down.	X
SCA (VoiceService.1.Phone.LED.SCA.)		
Idle	SCA not used by anyone.	X
Seized	SCA has been seized by another user.	R150, X150
Trying	Another user is trying an outgoing call on the SCA.	R
PeerRingng	The call peer is ringing.	R
Connected	Call connected.	R
Held	Holding the call.	G500, X500
PrivateHeld	Another is user is holding the call as private.	R500, X500
ServiceDown	SCA Service is down.	X
BLF (VoiceService.1.Phone.LED.BLF.)		
Idle	Monitored extension is idle.	X
CallParked	A call is parked against the monitored extension.	R100, X1000
Ringng	Monitored extension is ringing.	R50, X50
Busy	Monitored extension is on a call.	R
Holding	Monitored extension is holding a call.	R500, X500
ServiceDown	BLF Service is down.	X
Service State (VoiceService.1.Phone.LED.Service.)		
Idle	No calls.	G
InUse	One or more calls connected or trying.	G500, X500
Ringng	One or more incoming calls ringing.	R50, X50

LED Settings

Parameter	Description	Default Setting
Holding	One or more calls holding.	R500, X500
ServiceDown	Service is down.	X1000, G1000
<i>ACD Agent State (VoiceService.1.Phone.LED.ACD.)</i>		
LoggedOff	Agent has logged off.	X
Available	Agent logged on and is available.	G
Unavailable	Agent logged on but isn't available.	R
WrappingUp	Agent logged on but is wrapping up last call.	R500, X500
ServiceDown	Service is down.	X
<i>Presence (VoiceService.1.Phone.LED.Presence.)</i>		
Offline	Same as Invisible.	X
Online	Online or Available.	G
Busy	Same as DND or Do Not Disturb.	R
Away	Away.	G500, X500
ExtendedAway	Same as XA.	R500, X500
ServiceDown	Service is down.	X
<i>Feature State (VoiceService.1.Phone.LED.FeatureKey.)</i>		
AnonymousCallEnabled	Anonymous Call feature is enabled.	R
AnonymousCallDisabled	Anonymous Call feature is disabled.	X
AnonymousCallServiceDown	Anonymous Call service is down.	X
AnonymousCallBlockEnabled	Block Anonymous Callers feature is enabled.	R
AnonymousCallBlockDisabled	Block Anonymous Callers feature is disabled.	X
AnonymousCallBlockServiceDown	Block Anonymous Callers service is down.	X
AutoAnswerIntercomEnabled	Auto Answer Intercom feature is enabled.	X
AutoAnswerIntercomDisabled	Auto Answer Intercom feature is disabled.	R
CallForwardEnabled	Call Forward feature is enabled.	R

LED Settings

Parameter	Description	Default Setting
CallForwardDisabled	Call Forward feature is disabled.	X
CallForwardServiceDown	Call Forward service is down.	X
CallParkedYes	Call is parked on line.	R
CallParkedNo	No Call parked on line.	X
CallParkMonitorServiceDown	Call Park Monitor service is down.	X
CallWaitingEnabled	Call Waiting feature is enabled.	X
CallWaitingDisabled	Call Waiting feature is disabled.	R
DoNotDisturbEnabled	Do Not Disturb feature is enabled.	R
DoNotDisturbDisabled	Do Not Disturb feature is disabled.	X
DoNotDisturbServiceDown	Do Not Disturb service is down.	X
DoNotRingEnabled	Do Not Ring feature is enabled.	R
DoNotRingDisabled	Do Not Ring feature is disabled.	X
ExecFilterEnabled	Executive (Call) Filter feature is enabled.	R
ExecFilterDisabled	Executive (Call) Filter feature is disabled.	X
ExecFilterServiceDown	Executive (Call) Filter service is down.	X
ExecAssistEnabled	Executive Assistant feature is enabled.	R
ExecAssistDisabled	Executive Assistant feature is disabled.	X
ExecAssistDivertOn	Executive Assistant Call Divert is on.	R100, X500
ExecAssistServiceDown	Executive Assistant Call Divert is off.	X
HotelingGuestLoggedOn	Hoteling Guest has logged on.	R
HotelingGuestLoggedOff	Hoteling Guest has logged off (same as No Hoteling Guest has logged on).	X
HotelingServiceDown	Hoteling Guest logon/logoff service is down.	X
NewMessagesWaitingYes	New messages waiting.	R
NewMessagesWaitingNo	No new messages waiting.	X

LED Settings

Parameter	Description	Default Setting
MWIServiceDown	MWI (message waiting indication) service is down.	X
PageGroupJoined	Joined Page Group.	R
PageGroupLeft	Left Page Group.	X
PageGroupMeTalking	Currently I am talking to the page group.	R500, X500
PageGroupThemTalking	Currently someone else is talking to the page group.	R50, X50
SecurityClassServiceDown	Security Classification service is down.	X
VMWI Lamp (VoiceService.1.Phone.LED.VMWI.)		
NewMessagesWaiting Yes	New messages available.	R
NewMessagesWaiting No	No new messages available	X
Disabled	VMWI is disabled.	X
Ringling	Phone is ringing.	None

Soft Keys Web Page

The following configuration parameters are available on this page.

