OBI Digitmaps – Unravelling the Mysteries

v1.1 – 2014-03-30

You'll wonder how you ever lived without 'em!

Contents

PART 1

Introduction

When it comes to ATAs, most of us cut our teeth on the original Sipura SPA-2000/SPA-3000/SPA-3102 devices nearing on ten years now. It didn't take long before we finally managed to get our heads around the Sipura dial plan syntax, becoming competent at crafting them for our own individual needs and requirements and helping out the less fortunate VoIP-challenged users.

Since the Sipura devices first appeared on the VoIP scene, Internet Telephony slowly became accessible for the average person with a myriad of ATAs and VoIP enabled ADSL modems/routers appearing on the market from various companies such as Billion, Netgear, Netcomm, TP-Link, etc, just to name a few. Because the VoIP features of such devices are rather limited, configuring dial plans on such devices is generally a basic and fairly simple exercise once the syntax is understood.

By contrast, OBIHAI have incorporated extra features & functionality into their devices – for example:

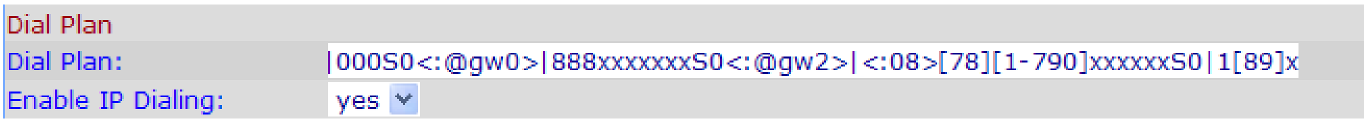
- The ability to bridge a call between two interfaces.
- The ability to configure Automatic Call Back.
- Provision of a basic Automated Attendant.
- The ability to utilise Bluetooth devices.

In order to accommodate the extra features, the Obi device has taken the functionality of what most dial plans do – i.e. to filter and route calls to the desired interface as a single function, and divided it into two individual functions – filtering and routing. Such an approach allows the user more control over implementing and configuring the additional features and keeps things more manageable. Trying to stick with the 'keep it all in one line' Sipura SPA syntax would have probably been doable in principle, but completely impractical.

Now that we have an Obi202 at our disposal it seems like we have gone back ten years when it comes to deciphering and crafting dial plans within the Obi. No matter how many times you read the digitmap section in the Admin guide, some concepts make sense yet other concepts leave you confused and lost.

The aim of this document is to share some of the things I have done, tried and tested, in the pursuit of nailing digitmaps (DMs) and Outbound CallRoutes (OBCRs). I experienced my 'eureka' moment once I saw Mark W present his format of crafting DM's & OBCRs. Hopefully by the end of this document, you too, will have a better, deeper understanding of the concepts and benefits and wonder '*how you ever lived without DMs*.'

I view the Obi202 as a SPA-3102 on 'steroids' that has been enhanced with extra features and capabilities. With the dial plan for a SPA-3102, all the various dialling sequences are entered into the one field on the 'Line 1' tab which can also include GW(n) parameters to instruct the 3102 via which gateway the call should be routed:



Dial Plan
Dial Plan: |000S0<:@gw0>|888xxxxxxxS0<:@gw2>|<:08>[78][1-790]xxxxxxS0|1[89]x
Enable IP Dialing: yes

As most of you are aware, as time went on, the various VSP's changed their backend systems forcing users to 'register' their accounts which made the use of gateways redundant with those VSPs. Even if the gateways were usable, you couldn't receive and answer calls on those accounts – you could only make outgoing calls.

The Obi202 allows the registration of 4 VSP accounts which allows you to make & receive calls on all 4 services as well as bridge calls between accounts or services.

Whereas the dial plan syntax is entered into one field on the SPA-3102 to analyse the dialled digits and route the calls out via a VSP, the Obi has split that function into 2 individual steps or processes – the DigitMap (DM) & the Outbound CallRoute (OBCR).

In a nutshell, a digitmap contains a 'dial plan' sequence – e.g. (04xxxxxxxx|1[38]00xxxxxx). As part of setting up a call, one or more DMs are used to either allow or deny (think security guard) a set of digits dialled by the user and the OBCR also uses one or more DMs to route (think traffic cop) a valid number out via a VSP.

The Security Guard and Traffic Cop

I'll attempt to use an analogy to explain the basic concepts of the DM & OBCR.

Let's assume there is a courier company that has a large fleet of vehicles (aka phone numbers) in a fairly large car park and the exit ramp fronts a large intersection with roads heading off to the North, South, East and West (aka VSPs 1-4).

At the exit ramp is a security guard that has a list (aka DM) of vehicle types allowed to exit:

- Sedans (local numbers).
- 4WDs (national numbers).
- Vans (Mobile numbers).
- Wagons (1300/1800 numbers).

In the middle of the intersection is a traffic cop who also has a copy of the DM list but with routing instructions added:

- Sedans (local numbers) - To be directed North (VSP 1)
- 4WDs (national numbers) - To be directed South (VSP 2)
- Vans (Mobile numbers) - To be directed East (VSP 3)
- Wagons (1300/1800 numbers) - To be directed West (VSP 4)

The dispatcher organises a driver to make a delivery up north (i.e. make a local call). The driver hops into the sedan and proceeds to exit the car park. The security guard checks his list and confirms that sedans are allowed to exit. The driver exits the car park and is then processed by the traffic cop who checks against his list and confirms that sedans are allowed but can only be directed north. The driver proceeds in a northerly direction (i.e. local number is sent to VSP 1.

Same principles apply for the other vehicles (call types).

Should a semi-trailer (international number) attempt to leave the car park, exit would be denied since the security guard's list does not contain an entry for semi-trailers.
This is a very simple analogy to demonstrate and visualise a concept and by no means covers all possible scenarios.

Configuring An Obi The SPA Way

Now it’s time to pop the bonnet of the Obi202 and dig into the Digitmaps and Outbound CallRoutes. I’ll set up a scenario concentrating mainly on configuring dial plans for use by PH1 and 4 VSPs on SP1-4 as well as PSTN on LI. I won’t go into the details of setting up the VSPs since that info is already covered in the Admin guide.

All my configurations are preformed locally using the Obi’s Web UI after having reset the Obi to factory default. I haven’t bothered with tweaking any regional or local settings, etc, since they don’t have any adverse effects on dialling and making test calls using my test configurations. For the time being, I’ve avoided using the ObiTalk portal to eliminate the potential for errors considering there are discrepancies between the menu items displayed via the portal compared to what is displayed via the Web UI. Once I become competent in navigating my way through all the different menus & settings locally, then I’ll look at using the ObiTalk portal for tweaking.

As I mentioned earlier, the Obi is a SPA-3102 on steroids and when you look at the underlying Obi dial plan syntaxes, they are the same as used in the SPA ATAs. So, the Obi dial plan engine is essentially the same as in earlier SPA models. There are a few slight variations in how the dial plan ‘rules’ are formatted depending whether it’s used in a DM field – e.g. ([78]xxxxxxx|04xxxxxxxx) is enclosed in brackets whereas in an OBCR field – e.g. {([78]xxxxxxx|04xxxxxxxx):SP1} the rules including the brackets are enclosed in curly braces which also includes the outbound interface – SP1.

