# Configuring an OBi Voice Service Bridge to connect a legacy IP Phone

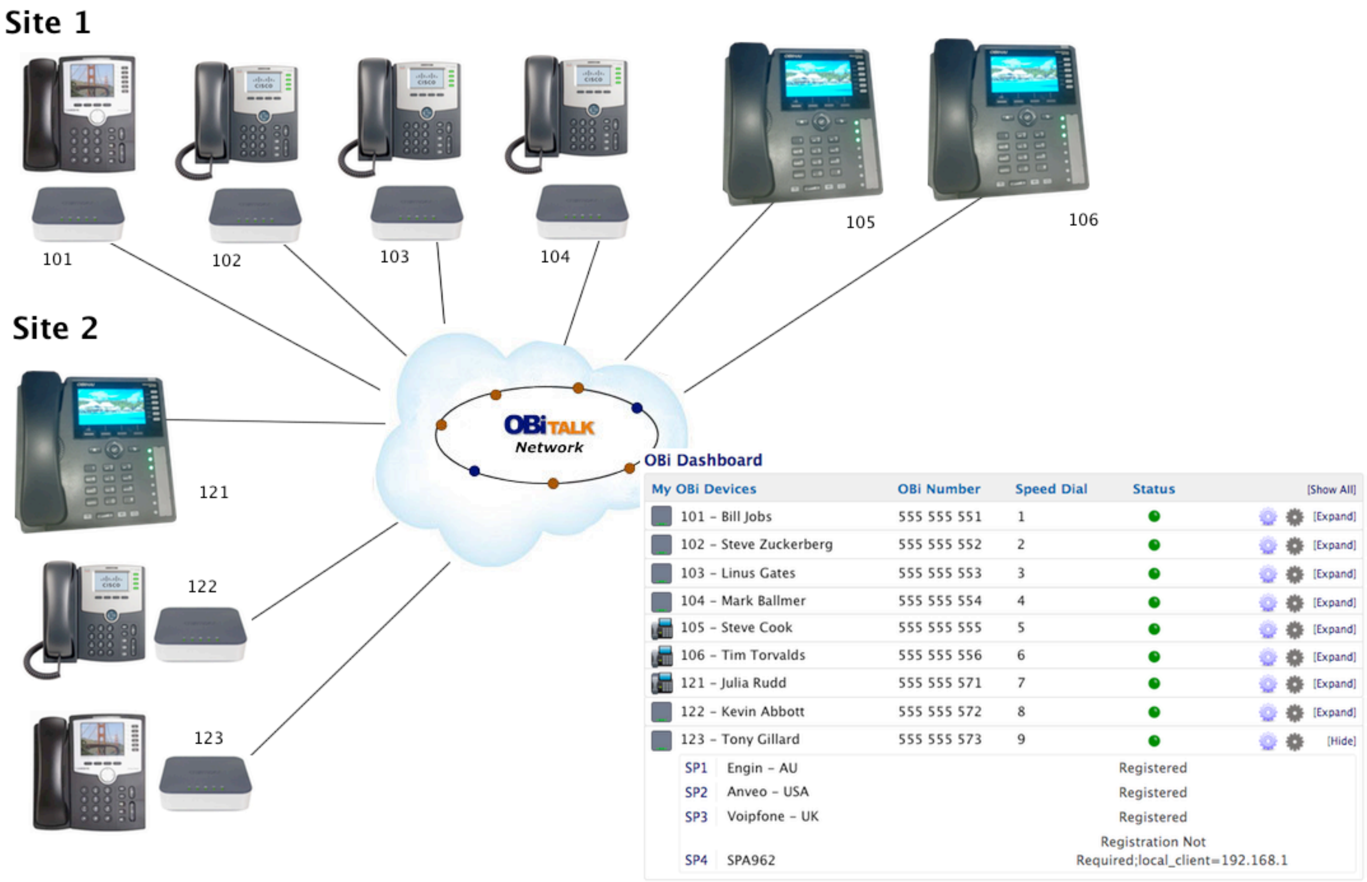
v1.0

## Table of Contents

## Why connect an IP Phone to an OBi Voice Service bridge?

- Easier management of services for legacy devices using the OBiTALK cloud portal
- Enable flexible call routing and multiple line features that the legacy device may lack
- Add Bluetooth calling, Wi-Fi or a PSTN Line to the device (via optional OBi accessories)
- Make it easy to integrate legacy IP Phones with OBi1032 and OBi1062 IP Phones
- Allows a legacy IP phone to take advantage of OBi inter-device call routing
- Adds OBiTALK device-to-device calling