Soft Keys Settings

Parameter	Description	Default Setting
Soft Key Set (VoiceService.1.Phone.SoftKeySet.)		
Home	Home screen.	redial, cfa, dnd, missed lines
SCAInUse	SCA being used by another phone.	sca.barge , sca.monitor , , newcall
CallParked	Call is parked.	pickup, , , newcall
CallConnected	Call is connected.	end, hold, totrs, tospk, conf, transfer, privhold, park, dispcode, escalate, trace, rec.start, rec.stop, rec.pause, rec.resume, tomobile
CallHolding	Call is holding.	end, resume, add2conf, conf, transfer, park, dispcode, escalate, trace, rec.start, rec.stop, rec.pause, rec.resume, tomobile

Soft Keys Settings

Parameter	Description	Default Setting
Dialtone	Dial tone playing.	redial,phbk,mode,lines
Dialing	Dialing a number.	redial,dial,mode,backspace
OnDialing	Dialing a number while on-hook.	switch.line,dial,mode,backspace
Ringing	Incoming call is ringing.	answer,reject,ignore
CallConnecting	Trying outgoing or called party is ringing.	end,,,newcall
ConfTrying	Calling 2nd conferee.	end
ConfRinging	2nd Conferee ringing.	end,,conf.now
ConfConnected	Connected with 2nd conferee.	end,hold,resume,conf.now
ConfBridgeControl	Connected with an external conference bridge.	end,deaf,mute
Conferencing	Merged conference call item.	end,split,newcall
XferTrying	Calling transfer target.	end
XferRinging	Transfer target ringing.	end,,xfer.now
XferConnected	Connected with transfer target.	end,hold,resume,xfer.now
NetServices	Network services top level.	ns.bci,ns.dnd,ns.cfa,ns.acd
BLFCall	BLF Call options.	blf.call,blf.answer,blf.pickup,blf.barge,blf.coach,blf.monitor
Settings	Settings menu.	reboot,reset,fwupdate

LDAP Search Web Page

The following configuration parameters are available on this page.

LDAP Search Settings

Parameter	Description	Default Setting
IP Phone – LDAP – Server (X_DeviceManagement.LDAPServer.)		
Host	IP address or domain name of host, with optional ldap:// or ldaps:// scheme prefix.	None.
Port	LDAP Server listening (TCP) port. The standard port is 389 for ldap:// and 636 for ldaps://. If the port value is 0 or blank, the phone uses the corresponding standard port.	0
Password	Password for authentication with the LDAP server.	None.

LDAP Search Settings

Parameter	Description	Default Setting
<i>IP Phone – LDAP – Search Parameters (X_DeviceManagement.LDAPSearch.)</i>		
BindDN	A Distinguished Name (DN) that is authorized to use the LDAP service.	None.
SearchBase	Specifies the starting point of the LDAP search.	None.
ProtocolVersion	Protocol version. Choice of: <ul style="list-style-type: none"> • 2 • 3 	3
TLS_ReqCert	Controls whether to verify the server's certificate on a TLS connection. Choice of: <ul style="list-style-type: none"> • never • demand 	never
ResultsPerPage	Specifies how many results to display on screen per page. Choice of: <ul style="list-style-type: none"> • 20 • 40 • 60 • 80 • 100 	20
DefaultSearchFilter	Default search filter to append to each search.	(objectclass=*)
QueryFields	A comma-separated list of user input LDAP attributes/Display-Name to form the query filter.	givenName/First Name,sn/Last Name,telephoneNumber/Tel,mobile/Mobile,homePhone/Home
ResultFields	Comma-separated list of LDAP fields to display for each entry of the search result.	cn/Common Name,sn/Last Name,givenName/First Name,telephoneNumber/Tel/y,mobile/Mobile/y,homePhone/Home/y
NameFieldPreference	Comma-separated list of LDAP attributes to be used as the Caller ID Name to display on screen, ordered by preference.	cn,givenName sn
NumberFieldPreference	Comma-separated list of LDAP attributes to be used as the Number to display on screen and to call by default, ordered by preference.	telephoneNumber,mobile,homePhone
PhotoFieldPreference	Comma-separated list of LDAP attributes to be used as the photo to display on screen, ordered by preference.	thumbnailPhoto
SortByAttribute	The LDAP attribute to use for sorting the search results by the server.	cn

LDAP Search Settings

Parameter	Description	Default Setting
<i>IP Phone – LDAP – SASL Authentication Parameters (X_DeviceManagement.LDAPSearchSASL.)</i>		
SASL_AuthMethod	Method to use for SASL authentication. If the value is <code>Disabled</code> , the phone uses Simple authentication only. Choice of: <ul style="list-style-type: none"> Disabled Plain MD5 	Disabled
SASL_AuthCID	The authentication ID for SASL authentication.	None.

Line Key Customization Web Page

The following configuration parameters are available on this page.

Line Key Customization Settings

Parameter	Description	Default Setting
<i>Call Appearance (VoiceService.1.Phone.CstmLineKey.Call.)</i>		
Enable	Boolean. Enables using customized feature key display format for Call Appearance key function.	False
TextLine1	Use this text for line 1 when the key has no calls.	@text1
TextLine2	Use this text for line 2 when the key has no calls.	@text2
TextLine1InCall	Use this text for line 1 when the key has one or more calls.	@text1
TextLine2InCall	Use this text for line 2 when the key has one or more calls.	@text2
<i>Busy Lamp Field (VoiceService.1.Phone.CstmLineKey.BLF.)</i>		
Enable	Boolean. Enables using customized feature key display format for Busy Lamp Field key function.	False
TextLine1	Use this text for line 1 when the monitored extension has no calls.	@text1
TextLine2	Use this text for line 2 when the monitored extension has no calls.	@text2
TextLine1InCall	Use this text for line 1 when the monitored extension has one or more calls.	@text1
TextLine2InCall	Use this text for line 2 when the monitored extension has one or more calls.	@text2
<i>Speed Dial (VoiceService.1.Phone.CstmLineKey.SpeedDial.)</i>		
Enable	Boolean. Enables using customized feature key display format for Speed Dial key function.	False