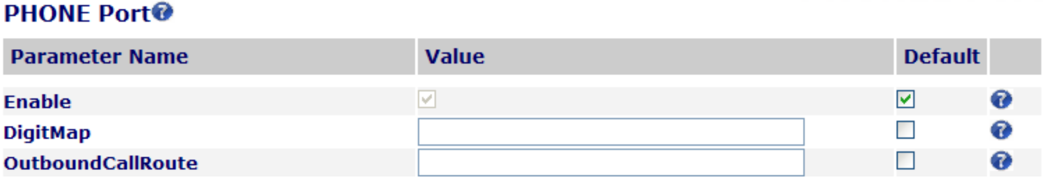
Read the section on DMs & OBCRs in the Admin guide for further info.

To set the scenario for my config and testing:

- The Obi202 has been reset to factory default & then the VSPs configured. (Don’t forget to re-point your SP services to the correct ITSP profile, otherwise only SP1 will register and the remainder won’t).
- I’ve deleted the default DM & OBCR entries for PH1 leaving them empty.
- I’ve deleted the default DM entries for ISTP A/B/C/D (SP1-4) and LI to prevent any of those default settings adversely affecting my config & testing.
- Local/National calls to use MNF configured on SP1. (My local calls are SA based so amend yours for your state accordingly).
- Mobile calls to use PT (PennyTel) configured on SP2.
- 13/1300/1800 calls to use TC (Telecube) configured on SP3.
- International calls to use MaxoTel configured on SP4.
- 000 & 11xx calls to use the PSTN on LI.
- The dial plans that I’ll configure are by no means comprehensive and are used solely to demonstrate the principles in crafting dial plans to make outgoing calls via the various VSPs and PSTN. Therefore, do not rely on what I present as being a configuration that will cover every call scenario.

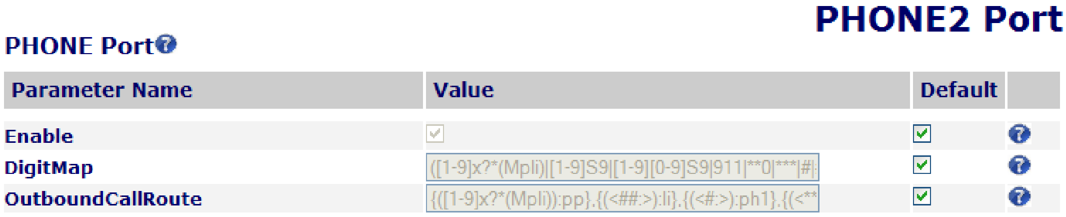
Note: As the PH1 DM & OBCR have been deleted, there is no longer any access to the IVR or other dialling features normally available with default settings. In case IVR access is required during config & testing, PH2 can be used.

PHONE1 Port



PHONE Port

Parameter Name	Value	Default
Enable	☑	☑
DigitMap		☐
OutboundCallRoute		☐



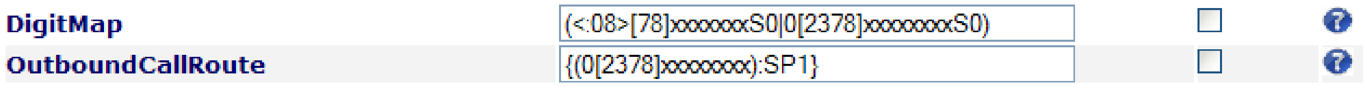
PHONE2 Port

PHONE Port

Parameter Name	Value	Default	
Enable	☑	☑	?
DigitMap	([1-9]x?*(Mpli)|[1-9]S9|[1-9][0-9]S9|911|**0|***|#|	☑	?
OutboundCallRoute	{([1-9]x?*(Mpli)):pp},{(<##:>):li},{(<#:>):ph1},{(<**	☑	?

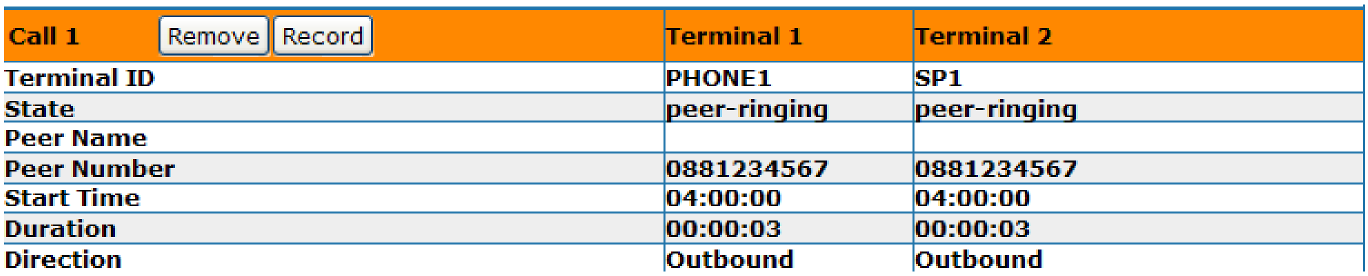
The SPA Configuration

To start off, I'm going to configure the PH1 DM & OBCR to allow dialling of local 8 digit Adelaide numbers prefixed with 08 as well as 10 digit numbers for national calls and route them out via SP1.

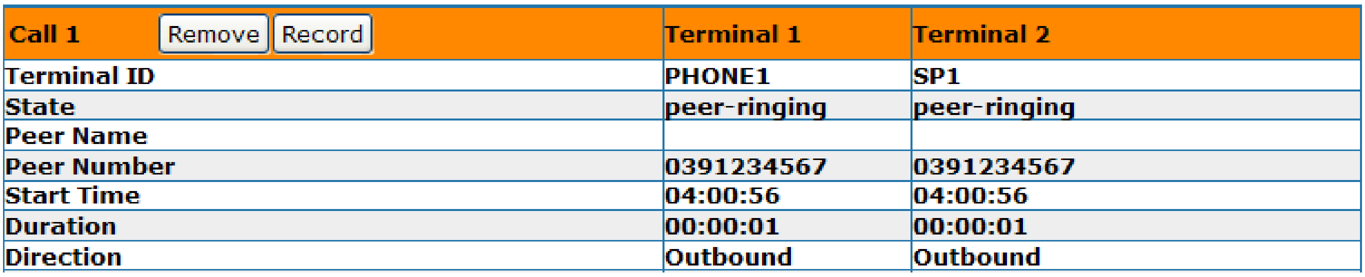


DigitMap	(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0)	☐	?
OutboundCallRoute	{(0[2378]xxxxxxxx):SP1}	☐	?

Test calls to local 8123 4567 and 03 9123 4567 result in successful calls via MNF on SP1.



Call 1 Remove Record	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP1
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	0881234567	0881234567
Start Time	04:00:00	04:00:00
Duration	00:00:03	00:00:03
Direction	Outbound	Outbound



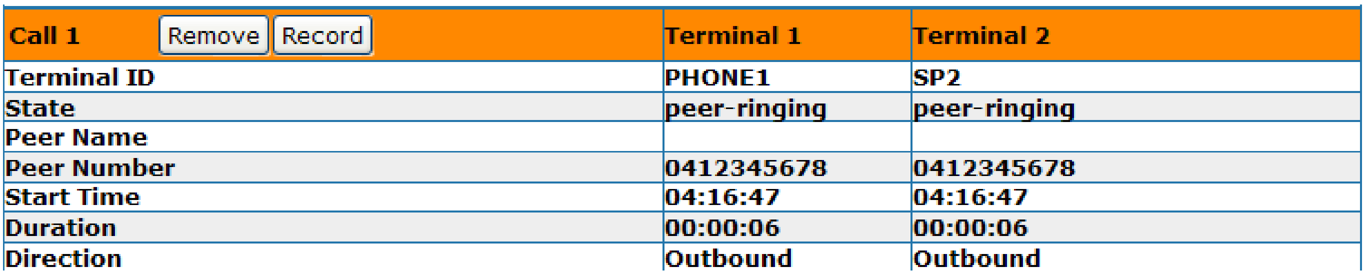
Call 1 Remove Record	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP1
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	0391234567	0391234567
Start Time	04:00:56	04:00:56
Duration	00:00:01	00:00:01
Direction	Outbound	Outbound

Next, I'll add the capability to dial mobile numbers via PT on SP2.