Here's a sample topology mixing Cisco SPA504 and SPA962 IP Phones with OBI1062 IP Phones, managing calling features for all devices within the OBiTALK portal:



OBi Dashboard

| My OBi Devices | OBi Number | Speed Dial | Status | [Show All] |
|---|---|---|---|---|
| 101 – Bill Jobs | 555 555 551 | 1 | ● | [Expand] |
| 102 – Steve Zuckerberg | 555 555 552 | 2 | ● | [Expand] |
| 103 – Linus Gates | 555 555 553 | 3 | ● | [Expand] |
| 104 – Mark Ballmer | 555 555 554 | 4 | ● | [Expand] |
| 105 – Steve Cook | 555 555 555 | 5 | ● | [Expand] |
| 106 – Tim Torvalds | 555 555 556 | 6 | ● | [Expand] |
| 121 – Julia Rudd | 555 555 571 | 7 | ● | [Expand] |
| 122 – Kevin Abbott | 555 555 572 | 8 | ● | [Expand] |
| 123 – Tony Gillard | 555 555 573 | 9 | ● | [Hide] |
| SP1 Engin – AU | | | Registered | |
| SP2 Anveo – USA | | | Registered | |
| SP3 Voipfone – UK | | | Registered | |
| SP4 SPA962 | | | Registration Not Required;local_client=192.168.1 | |

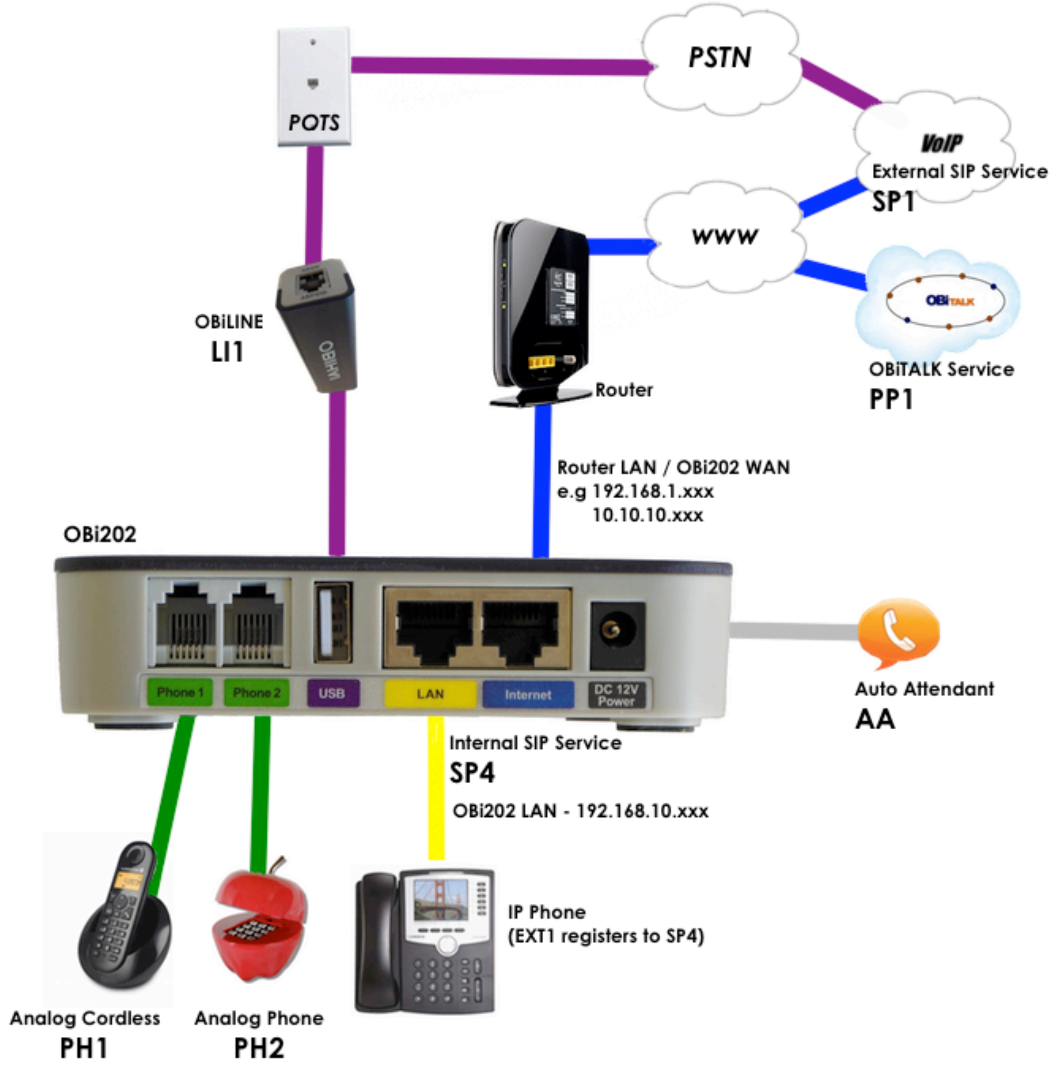
## What will be achieved at completion of this tutorial?