Line Key Customization Settings

Parameter	Description	Default Setting
TextLine1	Information to show on line 1 of a customized Speed Dial line key.	@text1
TextLine2	Information to show on line 2 of a customized Speed Dial line key.	@text2
Action URL (VoiceService.1.Phone.CstmLineKey.ActionURL.)		
Enable	Boolean. Enables using a customized feature key display format for ActionURL key function.	False
TextLine1	Information to show on line 1 of a customized ActionURL line key.	@text1
TextLine2	Information to show on line 2 of a customized ActionURL line key.	@text2
Group n (VoiceService.1.Phone.CstmLineKey.Group.n.), n = 1, 2, 3, 4		
ID	The group ID of a customized line key.	None
TextLine1	Information to show on line 1 of a line key that is in this group.	@text1
TextLine2	Information to show on line 2 of a line key that is in this group.	@text2
Icon	The icon to show for a line key that is in this group.	@icon

Screen Item Customization Web Page

The Screen Item Customization web page is available only for the VVX 250, VVX 350, and VVX 450 phones.

Screen Item Customization Settings

Parameter	Description	Default Setting
Screen Item (VoiceService.1.Phone.Screen.)		
RingItem	The value is a <ScreenItem> XML that customizes the format of the Ringing Calls display on the phone screen.	None
CallItem	The value is a <ScreenItem> XML that customizes the format of the Current Calls display on the phone screen.	None
CallTransferItem	The value is a <ScreenItem> XML that customizes the format of the Call Transfer display on the phone screen. Unlike the last two items which are items in a table, there is only 1 CallTransferItem .	None
NetDirItem	Customization XML that renders items displayed in a Network Directory.	None

Screen Item Customization Settings

Parameter	Description	Default Setting
ScreenSaverCustomContents	The value is a <ScreenItem> XML that customizes the contents to be shown on top of the current screen saver picture when the screen saver is being shown.	<pre data-bbox="1143 369 1414 842"><ScreenItem> <time align='center' ypos='60' font='@gfont-bold' size='60' format='@timeFmt' shadow='2,1,0x444444' /> <date align='center' ypos='134' font='@gfont' size='24' format='w &#160;&#160; mn/d/y' shadow='2,1,0x444444' /> </ScreenItem></pre>
HomeScreenCustomContents	<p data-bbox="493 867 1105 951">The value is a <ScreenItem> XML script that defines the size, position, and contents of the rectangular area on the Home Screen.</p> <p data-bbox="493 961 1105 1014">If this value is not specified or invalid, the real time clock displays.</p>	None

Screen Item Customization Settings

Parameter	Description	Default Setting
TitleBarStyle	Customized style to render the Title Bar (the top Status Bar) of the phone.	<pre><!-- TitleBarStyle size='12'> <time xpos='2'/><date xpos='68' /> <text xpos='2' ypos='16'>\$prod.obinu mber (\$prod.mac)</text> <Notifications list='*' width='\$eval(\$sys.hid etabs?@dispW, (\$sys.nt abs==4)?@lkpg1,@lkpg1 v2)' align='right' xpos='0'> </Notifications> </TitleBarStyle --></pre>
SoftKeyStyles	Contains a list of as many as 32 <style> elements to be used to format a soft key. The enclosing root element must be <SoftKeyStyles>. To use a style, specify a style attribute in the custom soft key (in a soft key set parameter), with the value equal to id attribute of the corresponding <style> element. If no style attribute is specified or the given style isn't found, the <style> with id="default" is used for the soft key, if one is found. Otherwise the internally defined style is used for the soft key.	None

Codec Profiles

The Codec Profile parameters web pages include one page for each codec profile, as shown:

Codec Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Codec Profile Settings

Parameter	Description	Default Setting
G711U Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.1.), n = 1, 2 for X = A, B respectively		
Codec	Codec name.	PCMU
BitRate	Read-only parameter for bit rate in bits/sec.	64000
Enable	Enables this codec.	True

Codec Profile Settings

Parameter	Description	Default Setting
SilenceSuppression	Enables silence suppression for this codec.	False
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	3
PayloadType	Dynamic payload type for this codec.	0

G711A Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.2.), n = 1, 2 for X = A, B respectively

Codec	Codec name.	PCMA
BitRate	Read-only parameter for bit rate in bits/sec.	64000
Enable	Enables this codec.	True
SilenceSuppression	Enables silence suppression for this codec.	False
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	4
PayloadType	Dynamic payload type for G711-alaw codec.	8

G729 Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.3.), n = 1, 2 for X = A, B respectively

Note: Opus is not compatible with G.729 and iLBC. If you set Opus to the highest priority, G.729 and iLBC are not published; if you set G.729 and iLBC to the highest priority, Opus is not published. The VVX 150 doesn't support the OPUS codec.

Codec	Codec name.	G729
BitRate	Read-only parameter for bit rate in bits/sec.	8000
Enable	Enables this codec.	True
SilenceSuppression	Enables silence suppression for this codec.	False
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	5
PayloadType	Dynamic payload type for G.729 codec.	18

iLBC Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.8.), n = 1, 2 for X = A, B respectively

Note: Opus is not compatible with G.729 and iLBC. If you set Opus to the highest priority, G.729 and iLBC are not published; if you set G.729 and iLBC to the highest priority, Opus is not published.