DM – (<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0|04xxxxxxxxS0)

OBCR – {(0[2378]xxxxxxxx):SP1},{(04xxxxxxxx):SP2}

Test call to 0412345678 results in a successful call via PT on SP2.



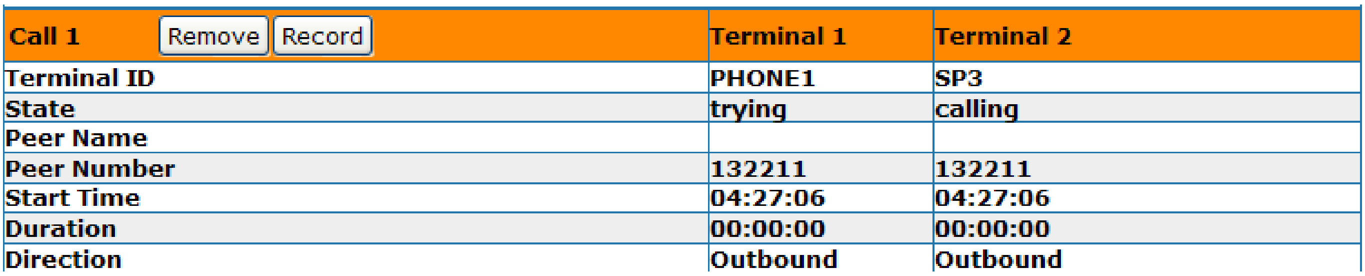
Call 1 Remove Record	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP2
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	0412345678	0412345678
Start Time	04:16:47	04:16:47
Duration	00:00:06	00:00:06
Direction	Outbound	Outbound

Next, I'll add the capability to dial 13/1300/1800 numbers via TC on SP3.

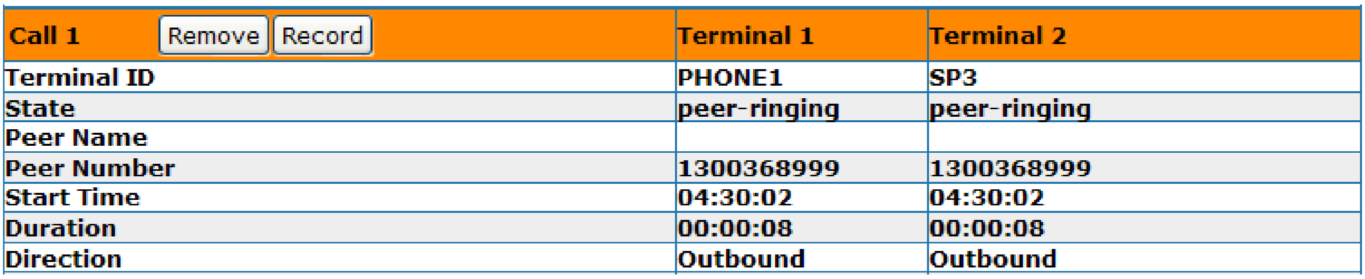
DM – (<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0|04xxxxxxxxS0|13[1-9]xxxS0|1[38]xxxxxxxxS0)

OBCR – {(0[2378]xxxxxxxx):SP1},{(04xxxxxxxx):SP2},{(13[1-9]xxx|1[38]xxxxxxxx):SP3}

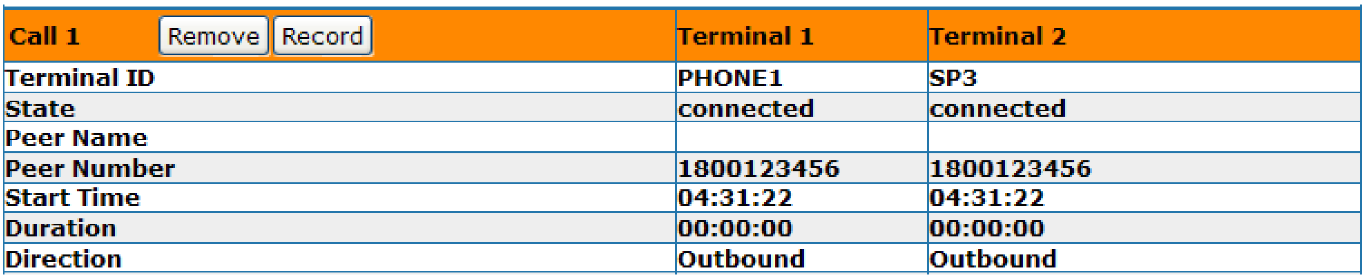
Test calls to 132211, 1300368999 & 1800123456 result in successful calls via TC on SP3.



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP3
State	trying	calling
Peer Name		
Peer Number	132211	132211
Start Time	04:27:06	04:27:06
Duration	00:00:00	00:00:00
Direction	Outbound	Outbound



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP3
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	1300368999	1300368999
Start Time	04:30:02	04:30:02
Duration	00:00:08	00:00:08
Direction	Outbound	Outbound



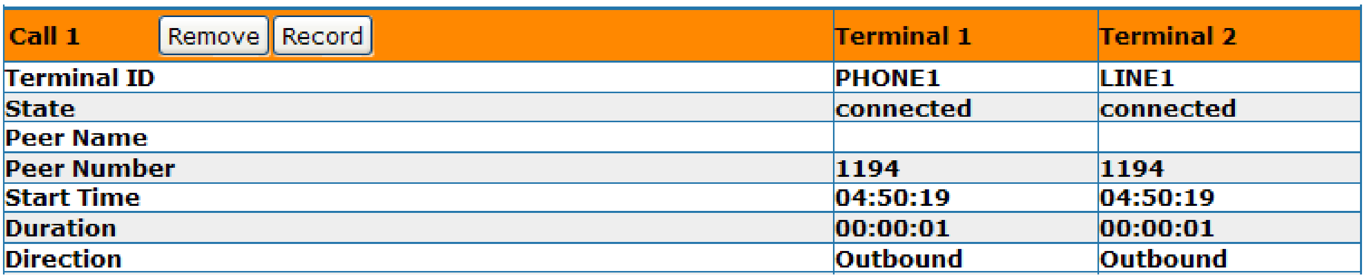
Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP3
State	connected	connected
Peer Name		
Peer Number	1800123456	1800123456
Start Time	04:31:22	04:31:22
Duration	00:00:00	00:00:00
Direction	Outbound	Outbound

And finally, I'll add international dialling via MaxoTel on SP4 and 000/11xx via the PSTN on LI.

DM – (<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0|04xxxxxxxxS0|13[1-9]xxxS0|1[38]xxxxxxxxS0|0011xx.|000S0 |11xxS0)

OBCR – {(0[2378]xxxxxxxx):SP1},{(04xxxxxxxx):SP2},{(13[1-9]xxx|1[38]xxxxxxxx):SP3},{(0011xx.):SP4}, {(000|11xx):LI}

Although I didn't have an international number to test and wasn't game to ring 000, a call to 1194 via the PSTN was successful.