- You will be able to make and receive calls from the IP phone via the OBi202 using:
  - Configured SIP service(s)
  - Optional OBiLINE connected to the PSTN
  - Optional OBiBT connected via a Cellular Device
- The Caller ID will be bridged to show the original caller's incoming phone number.

## What You Need

- An OBi202 with an available "SP" service to be configured as a SIP Proxy for the IP phone.
- An IP Voice account for your SIP service correctly configured on the OBi202.
- A SIP IP-phone connected to the LAN-port (Yellow port) of the OBi202. We are using a Cisco SPA IP Phone as the example in this tutorial, but you can use any standard SIP IP Phone.
- An available EXT (or Line) on the IP-phone to be used with the OBi202.

## Option 1 - Setup the OBi as a NAT'ed IP Phone proxy (OBi202 & 302 only)



We're now going to use the OBi device as the SIP Proxy Server for the IP phone attached to the OBi's LAN-facing port. This way the IP phone registers to the OBi and can use any service configured on the OBi.

First, we need to create an ITSP profile. Just as the OBi connects to an external SIP voice service using the addressing details in the ITSP profile, we're now going to create one on the OBi as a local host.

Second we will create the credentials in the SP service just as we did for our SIP voice service.

And so, ITSP Profile D and SP4 on the OBi202 shall be configured as a proxy for the IP phone.

In the following example:

- We will use EXT 1 (or line 1) on the IP phone for this configuration, with the private/extension number 1001

- The OBi202's example OBiTALK Number is: 555 555 555 (replace this with your OBiTALK number)

First we will setup the ITSP profile for the OBi-LAN-facing SIP service that will act as the proxy for the IP phone.

Go into the service providers menu and change the settings as follows:
*(Please note: uncheck "Use OBiTALK settings" then uncheck "Device default" and enter as appropriate for each line item. In this example we are using SP4 and ITSP profile D)*

- ITSP Profile D/General/SignalingProtocol = SIP (default)
- ITSP Profile D/SIP/ProxyServer = 127.0.0.1 (ie OBi as local host is the proxy)
- ITSP Profile D/SIP/X_SpoofCallerID = yes (or checked, this will ensure we pass the caller ID through)

Next, create the local voice service that the IP phone will use as its proxy:

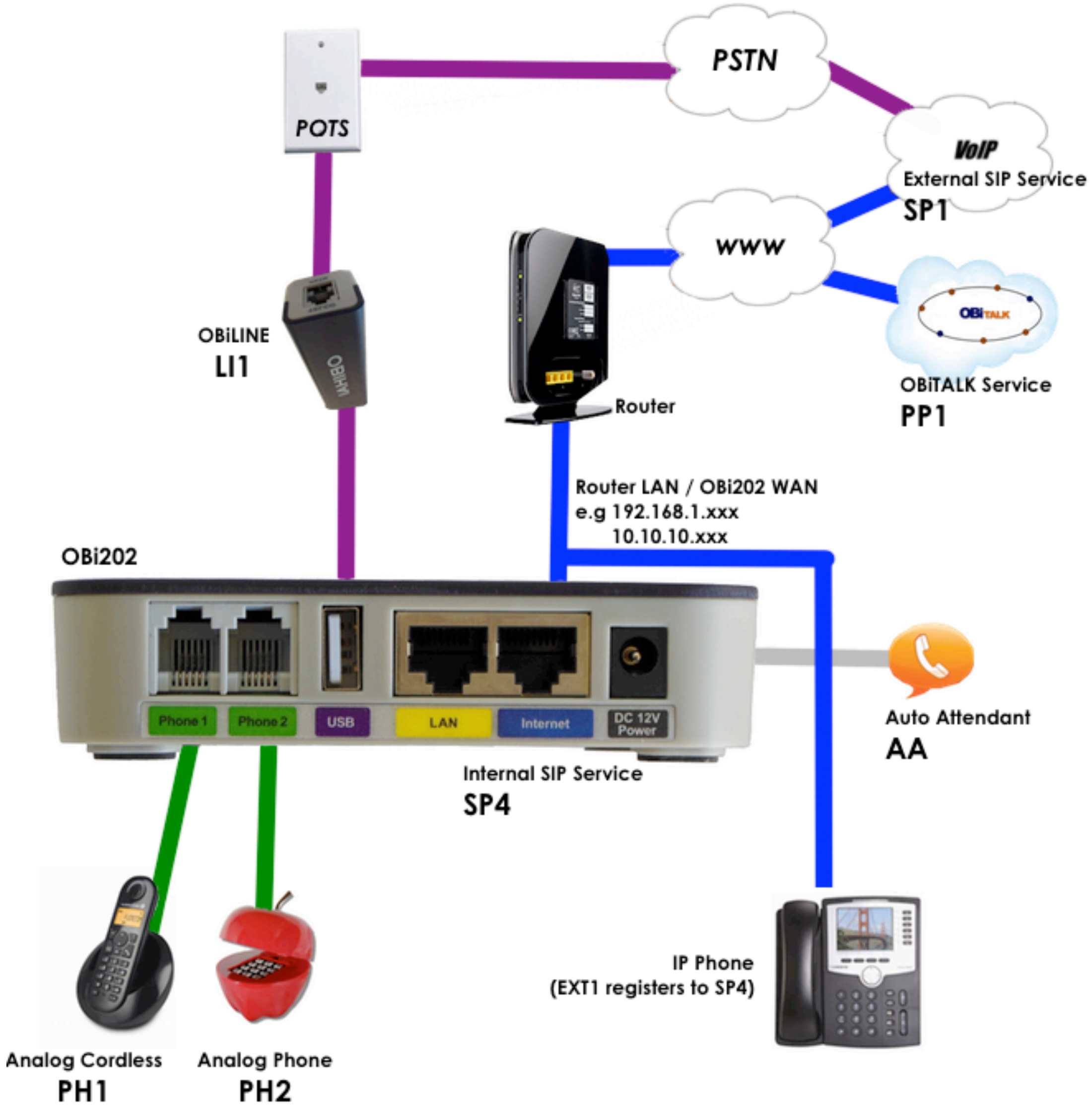
- SP4/X_ServProvProfile = D (We're saying use the ITSP D profile with this voice service)
- SP4/X_UserAgentPort = 5063 *(Note: you can choose a port that you prefer, just be sure to use the same port number throughout the tutorial*)
- SP4/X_RegisterEnable = no (or unchecked)
- SP4/AuthUserName = 1001 (This can be defined by you, but needs to be the same on the IP Phone)
- SP4/AuthPassword = AnyPhonePassword (Once again, you can define this, but must be the same on the IP Phone)
- SP4/X_Proxy = yes (or checked)