Codec	Codec name.	iLBC
BitRate	Bit rate in bits/sec. Choice of: <ul style="list-style-type: none"> • 13333 bps • 15200 bps 	13333
Enable	Enables this codec.	False

Codec Profile Settings

Parameter	Description	Default Setting
SilenceSuppression	Enables silence suppression for this codec.	False
PacketizationPeriod	Packet size in ms.	30
Priority	Priority assigned to this codec (1 is the highest).	10
PayloadType	Dynamic payload type for this codec. Valid range is 96–127.	98

G722 Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.9.), n = 1, 2 for X = A, B respectively

Codec	Codec name for this RTP event, as used in SDP.	G722
BitRate	Read-only parameter for bit rate in bits/sec.	64000
Enable	Enables this codec.	True
SilenceSuppression	Enables silence suppression for this codec.	False
PacketizationPeriod	Packet size in ms.	20
Priority	Priority assigned to this codec (1 is the highest).	2
PayloadType	Dynamic payload type for this codec.	9

OPUS Codec (VoiceService.1.VoiceProfile.1.Line.n.Codec.List.10.), n = 1, 2 for X = A, B respectively

Note: Opus is not compatible with G.729 and iLBC. If you set Opus to the highest priority, G.729 and iLBC are not published; if you set G.729 and iLBC to the highest priority, Opus is not published. The VVX 150 doesn't support the OPUS codec.

Codec	Codec name for this RTP event, as used in SDP. Note that some servers may expect this to be lower case <code>opus</code> .	OPUS
BitRate	Variable.	20000
Enable	Enables this codec.	True
SilenceSuppression	Enables silence suppression for this codec.	False
PacketizationPeriod	Packet size in milliseconds.	20
Priority	Priority assigned to this codec (1 is the highest).	2
PayloadType	Dynamic payload type for this codec.	109
UseInbandFEC	Enables in-band forward error correction.	False

Telephone Event (VoiceService.1.VoiceProfile.1.Line.n.Codec.X_TelephoneEvent.), n = 1, 2 for X = A, B respectively

Codec	Codec name for this RTP event, as used in SDP.	telephone-event
Enable	Enables this codec.	True
PayloadType	Dynamic payload type to be used for RFC2833 telephone (DTMF) events. Valid range is 96–127.	101

Codec Profile Settings

Parameter	Description	Default Setting
<i>Encap RTP (VoiceService.1.VoiceProfile.1.Line.n.Codec.X_EncapRTP.), n = 1, 2 for X = A, B respectively</i>		
Codec	Codec name. This codec encapsulates RTP packets during a packet loopback call.	encaprtsp
PayloadType	Dynamic payload type for this codec. Valid range is 96–127.	107
<i>Loopback Primer (VoiceService.1.VoiceProfile.1.Line.n.Codec.X_LoopbackPrimer.), n = 1, 2 for T X = A, B respectively</i>		
Codec	Codec name. The phone uses this codec when it acts as a media loopback mirror and before receiving any packets from the loopback source during a media loopback call.	loopbkprimer
PayloadType	Dynamic payload type for this codec. Valid range is 96–127.	108

Tone Settings

The Tone Profile parameters web pages include one page for each tone profile, as shown:

Tone Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Tone Profile Settings

Parameter	Description	Default Setting
<i>Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.1.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Dial Tone.	
TonePattern	Obihai Tone Pattern Script.	350-18, 440-18;20
<i>Ringback Tone (VoiceService.1.VoiceProfile.n.Tone.Description.2.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Ringback Tone.	
TonePattern	Obihai Tone Pattern Script.	440-18, 480-18;-1; (2+4)
<i>Busy Tone (VoiceService.1.VoiceProfile.n.Tone.Description.3.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Busy Tone.	
TonePattern	Obihai Tone Pattern Script.	480-18, 620-18;10; (.5+.5)
<i>Reorder Tone (VoiceService.1.VoiceProfile.n.Tone.Description.4.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Reorder tone or Fast busy.	
TonePattern	Obihai Tone Pattern Script.	480-18, 620-18;10; (.25+.25)
<i>Confirmation Tone (VoiceService.1.VoiceProfile.n.Tone.Description.5.), n = 1, 2 for X = A, B respectively</i>		

Tone Profile Settings

Parameter	Description	Default Setting
ToneName	Confirmation Tone.	
TonePattern	Obihai Tone Pattern Script.	600-18;1; (.2+.2)
<i>Holding Tone (VoiceService.1.VoiceProfile.n.Tone.Description.6.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Holding Tone played when peer holding the call.	
TonePattern	Obihai Tone Pattern Script.	800-18;30; (.1+10)
<i>Second Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.7.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Second Dial Tone played when dialing second call in a 3-way call.	
TonePattern	Obihai Tone Pattern Script.	385-18,484-18;20
<i>Stutter Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.8.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Stutter Dial Tone.	
TonePattern	Obihai Tone Pattern Script.	350-18,440-18;20;2 (.1+.1); ()
<i>Howling Tone (VoiceService.1.VoiceProfile.n.Tone.Description.9.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Howling Tone for off-hook warning.	
TonePattern	Obihai Tone Pattern Script.	480+3,620+3;10; (.125+.125)
<i>Prompt Tone (VoiceService.1.VoiceProfile.n.Tone.Description.10.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Prompt Tone to prompt user to enter a number for configuration, such as speed dial.	
TonePattern	Obihai Tone Pattern Script.	480-16;20
<i>Call Forward Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.11.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Call Forward Dial Tone (Special dial tone to indicate call-forward-all active).	
TonePattern	Obihai Tone Pattern Script.	350-18,440-18;20; (.2+.2)
<i>DND Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.20.), n = 1, 2 for X = A, B respectively</i>		
ToneName	DND Dial Tone (Special dial tone to indicate DND active).	
TonePattern	Obihai Tone Pattern Script.	350-18,440-18;20; (.2+.2)
<i>Conference Tone (VoiceService.1.VoiceProfile.n.Tone.Description.12.), n = 1, 2 for X = A, B respectively</i>		
ToneName	Conference Tone (Indicates conference has started).	