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	LINE1
State	connected	connected
Peer Name		
Peer Number	1194	1194
Start Time	04:50:19	04:50:19
Duration	00:00:01	00:00:01
Direction	Outbound	Outbound

We now have a functional Obi that can make outgoing calls via its various interfaces having used SPA type dial syntaxes – or, dial plans if you prefer.

Forcing a Call Using Access Codes

With our current config, dialling a mobile, for example, will send the call out via PT on SP2 automatically. What if we wanted to force a mobile call out via the PSTN on LI or an international call via MNF on SP1?

We can achieve that functionality by configuring access codes to force a call using a particular outgoing interface. For example, if we use **6 as the access code for accessing LI, we would dial **60412345678, which the Obi would then strip **6 and send the remaining digits (0412345678) out via LI.

So let's configure **1 - **4 as access codes for forcing numbers out via SP1 - 4 and **6 for LI. We'll amend the DM & OBCR so that any number can be forced out any interface. I initially used **5 but test calls failed. I then realised that **5 is used during the set up process when adding an Obi device to the ObiTalk portal. Hence I'm now using **6.

Our config currently looks like this:

DM

```
(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0|04xxxxxxxxS0|13[1-9]xxxS0|1[38]xxxxxxxxS0|0011xx.|000S0 |11xxS0)
```

OBCR

```
 {(0[2378]xxxxxxxx):SP1},{(04xxxxxxxx):SP2},{(13[1-9]xxx|1[38]xxxxxxxx):SP3},{(0011xx.):SP4}, {(000|11xx):LI}
```

Now let's add the access codes and appropriate rules:

DM

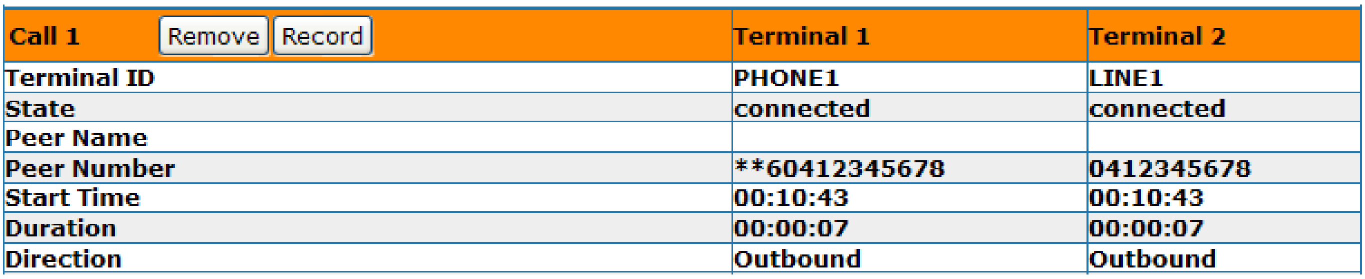
```
(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0|04xxxxxxxxS0|13[1-9]xxxS0|1[38]xxxxxxxxS0|0011xx.|000S0|11xxS0|
**1[78]xxxxxxxS0|**10[2378]xxxxxxxxS0|**104xxxxxxxxS0|**113[1-9]xxxS0|**11[38]xxxxxxxxS0|**10011xx.|
**1000S0|**111xxS0|**2[78]xxxxxxxS0|**20[2378]xxxxxxxxS0|**204xxxxxxxxS0|**213[1-9]xxxS0|
**21[38]xxxxxxxxS0|**20011xx.|**2000S0|**211xxS0|**3[78]xxxxxxxS0|**30[2378]xxxxxxxxS0|
**304xxxxxxxxS0|**313[1-9]xxxS0|**31[38]xxxxxxxxS0|**30011xx.|**3000S0|**311xxS0|**4[78]xxxxxxxS0|
**40[2378]xxxxxxxxS0|**404xxxxxxxxS0|**413[1-9]xxxS0|**41[38]xxxxxxxxS0|**40011xx.|**4000S0|
**411xxS0|**6[78]xxxxxxxS0|**60[2378]xxxxxxxxS0|**604xxxxxxxxS0|**613[1-9]xxxS0|**61[38]xxxxxxxxS0|
**60011xx.|**6000S0|**611xxS0)
```

OBCR

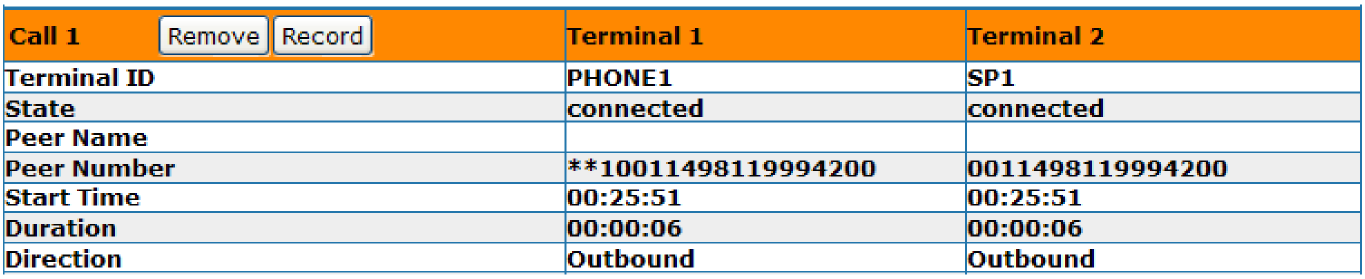
```
{(0[2378]xxxxxxxx):SP1},{(04xxxxxxxx):SP2},{(13[1-9]xxx|1[38]xxxxxxxx):SP3},{(0011xx.):SP4},{(000|11xx):LI},
{(<**1:08>[78]xxxxxxxS0|<**1:>0[2378]xxxxxxxxS0|<**1:>04xxxxxxxxS0|<**1:>13[1-9]xxxS0|
<**1:>1[38]xxxxxxxxS0|<**1:>0011xx.|<**1:>000S0|<**1:>11xxS0):SP1},{(<**2:08>[78]xxxxxxxS0|
<**2:>0[2378]xxxxxxxxS0|<**2:>04xxxxxxxxS0|<**2:>13[1-9]xxxS0|<**2:>1[38]xxxxxxxxS0|<**2:>0011xx.|
<**2:>000S0|<**2:>11xxS0):SP2},{(<**3:08>[78]xxxxxxxS0|<**3:>0[2378]xxxxxxxxS0|<**3:>04xxxxxxxxS0|
<**3:>13[1-9]xxxS0|<**3:>1[38]xxxxxxxxS0|<**3:>0011xx.|<**3:>000S0|<**3:>11xxS0):SP3},
{(<**4:08>[78]xxxxxxxS0|<**4:>0[2378]xxxxxxxxS0|<**4:>04xxxxxxxxS0|<**4:>13[1-9]xxxS0|
<**4:>1[38]xxxxxxxxS0|<**4:>0011xx.|<**4:>000S0|<**4:>11xxS0):SP4},{(<**6:08>[78]xxxxxxxS0|
<**6:>0[2378]xxxxxxxxS0|<**6:>04xxxxxxxxS0|<**6:>13[1-9]xxxS0|<**6:>1[38]xxxxxxxxS0|<**6:>0011xx.|
<**6:>000S0|<**6:>11xxS0):LI}
```

Keep in mind that the DM rules are all on one line as are all the rules for OBCR.