## Configure the IP phone to register to the OBi and use it as it's proxy

- EXT 1/Proxy = 555555555.pnn.obihai.com:5063
- EXT 1/User ID = 1001
- EXT 1/Password = AnyPhonePassword
- EXT 1/Register Expires = 60
- EXT 1/Register = yes
- SIP/Send Resp To Src Port = yes

(These parameters are specific to Cisco SPA IP phones, use the equivalent parameters on other brands of IP phone. Please note, your IP phone may have a separate field for the proxy port. If it does, omit the port (ie :xxxx) from the proxy address and enter the port into the port field separately.)

## Option 2 - Setup the OBi as the IP Phone proxy on a flat network (Any OBi)



We're now going to use the OBi device as the SIP Proxy Server for the IP phone attached to router's switch port so it sits on the same flat network as all other devices.

*Note: You can also re-use the OBi's yellow LAN port and keep the topology flat. To do this you need to enter the device local UI and in the router settings change the OBi to operate in a bridged config.*

As with our first option, we need to create an ITSP profile. Just as the OBi connects to an external SIP voice service using the addressing details in the ITSP profile, we're now going to create one on the OBi as a local host.

Second we will create the credentials in the SP service just as we did for our SIP voice service.

And so, ITSP Profile D and SP4 on the OBi202 shall be configured as a proxy for the IP phone.

- ITSP Profile D/General/SignalingProtocol = SIP (default)
- ITSP Profile D/SIP/ProxyServer = 192.168.1.10 (ie IP address of our SPA962 IP Phone)
- ITSP Profile D/SIP/ProxyServerPort = 5063 (SIP listening port for our IP Phone)
- ITSP Profile D/SIP/X_SpoofCallerID = yes (or checked, this will ensure we pass the caller ID through)

Next, create the local voice service that the IP phone will use as its proxy:

- SP4/X_ServProvProfile = D (We're saying use the ITSP D profile with this voice service)
- SP4/X_UserAgentPort = 5063 *(Note: you can choose a port that you prefer, just be sure to use the same port number throughout the tutorial*
- SP4/X_RingProfile = B
- SP4/X_RegisterEnable = no (or unchecked)
- SP4/AuthUserName = 1001 (This can be defined by you, but needs to be the same on the IP Phone)
- SP4/AuthPassword = AnyPhonePassword (Once again, you can define this, but must be the same on the IP Phone)
- SP4/X_Proxy = yes (or checked)

## Configure the IP phone to register to the OBi and use it as it's proxy

On the IP phone, set-up the following parameters:

- EXT 1/Proxy = 192.168.1.20:5063 (ie: IP address of OBi202)
- EXT 1/User ID = 1001
- EXT 1/Password = AnyPhonePassword
- EXT 1/Register Expires = 60
- EXT 1/Register = yes
- SIP/Send Resp To Src Port = yes

*(These parameters are specific to Cisco SPA IP phones, use the equivalent parameters on other brands of IP phone. Please note, your IP phone may have a separate field for the proxy port. If it does, omit the port (ie :xxxx) from the proxy address and enter the port into the port field separately.)*

## Call Route Syntax

For each Trunk (SP, LI, PP, and BT), configure routing rules for incoming calls in the following format:
rule OR {rule},{rule},…., where:

<table>
<tr><td colspan="7">{rule},{rule},{rule},{rule}</td></tr>
<tr><td colspan="7">rule =</td></tr>
<tr><td colspan="3">{peer-list</td><td>:</td><td colspan="3">terminal-list}</td></tr>
<tr><td colspan="3">peer-list =</td><td></td><td colspan="3"></td></tr>
<tr><td colspan="3">{(peering), (peering), (peering)</td><td>:</td><td colspan="3">(terminal), (terminal)}</td></tr>
<tr><td colspan="3">peering =</td><td></td><td colspan="3">terminal =</td></tr>
<tr><td>{(caller-list</td><td>></td><td>callee-list)</td><td>:</td><td colspan="3">PHx OR AAx OR LIx(arg) OR SPx(arg) OR PPx(arg)}</td></tr>
<tr><td>caller list =</td><td></td><td>callee-list =</td><td></td><td colspan="3">arg=</td></tr>
<tr><td>(caller|caller|caller</td><td>></td><td>callee|callee)</td><td></td><td>cid</td><td>></td><td>target</td></tr>
<tr><td>caller =</td><td></td><td>callee =</td><td></td><td>cid =</td><td></td><td>target =</td></tr>
<tr><td>number OR digit-map OR ? OR @<br><i>(?=anonymous, @=any #)</i></td><td></td><td>number OR digit-map OR @</td><td></td><td>digit-map or spoofed-caller-number<br><i>(the outbound CallerID)</i></td><td>></td><td>digit-map or number-to-call</td></tr>
</table>

- Rules and peerings are tested from left to right
- First matched rule will be selected and used to route the call
- For OutboundCallRoute, every rule must be placed inside a pair of { } curly brackets
- If there is only 1 rule, then { } is optional.
- {0255556666:SP2} is the same as {(0255556666):SP2} - since 0255556666 is just a constant, the parentheses are optional - however:
- If the rule includes things like xxxx and so forth that is part of digitmap syntax, the round brackets are required, like this: {(025xx.):SP2}

First we will look at the call flow and then we will create the syntax we are going to use to manage call routing from the IP Phone.

## Call flow from the IP Phone to the OBi

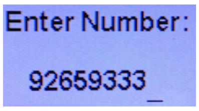
As our IP Phone in this example is connecting to SP4, SP4 will see what is sent from the IP Phone as an Inbound Call and apply it's inbound call route.

To visualise the call flow, we can imagine:

- The Inbound Call Route for SP4 to be the Outbound Call Route for the legacy IP phone.
- The digit map on the phone only serves to match or block numbers dialled on the handset.
- SP4's Inbound Call Route will determine which terminal within the OBi to pass the incoming digits to after evaluating the digits against the digit maps included in the SP4 Inbound CallRoute and applying any digit transformation if required.
- After determining the digits as still being valid, they are then sent directly to line via the specified SP being SP1 in this scenario.

Let's look at how an 8-digit local call dialled in Sydney would flow from the IP Phone to SP4 where it will have 02 added before being sent to the VSP/ITSP:

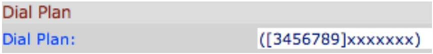
User Dials number on IP phone, for example:
9265 9333 (City of Sydney)



Enter Number:
92659333_

---

IP Phone digit map (or dial plan) will match the number or if the phone doesn't have a digit map, the user simply hits dial



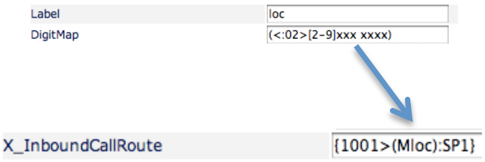
Dial Plan
Dial Plan: ([3456789]xxxxxxx)

*or*

---

The OBi receives the call on SP4 and reads the number received and applies it's inbound call route: {1001>(Mloc):SP1}, which means take the caller 1001 and match the callee with Mloc then send directly to the VSP/ITSP via SP1.

In this case it's matching 92659333 with the user defined digit map, (Mloc). (Mloc) is going to add 02 to the number and so 02 9265 9333 is sent directly to the VSP/ITSP via SP1.



Label loc
DigitMap (<:02>[2-9]xxx xxxx)

X_InboundCallRoute {1001>(Mloc):SP1}

---

## Creating Our Call Routes

Just as in our regionalisation tutorial, we want the OBi to direct calls based on the number dialled to the correct outbound terminal. Additionally, we're going to add another feature to our IP Phone - the ability to choose which of the OBi's outbound terminals it uses by mapping the extra line keys on the IP phone.