Tone Profile Settings

Parameter	Description	Default Setting
TonePattern	Obihai Tone Pattern Script.	350-16;10; (.1+.1, .1+9.7)
SIT Tone 1 (VoiceService.1.VoiceProfile.n.Tone.Description.13.), n = 1, 2 for X = A, B respectively		
ToneName	Special Information Tone 1.	
TonePattern	Obihai Tone Pattern Script.	985-16,1428-16,1777-16;20; (1/.380+0,2/.380+0,4/.380+ 0,0/0+4)
SIT Tone 2 (VoiceService.1.VoiceProfile.n.Tone.Description.14.), n = 1, 2 for X = A, B respectively		
ToneName	Special Information Tone 2.	
TonePattern	Obihai Tone Pattern Script.	914-16,1371-16,1777-16;20; (1/.274+0,2/.274+0,4/.380+ 0,0/0+4)
SIT Tone 3 (VoiceService.1.VoiceProfile.n.Tone.Description.15.), n = 1, 2 for X = A, B respectively		
ToneName	Special Information Tone 3.	
TonePattern	Obihai Tone Pattern Script.	914-16,1371-16,1777-16;20; (1/.380+0,2/.380+0,4/.380+ 0,0/0+4)
SIT Tone 4 (VoiceService.1.VoiceProfile.n.Tone.Description.16.), n = 1, 2 for X = A, B respectively		
ToneName	Special Information Tone 4.	
TonePattern	Obihai Tone Pattern Script.	985-16,1371-16,1777-16;20; (1/.380+0,2/.380+0,4/.380+ 0,0/0+4)
Outside Dial Tone (VoiceService.1.VoiceProfile.n.Tone.Description.17.), n = 1, 2 for X = A, B respectively		
ToneName	Outside Dial Tone.	
TonePattern	Obihai Tone Pattern Script.	385-16;10
R-Command Tone (VoiceService.1.VoiceProfile.n.Tone.Description.18.), n = 1, 2 for X = A, B respectively		
ToneName	R-Command Tone.	
TonePattern	Obihai Tone Pattern Script.	400-16;5
Paging Tone (VoiceService.1.VoiceProfile.n.Tone.Description.19.), n = 1, 2 for X = A, B respectively		
ToneName	Paging Tone (The short tone played before playing the caller's voice).	
TonePattern	Obihai Tone Pattern Script.	480-16;1; (.2+.2)

Ring Settings

The Ring Profile parameters web pages include one page for each ring profile, as shown:

Ring Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Ring Settings

Parameter	Description	Default Setting
Call Waiting Tone 1 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively		
ToneName	Distinctive Call Waiting Tone 1.	Bellcore-dr1
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.25+10)
Call Waiting Tone 2 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively		
ToneName	Distinctive Call Waiting Tone 2.	Bellcore-dr2
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.1+.1, .3+.1, .1+10)
Call Waiting Tone 3 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively		
ToneName	Distinctive Call Waiting Tone 3.	Bellcore-dr3
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.1+.1, .1+10)
Call Waiting Tone 4 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively		
ToneName	Distinctive Call Waiting Tone 4.	Bellcore-dr4
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.1+.1, .1+.1, .1+10)
Call Waiting Tone 5 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively		
ToneName	Distinctive Call Waiting Tone 5.	Bellcore-dr5
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+.1, .1+.1, .3+10)
Call Waiting Tone 6 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively		
ToneName	Distinctive Call Waiting Tone 6.	User-dr1
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.1+.1, .3+.2, .3+10)
Call Waiting Tone 7 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively		
ToneName	Distinctive Call Waiting Tone 7.	User-dr2
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+.1, .3+.1, .1+10)

Ring Settings

Parameter	Description	Default Setting
<i>Call Waiting Tone 8 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively</i>		
ToneName	Distinctive Call Waiting Tone 8.	User-dr3
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+2)
<i>Call Waiting Tone 9 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively</i>		
ToneName	Distinctive Call Waiting Tone 9.	User-dr4
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+2)
<i>Call Waiting Tone 10 (VoiceService.1.VoiceProfile.n.Tone.Description.21.), n = 1,2 for X=A,B respectively</i>		
ToneName	Distinctive Call Waiting Tone 10.	User-dr5
TonePattern	Obihai Tone Pattern Script.	440-18;30; (.3+2)
<i>Ring Pattern 1 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.1.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 1	Bellcore-dr1
RingPattern	OBIHAI Ring Pattern Script.	60; (2+4)
<i>Ring Pattern 2 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.2.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 2	Bellcore-dr2
RingPattern	OBIHAI Ring Pattern Script.	60; (.3+.2, 1+.2, .3+4)
<i>Ring Pattern 3 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.3.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 3	Bellcore-dr3
RingPattern	OBIHAI Ring Pattern Script.	60; (.8+.4, .8+4)
<i>Ring Pattern 4 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.4.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 4	Bellcore-dr4
RingPattern	OBIHAI Ring Pattern Script.	60; (.4+.2, .3+.2, .8+4)
<i>Ring Pattern 5 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.5.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 5	Bellcore-dr5
RingPattern	OBIHAI Ring Pattern Script.	60; (.2+.2, .2+.2, .2+.2, 1+4)
<i>Ring Pattern 6 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.6.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 6	User-dr1
RingPattern	OBIHAI Ring Pattern Script.	60; (.2+.4, .2+.4, .2+4)
<i>Ring Pattern 7 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.7.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 7	User-dr2