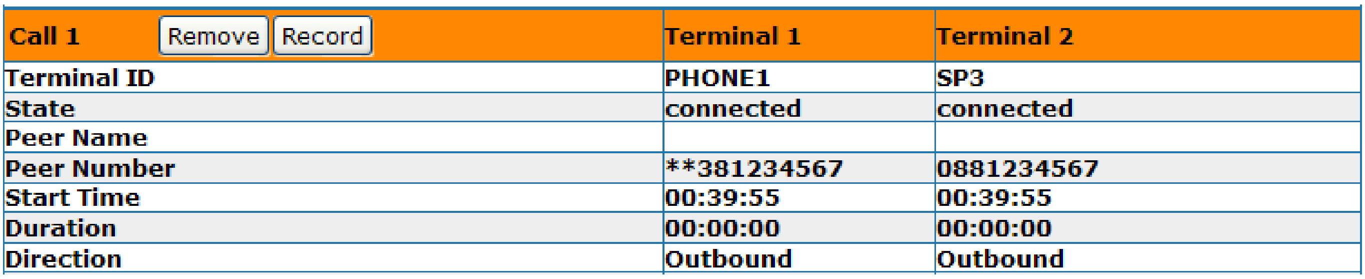
And to prove the above changes worked, a test call to a mobile was forced via PSTN, an international fax call forced via SP1 (MNF) and a local call forced via SP3 (TC):



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	LINE1
State	connected	connected
Peer Name		
Peer Number	**60412345678	0412345678
Start Time	00:10:43	00:10:43
Duration	00:00:07	00:00:07
Direction	Outbound	Outbound



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP1
State	connected	connected
Peer Name		
Peer Number	**10011498119994200	0011498119994200
Start Time	00:25:51	00:25:51
Duration	00:00:06	00:00:06
Direction	Outbound	Outbound



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP3
State	connected	connected
Peer Name		
Peer Number	**381234567	0881234567
Start Time	00:39:55	00:39:55
Duration	00:00:00	00:00:00
Direction	Outbound	Outbound

By this stage you can appreciate the large size of the dial plan entry both for the DM & OBCR and this is only for PH1. Duplicate that for PH2 and you can imagine trying to find an error where you may have typed a semi-colon ';' instead of a colon ':' for instance or omitted an opening '{' or closing '}' curly brace.

Also consider the limited amount of data that is actually visible in the DM parameter window – there is quite a bit of scrolling backwards & forwards involved if looking for a particular rule in the dial plan. Imagine if a Telco carrier opened up a new number range, e.g. 6xxx xxxx for SA and 5xxx xxxx for NSW. You would need to go through the DMs & OBCRs for PH1 & PH2 and meticulously manipulate the dial plan rules in multiple locations.
Up to this point, our dial plan configuration has been based on the SPA-2000/3102 syntax. In the next section I'll attempt to explain & demonstrate the flexibility of the Obi in configuring our dial plans by leveraging the features of the digitmap, including the use of 'labelled' digitmaps, to reduce our dial plans into manageable 'bite size' pieces.

PART 2

The DigitMap

When it comes to the Obi202, what is a DigitMap? I view a digitmap as being a repository, a bin, a bucket, a container – call it what you like – that is a placeholder for storing a dial plan sequence, i.e. a rule or multiple rules, akin to a spreadsheet cell containing data, and every digitmap in the Obi is essentially a ‘row’ in a small database or table which can be represented, with default parameters, as follows:

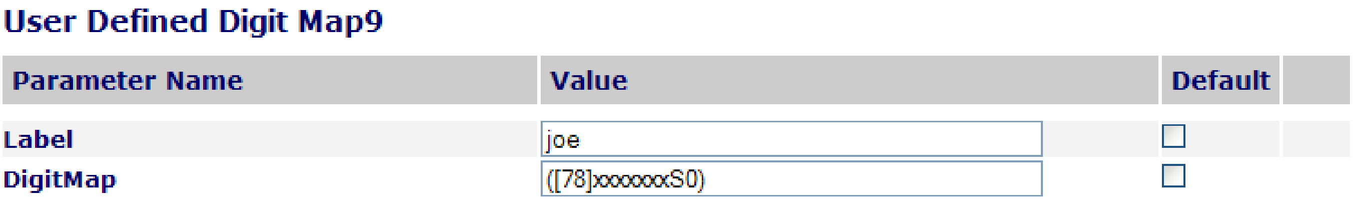
Interface	Ref As	Dial Plan
ITSP Profile A	Msp1	(1xxxxxxxxxx|<1>[2-9]xxxxxxxxx|011xx.|xx.|(Mipd)|[^*#]@@.)
ITSP Profile B	Msp2	(1xxxxxxxxxx|<1>[2-9]xxxxxxxxx|011xx.|xx.|(Mipd)|[^*#]@@.)
ITSP Profile C	Msp3	(1xxxxxxxxxx|<1>[2-9]xxxxxxxxx|011xx.|xx.|(Mipd)|[^*#]@@.)
ITSP Profile D	Msp4	(1xxxxxxxxxx|<1>[2-9]xxxxxxxxx|011xx.|xx.|(Mipd)|[^*#]@@.)
OBiTALK Service	Mpp	(<ob>xxxxxxxxx|obxxxxxxxxx)
Auto Attendant	Maa1	([1-9]x?*(Mpli)|[1-9]|[1-9][0-9]|<00:$1>|0|**1(Msp1)|**2(Msp2)|**3(Msp3)|**4(Msp4)|**70(Mli)|**8(Mbt)|**81(Mbt)|**82(Mbt2)|**9(Mpp)|(Mpli))
Voice Gateway1	Mvg1	(xx.)
Voice Gateway2	Mvg2	(xx.)
Voice Gateway3	Mvg3	(xx.)
Voice Gateway4	Mvg4	(xx.)
Voice Gateway5	Mvg5	(xx.)
Voice Gateway6	Mvg6	(xx.)
Voice Gateway7	Mvg7	(xx.)
Voice Gateway8	Mvg8	(xx.)
Trunk Group1	Mtg1	(Msp1)
Trunk Group2	Mtg2	(Msp1)
Trunk Group3	Mtg3	(xx.)
Trunk Group4	Mtg4	(xx.)
OBiBlueTooth 1	Mbt1	(1xxxxxxxxxx|011xx.|[1-9]11S2|[1-9]xx)
OBiBlueTooth 2	Mbt2	(1xxxxxxxxxx|011xx.|[1-9]11S2|[1-9]xx)
Phone Port 1	Mph1	([1-9]x?*(Mpli)|[1-9]S9|[1-9][0-9]S9|911|**0|***|#|##|**70(Mli)|**8(Mbt)|**81(Mbt)|**82(Mbt2)|**1(Msp1)|**2(Msp2)|**3(Msp3)|**4(Msp4)|**9(Mpp)|(Mpli))
Phone Port 2	Mph2	([1-9]x?*(Mpli)|[1-9]S9|[1-9][0-9]S9|911|**0|***|#|##|**70(Mli)|**8(Mbt)|**81(Mbt)|**82(Mbt2)|**1(Msp1)|**2(Msp2)|**3(Msp3)|**4(Msp4)|**9(Mpp)|(Mpli))
Line Port	Mli	(xxxxxxxS4|1xxxxxxxxxx|xx.)
User Defined DM 1	Mipd	(xx.<*:@>xx?x?<*:.>xx?x?<*:.>xx?x?<*:.>xx?x?|xx.<*:@>xx?x?<*:.>xx?x?<*:.>xx?x?<*:.>xx?x?<*::>xx?x?x?x?)
User Defined DM 2	Mlabel	
User Defined DM 3	Mlabel	
User Defined DM 4	Mlabel	
User Defined DM 5	Mlabel	
User Defined DM 6	Mlabel	
User Defined DM 7	Mlabel	
User Defined DM 8	Mlabel	
User Defined DM 9	Mlabel	
User Defined DM 10	Mlabel	

The Joe Bloggs DigitMap

In addition to using standard dial plan syntaxes in a DM – e.g. ([78]xxxxxxxS0|04xxxxxxxxS0), the Obi has added the capability for referencing another digitmap from within a digitmap. In other words you can 'embed' the dial plan contents of another digitmap into the current digitmap you are configuring. This is achieved by using the '(M*label)*' syntax where the 'M' tells the DMP to refer to the contents of another digitmap and the '*label*' refers to the name of that particular digitmap – e.g. Msp1.