In the voice service menu, go into SP1 (which is our SIP voice service) and we'll tell it to send all the calls it receives to the two phone ports as well as to SP4, which is where our IP Phone is proxied to: (By default this reads ph,ph2 which means send the call to phone port 1 and 2)

So, if you used option 1 to setup your network, set the call route as follows:

- SP1/X_InboundCallRoute = {SP4(1001@local_client),ph,ph2}

And, if you used option 2 to setup your network, set the call route as follows:

- SP1/X_InboundCallRoute = {ph,ph2,sp4(1001)}

Next we need to tell the OBi how to deal with calls the IP Phone makes. So, remember, always visualise from the OBi's perspective. In this case, the outbound call route of the IP Phone is seen as an inbound call route from the perspective of the OBi as the OBi is receiving an inbound request from the phone that needs to be routed. So, when the IP phone sends a number to the OBi, we want to send it somewhere. We are going to have the OBi direct all calls from the IP Phone using the same logic as the localisation tuorial and the user defined digit maps we created in that tutorial (or by uploading the Australian config file). We will send all emergency, free, local rate, special and premium numbers out the (optional) OBiLINE and everything else via SP1 as follows:

- SP4/X_InboundCallRoute = {1001>((Msos)|(Mfre)|(Mlrn)|(Mspe)|(Mpre)):LI},{1001>((Mloc)|(Mint)|(Mnat)|(Mipd)):SP1}

Of course, if you wish to just direct everything out SP1 (if you don't have an OBiLINE for example) then just set this to {1001>(Msp1):SP1}

So let's consider our configuration end goal for using the line keys on our IP Phone. Our IP Phone in this tutorial is the Cisco SPA962 that has 6 line (EXT) keys.

Let's say we have configured 3 VSPs on our OBi and we want to access them individually as virtual outbound lines. We use the same credentials we used for EXT 1 on all our lines and we then have the SPA insert a line code using it's dial plan and the OBi will use this to direct the call to the correct terminal.

The advantage of this approach is that we only occupy a single SP1 slot on our OBi. However, all inbound calls will appear on EXT1. Our final configuration will see our line keys mapped as follows:

EXT 1: Uses the call route above to direct calls.

EXT 2: SPA inserts *1 in front of the dialled digits via it's dial plan `(<:1*>xx.)` - The OBi will direct this to SP1 using the inbound call route `{1001>(<1*:>(Msp1)):SP1}`

EXT 3: SPA inserts *2 in front of the dialled digits via it's dial plan `(<:2*>xx.)` - The OBi will direct this to SP2 using the inbound call route `{1001>(<2*:>(Msp2)):SP2}`

EXT 4: SPA inserts *3 in front of the dialled digits via it's dial plan `(<:3*>xx.)` - The OBi will direct this to SP3 using the inbound call route `{1001>(<3*:>(Msp3)):SP3}`

EXT 5: SPA inserts *4 in front of the dialled digits via it's dial plan `(<:4*>xx.)` - The OBi will direct this to the PSTN using the inbound call route `{1001>(<4*:>(Mli)):LI}`

EXT 6: SPA inserts *5 in front of the dialled digits via it's dial plan `(<:5*>xx.)` - The OBi will direct this to the OBiTALK service using the inbound call route `{1001>(<5*:>(Mpp)):PP}`

Our line keys will map across to the OBi as follows:

| KEY LABEL | SPA LINE | SPA Dial Plan | OBi SP4 IBCR |
|---|---|---|---|
| Make a call | EXT 1 | (xx.) | (as above) |
| SP1 | EXT 2 | (<:1*>xx.) | {1001>(<1*:>(Msp1)):SP1} |
| SP2 | EXT 3 | (<:2*>xx.) | {1001>(<2*:>(Msp2)):SP2} |
| SP3 | EXT 4 | (<:3*>xx.) | {1001>(<3*:>(Msp3)):SP3} |
| PSTN | EXT 5 | (<:4*>xx.) | {1001>(<4*:>(Mli)):LI} |
| OBiTALK | EXT 6 | (<:5*>xx.) | {1001>(<5*:>(Mpp)):PP} |

If you are using a different accessory, such as the OBiBT, you could program an extension to support that accessory, so for the OBiBT you would use digit map (Mbt1) and send it to terminal BT1.