Ring Settings

Parameter	Description	Default Setting
RingPattern	OBIHAI Ring Pattern Script.	60; (.4+.2, .4+.2, .4+4)
<i>Ring Pattern 8 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.8.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 8	User-dr3
RingPattern	OBIHAI Ring Pattern Script.	60; (.25+9.75)
<i>Ring Pattern 9 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.9.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 9	User-dr4
RingPattern	OBIHAI Ring Pattern Script.	60; (.25+9.75)
<i>Ring Pattern 10 (VoiceService.1.VoiceProfile.1.Line.n.Ringer.Description.10.), n = 1,2 for X=A,B respectively</i>		
RingName	Ring Pattern 10	User-dr5
RingPattern	OBIHAI Ring Pattern Script.	60; (.25+9.75)

Star Code Profiles

The Star Code Profile parameters web pages include one page for each Star Code profile, as shown:

Star Code Profile X Web Page (X = A, B)

The following configuration parameters are available on each page.

Star Code Settings

Parameter	Description	Default Setting
Code1	Redial Star Code.	*07, Redial, call(\$Ldn)
Code2	Call Return Star Code.	*69, Call Return, call(\$Lcn)
Code3	Block Caller ID (Persistent) Star Code.	*81, Block Caller ID, set(\$Bci,1)
Code4	Unblock Caller ID (Persistent) Star Code.	*82, Unblock Caller ID, set(\$Bci,0)
Code5	Block Caller ID Once Star Code.	*67, Block Caller ID Once, set(\$Bci1,1)
Code6	Unblock Caller ID Once Star Code.	*68, Unblock Caller ID Once, set(\$Ubcil,1)
Code7	Call Forward Unconditional Star Code.	*72, Cfdw All, coll(\$Cfan), set(\$Cfa,1)
Code8	Disable Call Forward Unconditional Star Code.	*73, Disable Cfdw All, set(\$Cfa, 0)

Star Code Settings

Parameter	Description	Default Setting
Code9	Call Forward on Busy Star Code.	*60, C fwd Busy, coll(\$Cfbn), set(\$Cfb,1)
Code10	Disable Call Forward on Busy Star Code.	*61, Disable C fwd Busy, set(\$Cfb, 0)
Code11	Call Forward on No Answer Star Code.	*62, C fwd No Ans, coll(\$Cfnn), set(\$Cfn,1)
Code12	Disable Call Forward on No Answer Star Code.	*63, Disable C fwd No Ans, set(\$Cfn,0)
Code13	Block Anonymous Calls Star Code.	*77, Block Anonymous Call, set(\$Bac,1)
Code14	Unblock Anonymous Calls Star Code.	*87, Unblock Anonymous Call, set(\$Bac,0)
Code15	Enable Call Waiting Star Code.	*56, Enable Call Waiting, set(\$Cwa,1)
Code16	Disable Call Waiting Star Code.	*57, Disable Call Waiting, set(\$Cwa,0)
Code17	Do Not Disturb Star Code.	*78, Do Not Disturb, set(\$Dnd,1)
Code18	Disable Do Not Disturb Star Code.	*79, Disable DND, set(\$Dnd,0)
Code19	Repeat Dial Star Code.	*66, Repeat Dial, rpdi(\$Ldn)
Code20	Disable Repeat Dial Star Code.	*86, Disable Repeat Dial, rpdi
Code21	Set Speed Dial Star Code.	*74(x xx), Set Speed Dial, coll(\$Spd[\$Code])
Code22	Check Speed Dial Star Code.	*75(x xx), Check Speed Dial, say(\$Spd[\$Code])
Code23	Loopback Media Star Code.	*03, Loopback Media, set(\$Lbm1,1)
Code24	Loopback RTP Star Code.	*04, Loopback RTP Packet, set(\$Lbp1,1)
Code25	Force G711u Codec Star Code.	*4711, Use G711 Only, set(\$Cdm1,3)
Code26	Force G729 Codec Star Code.	*4729, Use G729 Only, set(\$Cdm1,4)
Code27	Clear Speed Dial Star Code.	*76([1-9] [1-9]x), Clear Speed Dial, set(\$Spd[\$Code],)
Code28		
Code29	Barge In Star Code.	*96, Barge In, set(\$Bar1,1)
Code30		

Star Code Settings

Parameter	Description	Default Setting
Code31	Star Code entry 31.	*4722, Use G722 Only, set(\$Cdm1,512)
Code32	Star Code entry 32. Note: The VVX 150 doesn't support the OPUS codec.	*4678, Use OPUS Only, set(\$Cdm1,1024)

User Settings

The User Settings parameters web pages include the following pages:

- [User Preferences Web Page](#)
- [Speed Dials Web Page](#)
- [User Defined Digit Maps Web Page](#)

User Preferences Web Page

The following configuration parameters are available on this page.