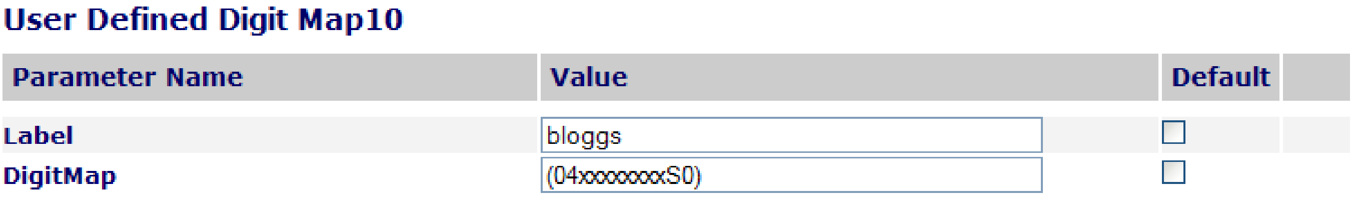
For example, if we wanted to send local calls via SP1 and mobile calls via SP2, we could add the rule ([78]xxxxxxxS0) to 'User Defined DM9' and give it a label of 'joe'. The digitmap can now be referenced as (Mjoe). We could also add the rule (04xxxxxxxxS0) to 'User Defined DM10' and give it a label of bloggs which can now be referenced as (Mbloggs) - and part of the 'dial plan' database would now look like:

ITSP Profile A	Msp1	Blank (from Part 1 config)
ITSP Profile B	Msp2	Blank (from Part 1 config)
User Defined DM 9	Mjoe	([78]xxxxxxxS0)
User Defined DM 10	Mbloggs	(04xxxxxxxxS0)



User Defined Digit Map9

Parameter Name	Value	Default	
Label	joe	☐	
DigitMap	([78]xxxxxxxS0)	☐	

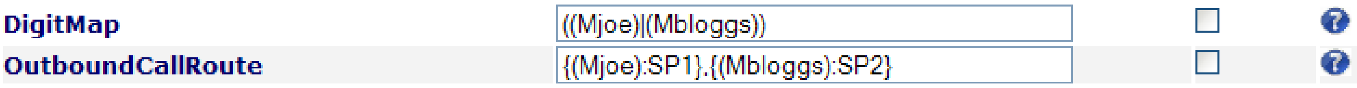


User Defined Digit Map10

Parameter Name	Value	Default	
Label	bloggs	☐	
DigitMap	(04xxxxxxxxS0)	☐	

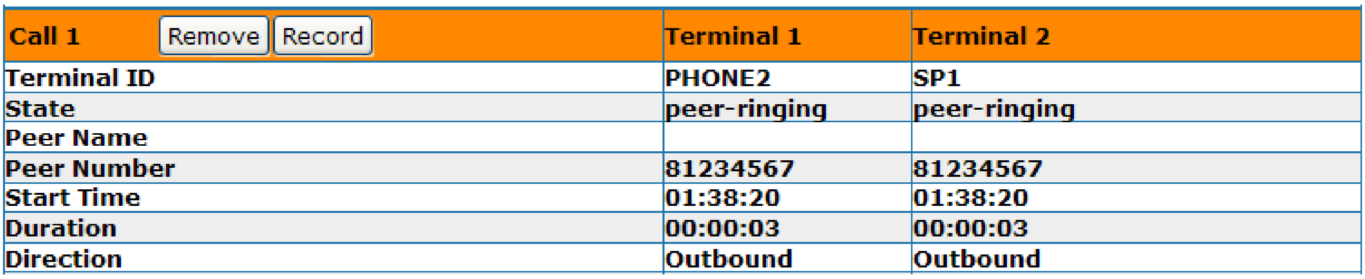
Using **Phone 2**, in this instance, our very simple PH2 config looks like:

PH2 DM – ((Mjoe)|(Mbloggs))
OBCR PH2 – {(Mjoe):SP1},{(Mbloggs):SP2}



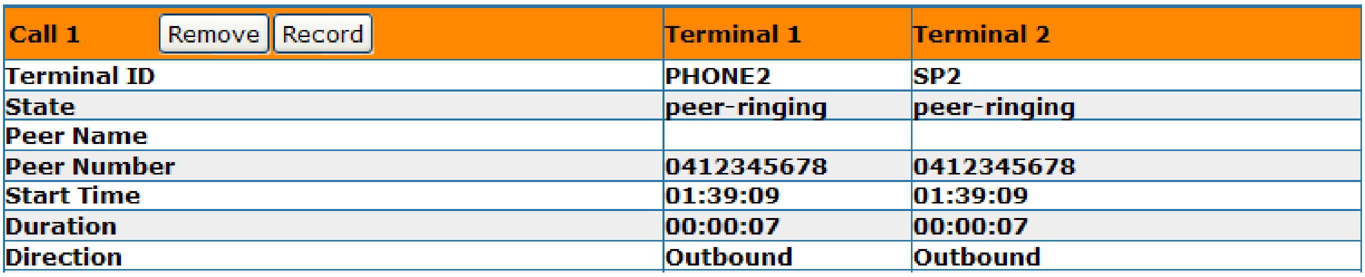
DigitMap	((Mjoe)|(Mbloggs))	☐	?
OutboundCallRoute	{(Mjoe):SP1},{(Mbloggs):SP2}	☐	?

A test call from PH2 to an 8 digit local number goes out via SP1:



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE2	SP1
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	81234567	81234567
Start Time	01:38:20	01:38:20
Duration	00:00:03	00:00:03
Direction	Outbound	Outbound

A test call from PH2 to a mobile goes out via SP2:



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE2	SP2
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	0412345678	0412345678
Start Time	01:39:09	01:39:09
Duration	00:00:07	00:00:07
Direction	Outbound	Outbound

Using ‘labelled’ DMs when crafting our ‘dial plans’ for outgoing calls allows us to be creative and reduce rule ‘clutter’ when compared to configuring a dial plan using the SPA method as we did in Part 1.

Hopefully, by now, ypu have a better understanding of the benefits of using the Obi digitmaps, especially when it comes to leveraging the capability of embedding digitmaps into a dial plan. With more of an insight into Obi DMs, let’s reconfigure our Obi202 using the same scenario and requirements from Part 1.

Clarifying the Call Flow Concept

Actually, before I go on I want to quickly touch on the Obi202 discussion thread on Whirlpool where comments have been posted regarding how a call is processed by the Obi and sent to a VSP. There appears to be some confusion and misunderstanding of the call flow concept which is compounded by a somewhat basic explanation in the Call Routing and Digit Map section of the Admin guide.

The confusion seems to be related mainly with how the OBCR processes the digits and whether the matched digits are sent directly out the interface to the VSP or whether the digits are re-evaluated one more time by the digitmap of the outgoing interface before being sent to the VSP.