Obi SP4 IBCR syntax:

{1001>((Msos)|(Mfre)|(Mlrn)|(Mspe)|(Mpre)):LI},{1001>((Mloc)|(Mint)|(Mnat)|(Mipd)):SP1},{1001>(<1*:>(Msp1)):SP1},{1001>(<2*:>(Msp2)):SP2},{1001>(<3*:>(Msp2)):SP2},{1001>(<4*:>(Msp3)):SP3},{1001>(<5*:>(Mpp)):PP}

If you are using a different accessory, such as the OBiBT, you could program an extension to support that accessory, so for the OBiBT you would use digit map (Mbt1) and send it to terminal BT1.

For example, let’s assume you are not using the OBiLINE adapter for PSTN calls but, instead, want to utilise the OBiBT.

We can repurpose Ext 5 as:

| KEY LABEL | SPA LINE | SPA Dial Plan | OBi SP4 IBCR |
|---|---|---|---|
| BT1 | EXT 5 | (<:4*>xx.) | {1001>(<4*:>(Mbt1)):BT1} |

We then edit the {1001>(<4*:>(Mli)):LI} rule as {1001>(<4*:>(Mbt1)):BT1} – so now the SP4 IBCR is as follows:

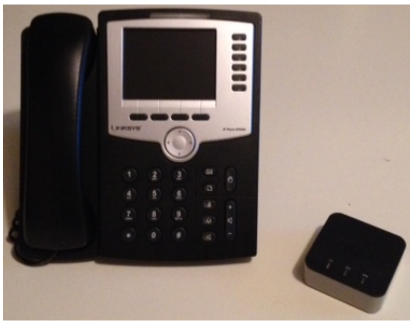
{1001>((Msos)|(Mfre)|(Mlrn)|(Mspe)|(Mpre)):LI},{1001>((Mloc)|(Mint)|(Mnat)|(Mipd)):SP1},{1001>(<1*:>(Msp1)):SP1},{1001>(<2*:>(Msp2)):SP2},{1001>(<3*:>(Msp3)):SP3},{1001>(<4*:>(Mbt1)):BT1},{1001>(<5*:>(Mpp)):PP}

You should now have a setup whereby you can pickup the handset and have the OBi decide where to direct the call or use a line key to direct the outbound call.

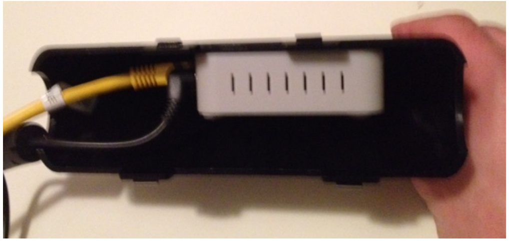
## Party trick: Hide an OBi200 inside a SPA handset

*(Note: Only works with OBi200 and OBi300, the OBi202 and OBi302 are too big)*

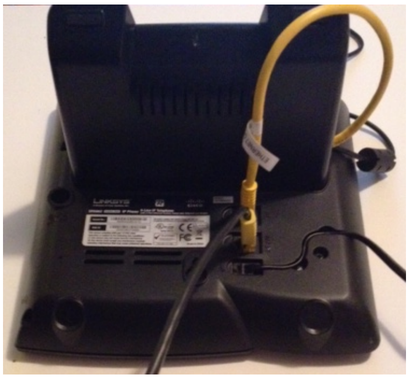
Take a SPA- IP phone and an OBi200:



Use the accessory base of the SPA- phone to house the 200 and cables:



Plugging the OBi into the SPA pass-through port:



Creating your OBi-Powered SPA. Odd the optional OBiBT Bluetooth dongle to have your mobile phone automatically attach to your desk phone when you are in range.