User Preferences Settings

Parameter	Description	Default
User Preferences Settings		
Language	<p>Sets the language for the on-screen messages. Choice of:</p> <ul style="list-style-type: none"> • English-US • Deutsche (German) • Francais (French) • Español (Spanish) • English-UK • Dansk (Danish) • Suomalainen (Finnish) • Nederlands (Dutch) • Norsk (German) • Svenska (Swedish) • 繁體中文 (not available on VVX 150) • 한국어 (not available on VVX 150) • 日本語 (not available on VVX 150) • 简体中文 (not available on VVX 150) • Custom <p>Except for <code>Custom</code>, each built-in language corresponds to a dictionary file in that language. When the value is set to <code>Custom</code>, the corresponding customer dictionary file (cloud or local) is used instead. The file will download once per reboot because it is not cached. If that custom dictionary file is not found or has an error, the phone reverts to use <code>English-US</code> as the language.</p>	English-US
CustomLanguage	Specifies the language corresponding to a custom dictionary file installed on the phone.	None
TimeFormat	<p>Chooses the time display format on the title bar of the phone screen. Choice of:</p> <ul style="list-style-type: none"> • default • 1:35 pm • 13:35 • 1:35 	default
DateFormat	<p>Chooses the date display format on the title bar of the phone screen. Choice of:</p> <ul style="list-style-type: none"> • default • 1/31/2014 • 1/31/14 • 1/31 • 31/1/2014 • 31/1/14 • 31/1 • Jan 31, 2014 	default

User Preferences Settings

Parameter	Description	Default
AutoHomePage	Chooses the delay to automatically return the screen to the Home (idle) screen. Choice of: <ul style="list-style-type: none"> disabled 30 seconds 1 minute 10 minutes 1 hour 4 hours 	1 minute
Skin	Choose the skin (or style and color scheme) of the UI. Choice of: <ul style="list-style-type: none"> Tomáš Ulrik Tomáš-Wide Ulrik-Wide 	Tomáš
BackgroundPicture	A URL (HTTP or HTTPS) to a picture file or the path name of an internally stored picture.	Default
DefaultRingtone	A URL (HTTP or HTTPS) to a ringtone file (.wav or MP3 format) or the path name of an internally stored ringtone file.	/data/ringtones/Office A.wav
CallParkRingtone	A URL (HTTP or HTTPS) to a ringtone file (.wav or MP3 format) or the name of an internally short ring: ding or blop. It can be followed by two optional comma-separated parameter={value} pairs: <ul style="list-style-type: none"> duration={value}. {value} is total duration to play in seconds. Must be > 0. 0 means play forever. Default is 60. interval={value}. {value} is interval between ring bursts in seconds. Must be > 0. Default is 10. 	ding,interval=10,duration=60
DefaultFont	The default font used to render text on the phone screen.	ptsans
PopUpMoreSoftKey Options	If enabled, shows more softkey options in a popup window when user presses the More softkey when there are more than four options. Otherwise, show softkey options in a circular list when user presses the More softkey.	True
PackCallsOnDisplay	This option is applicable when one or more Call Keys (feature key that is assigned the Call Appearance function) have the MaxCalls parameter set to greater than 1. If the option is disabled, calls controlled by different Call Keys are shown on separate screens as you scroll through the calls. Otherwise, calls controlled by different Call Keys can be shown on the same screen (packed together).	True
LineKeyTabs	This option controls whether to show multiple tabs of (virtual) Line Keys on the screen.	True
LineKeyMaxTabs	Must be either 2 or 4. Applies only if LineKeyTabs = true.	4

User Preferences Settings

Parameter	Description	Default
LeftLineKeyTabs	This option controls whether to show multiple tabs of (virtual) Left Line Keys on the screen.	True
LeftLineKeyMaxTabs	Must be either 2 or 4. Applies only if LineKeyTabs = true.	4
DimScreenDelay	If greater than 0, this is the delay in seconds before dimming the screen when there are no activities. Dimming is done by setting the LCD brightness to the minimum value of 0. Any key press or new call restores the normal LCD brightness. This operation is independent of Screen Saver settings.	0
ScreenSaver	Turn the screen saver features on or off.	False
ScreenSaverDelay	Number of seconds of inactivity before starting screen saver.	10
ScreenSaverType	Choose the type of screen saver to use: <ul style="list-style-type: none"> • Slide Show • Slide Show (User Pictures Only) • Slide Show (Specified URL Only) • Turn Off Display 	Slide Show
ScreenSaverPicture URL	When ScreenSaverType is Slide Show (Specified URL Only), this parameter specifies the URL to get the pictures for the slide show. It can be an http:// or https:// URL to get the screen save picture(s), or the absolute path of an internal folder to get the screen saver pictures. If an http URL is used, the server may indicate 'no-cache' in the Cache-Control header of the http response to the phone to force the phone to download a fresh picture from the server on every screen picture refresh. You can use \$count macro in the http/https URL which is expanded into the number of refreshes when the URL is executed. It starts with 1, 2, 3, ..., max-count and the recycles. The max-count value can be specified as a comma-separated attribute in this URL also. The default is 0x7fffffff. For example: http://xyz.com/pic-\$count.png,max-count=4 http://xyz.com/pic-{\$count}.png,max-count=15	0x7fffffff
SlideShowInterval	The interval to refresh each picture in a slide show when screen saver is running.	5
ScreenSaverLock	Control whether dismissing the screen saver requires a passcode.	False
ScreenSaverShowCustomContents	Show the custom contents defined in the parameter ScreenSaverCustomContents (under the Screen Item Customization group), super-imposed on the current screen saver picture.	True
UnlockPasscode	Passcode to unlock the screen saver.	123456