Let’s re-examine the call flow of a mobile call using the corresponding rule from the config in Part 1 where the call is routed via PT on SP2, keeping in mind the fact that part of the original set up involved deleting the default entries for all SP DMs, leaving them blank:

- DM –.....|04xxxxxxxxS0|.....
- OBCR –..... {(04xxxxxxxx):SP2}...

As the digits 0412345678 are dialled, the DMP tests the digits against the PH1 DM rules and eventually matches the digits against the |04xxxxxxxxS0| rule (role of security guard fulfilled). Digits 0412345678 are then passed onto the PH1 OBCR. The DMP now needs to find information in order to determine where to route the received digits to. The digits are tested against the OBCR rules from left to right and a match is finally determined against the {(04xxxxxxxx):SP2} rule which tells the DMP to send the digits out via SP2 (job of the traffic cop fulfilled).

Since a test call was actually successful, this confirms that the digits were sent directly to the VSP and that the DM for SP2 had no part to play in the call set up process since the SP2 DM is blank. Otherwise, if the SP2 DM was involved, the call would have failed.

Furthermore, the call routing process can also be confirmed using the Phone 2 test above where digitmaps (Mjoe) and (Mbloggs) were used and the DMs for SP1 and SP2 were left blank.

The Obi Configuration

Let’s recap on our original scenario from Part 1 and reconfigure the Obi. I’ve factory defaulted my device to start afresh:

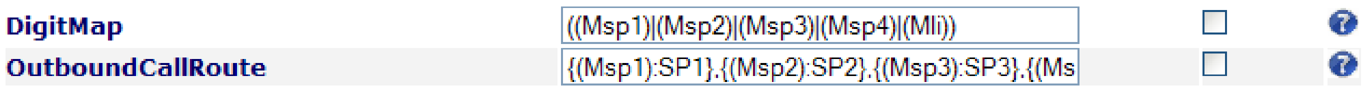
- The Obi202 has been reset to factory default & then the VSPs configured. (Don’t forget to re-point your SP services to the correct ITSP profile, otherwise only SP1 will register and the remainder won’t).
- I’ve deleted the default DM & OBCR entries for PH1 leaving them empty.
- The default DM entries for ISTP A/B/C/D (SP1-4) and LI can be left at default as we shall be over-writing them.
- Local/National calls to use MNF configured on SP1. (My local calls are SA based so amend yours for your state accordingly).
- Mobile calls to use PT (PennyTel) configured on SP2.
- 13/1300/1800 calls to use TC (Telecube) configured on SP3.
- International calls to use MaxoTel configured on SP4.
- 000 & 11xx calls to use the PSTN on LI.
- The dial plans that I’ll configure are by no means comprehensive and are used solely to demonstrate the principles in crafting dial plans to make outgoing calls via the various VSPs and PSTN. Therefore, do not rely on what I present as being a configuration that will cover every call scenario.

First off, we’ll amend the DMs for SP1-4 and LI to cater for our dialling requirements:

- **ITSP A DM** – (<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0)
- **ITSP B DM** – (04xxxxxxxxS0)
- **ITSP C DM** – (13[1-9]xxxS0|1[38]xxxxxxxxS0)
- **ITSP D DM** – (0011xx.)
- **LI DM** – (000S0 |11xxS0)

Next, we configure the DM & OBCR for PH1:

DM – ((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli))
OBCR – {(Msp1):SP1},{(Msp2):SP2},{(Msp3):SP3},{(Msp4):SP4},{(Mli):Li}



DigitMap	((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli))	☐	?
OutboundCallRoute	{(Msp1):SP1},{(Msp2):SP2},{(Msp3):SP3},{(Ms	☐	?

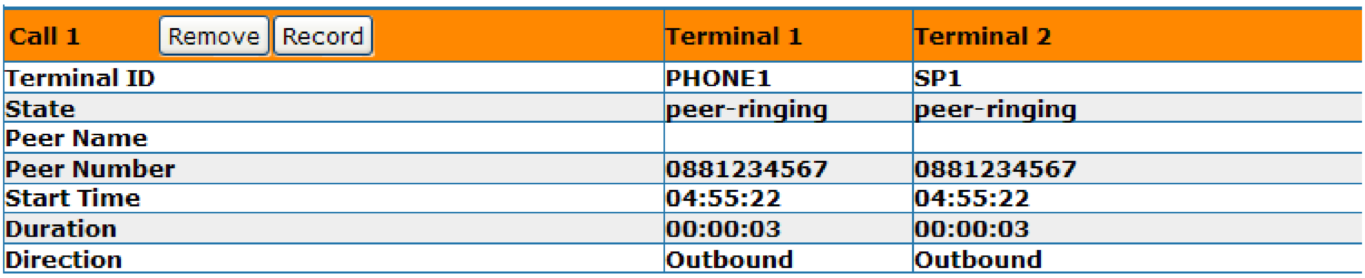
Our ‘dial plan’ database now looks like:

Interface	Ref As	Dial Plan
ITSP Profile A	Msp1	(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0)
ITSP Profile B	Msp2	(04xxxxxxxxS0)
ITSP Profile C	Msp3	(13[1-9]xxxS0|1[38]xxxxxxxxS0)
ITSP Profile D	Msp4	(0011xx.)
Line Port	Mli	(000S0 |11xxS0)
Phone Port 1	Mph1	((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli))

To prove that our config is functioning correctly, it’s time for some test calls to ensure the correct call type is routed via the desired interface & VSP.

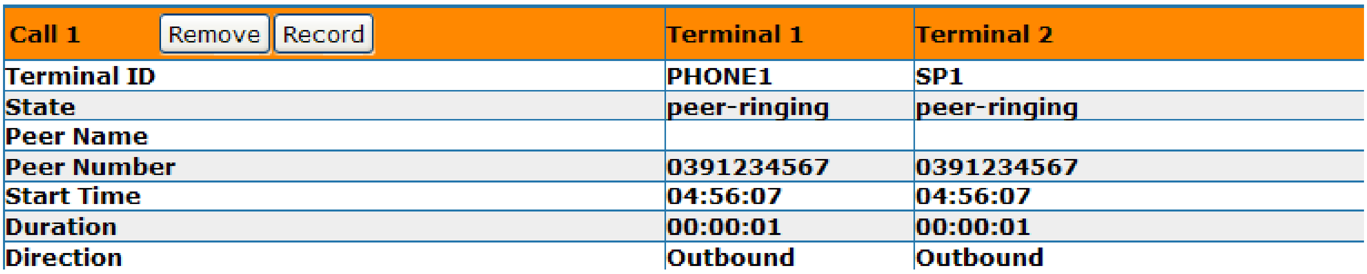
The following test calls were made successfully with no delay experienced after dialling the last digit for each number:

Local 8123 4567 via MNF on SP1.



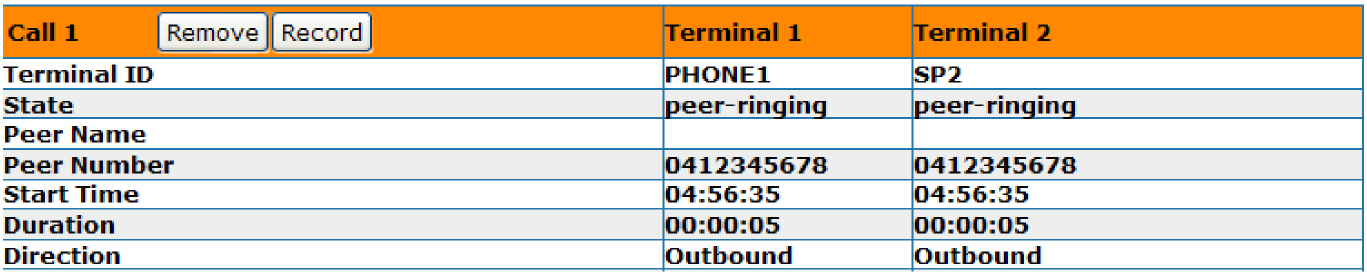
Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP1
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	0881234567	0881234567
Start Time	04:55:22	04:55:22
Duration	00:00:03	00:00:03
Direction	Outbound	Outbound

National 03 9123 4567 via MNF on SP1.