User Preferences Settings

Parameter	Description	Default
LockIncomingCall	When the screen saver is activated and protected by a passcode, this option controls whether incoming calls to the phone can be answered without unlocking the screen.	False
Brightness	Screen brightness level.	6
PreferredAudioDevice	The preferred audio device (also known as an audio terminal) to use when answering a call by pressing a key (for example, a softkey or line key) that is not the speakerphone key or the headset key. Choice of: <ul style="list-style-type: none"> Speaker Headset 	Speaker
PreferredHeadsetDevice	The preferred headset to use (when more than one are available) when answering a call by pressing the headset key. Choice of: <ul style="list-style-type: none"> RJ9 Headset USB Headset 	RJ9 Headset
USBHeadsetSync	Enables signaling to the USB headset device when the headset or mute key on the phone is pressed.	True
EhsEnable	Enables using an EHS Headset. Choice of: <ul style="list-style-type: none"> Disabled Plantronics Jabra Sennheiser 	Disabled
RingerVolume	The ringer audio volume when ringing.	9
SpeakerVolume	The speakerphone audio playback volume.	10
HandsetVolume	The handset audio playback volume.	11
HandsetGain	The gain to apply to the handset audio input.	5
HeadsetVolume	The audio playback level on the RJ9 Headset.	10
HeadsetGain	The audio gain to apply to the RJ9 Headset input.	5
HeadsetUSBVolume	The audio playback level on the USB headset.	9
HeadsetUSBGain	The audio gain to apply to the USB Headset input.	5
HandsetENS	Enables Enhanced Noise Suppression (Acoustic Fence) for the handset.	False
HeadsetENS	Enables Enhanced Noise Suppression (Acoustic Fence) for the headset.	False
HeadsetAEC	Enables Acoustic Echo Cancellation for the headset.	False

User Preferences Settings

Parameter	Description	Default
IncomingPageAlertTone	Choose whether to play a beep when the phone answers an incoming Intercom/Page call. Choice of: <ul style="list-style-type: none"> none beep 	beep
UseEnhancedDialer	Enables enhanced dialer where a list of recently dialed number that partially match the currently entered digits is displayed below the dialer input box. You can highlight an entry in the list to make the call to.	True
DefaultLineKeyTab	The line key page (on the right side of the phone) to fall back on after some idle time.	1
DefaultLeftLineKeyTab	The line key page on the left side of the phone to fall back on after some idle time.	1
HideMainMenu	Enables hiding the Main menu on screen by default. Press the Home Key to toggle the Main menu on or off.	False
HideLineKeys	Hide or show Line Keys on screen by pressing the Home key.	False
ToggleMainMenuAndLineKeysTogether	Hide or show Main menu and Line Keys together by pressing the Home key.	False
ToggleCallsApp	Hide or show the Calls app when pressing the Home Key when there is one or more calls.	False
CustomMainMenu	Use customized Main menu.	False
AutoAnswerAutoMute	Enables auto-mute when the phone auto-answers an incoming Intercom/page call.	False
AutoAnswerOffWhenBusy	Automatically turns off auto-answer intercom calls when the phone is busy (if you are making a call or talking on a call).	False
ShowSharedCalls	Enables showing calls on a shared line that are active on other phones. Choice of: <ul style="list-style-type: none"> Always Never User Select 	Always

Phone Book Fields

Address	Enables the Address field.	False
Company	Enables the Company field.	False
Group	Enables the Group field.	False
Email	Enables the Email field.	True
FirstName	Enables the First Name field.	False
HomeNumber	Enables the Home Number field.	False

User Preferences Settings

Parameter	Description	Default
LastName	Enables the Last Name field.	False
MobileNumber	Enables the Mobile Number field.	True
Name	Enables the Name field.	True
Number	Enables the General Number field.	True
OfficeNumber	Enables the Work Number field.	True
Picture	Enables the Picture field.	True
Ringtone	Enables the Ringtone field.	True
Service	Enables the Service field.	True

Phone Book Settings

Name	The label to display for this item on the phone's main menu screen.	Contacts
ActionURL	The (HTTP or HTTPS) URL to execute when you select the Phone Book item from the phone's menu, if one is specified. Otherwise, the phone launches the built-in Phone Book application.	None
Groups	A comma-separated list of group names to group contacts within the built-in phone book.	Co-Workers, Customers, Family, Friends

Speed Dials Web Page

The following configuration parameters are available on this page.

Speed DialsSettings

Parameter	Description	Default Setting
N (Repeated for N = 1 – 99)	Speed Dial N	None

User Defined Digit Maps Web Page

For an explanation of this feature, please refer to the [User-defined Digit Maps](#) section in the [Call Controls](#) chapter.

User Defined Digit Map Settings

Parameter	Description	Default Setting
User Defined Digit Map1		
Label	A 2- to 16-character long label to reference this digit map in other digit maps and call routing rules. It must be alphanumeric, not containing any spaces, and different from other user-defined or built-in digit map labels.	ipd
DigitMap	A valid digit map.	(xx.<*:@>xx?x?<*:.>xx?x?<*:.>xx?x?<*: xx.<*:@>xx?x?<*:.>xx?x?<*:.>xx?x?<*:.>xx?x?<*:.>xx?x?<*::>xx?x?x?x?) Note: This default value supports IPv4 dialing.
User Defined Digit MapN (N = 2 – 10)		
Label	A 2- to 16-character long label to reference this digit map in other digit maps and call routing rules. It must be alphanumeric, not contain any spaces, and be different from other user-defined or built-in digit map labels.	None
DigitMap	A valid digit map.	None