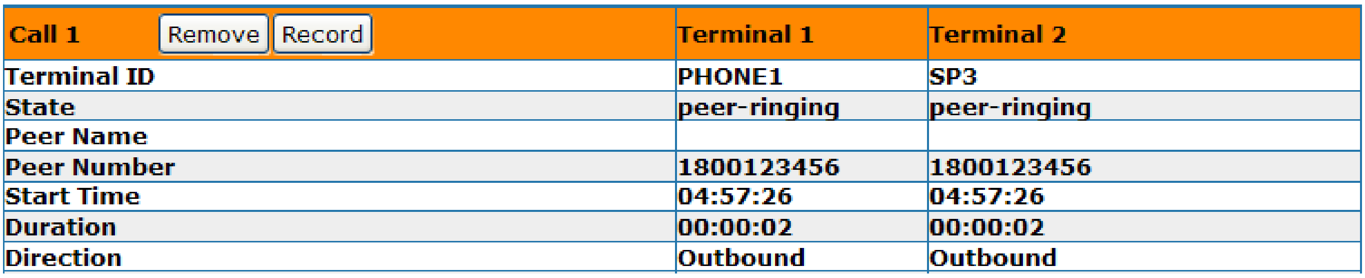
Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP1
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	0391234567	0391234567
Start Time	04:56:07	04:56:07
Duration	00:00:01	00:00:01
Direction	Outbound	Outbound

Mobile 0412 345 678 via PT on SP2.



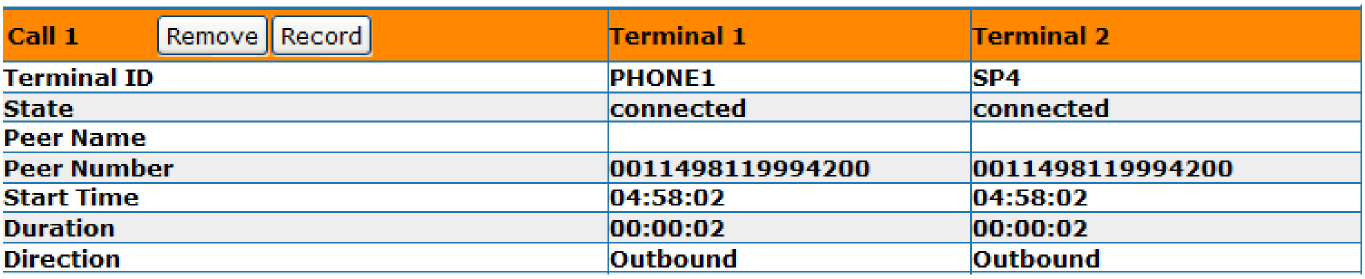
Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP2
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	0412345678	0412345678
Start Time	04:56:35	04:56:35
Duration	00:00:05	00:00:05
Direction	Outbound	Outbound

1800 123 456 via TC on SP3.



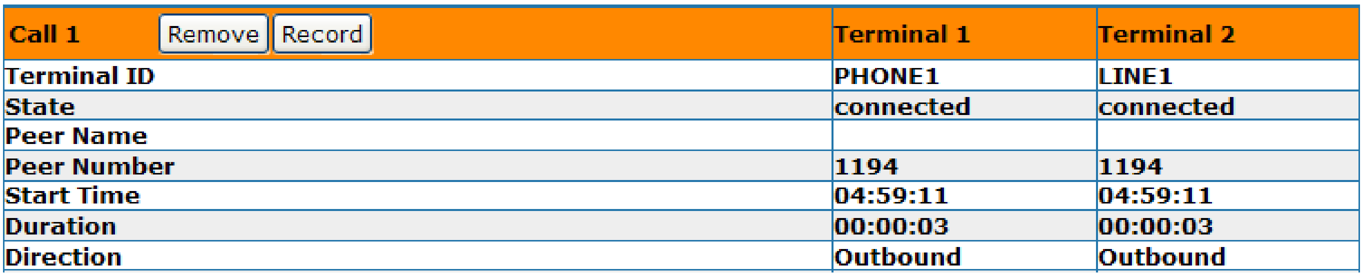
Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP3
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	1800123456	1800123456
Start Time	04:57:26	04:57:26
Duration	00:00:02	00:00:02
Direction	Outbound	Outbound

International 0011 49 811 999 4200 via MaxoTel on SP4.



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP4
State	connected	connected
Peer Name		
Peer Number	0011498119994200	0011498119994200
Start Time	04:58:02	04:58:02
Duration	00:00:02	00:00:02
Direction	Outbound	Outbound

1194 via PSTN on LI.



Call 1 Remove Record	Terminal 1	Terminal 2
Terminal ID	PHONE1	LINE1
State	connected	connected
Peer Name		
Peer Number	1194	1194
Start Time	04:59:11	04:59:11
Duration	00:00:03	00:00:03
Direction	Outbound	Outbound

Adding Access Codes

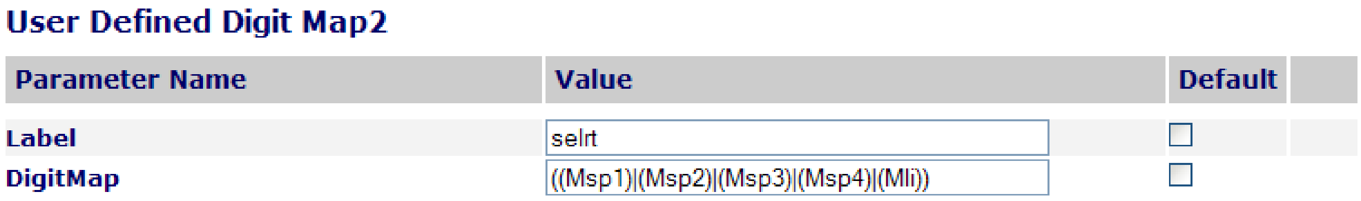
Now, let's modify the PH1 DM & OBCR and add the access codes so that we can force any call type via any interface:

DM – ((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)|**[1-46](Msp1)|**[1-46](Msp2)|**[1-46](Msp3)|**[1-46](Msp4)|**[1-46](Mli))

OBCR – {(Msp1):SP1},{(Msp2):SP2},{(Msp3):SP3},{(Msp4):SP4},{(Mli):Li},{(<**1:>xx.):SP1},{(<**2:>xx.):SP2},{(<**3:>xx.):SP3},{(<**4:>xx.):SP4},{(<**6:>xx.):LI}

With this config, making various test calls to different interfaces were successful. But we can still refine and condense the PH1 DM, making it somewhat smaller by utilising just one more digitmap.

Let's use spare User Defined DigitMap2 and give it a user friendly name - selrt (short for select route) and we'll add the existing DMs to it like so:



User Defined Digit Map2

Parameter Name	Value	Default
Label	selrt	☐
DigitMap	((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli))	☐

At this point our current dial plan database looks like:

Interface	Ref As	Dial Plan
ITSP Profile A	Msp1	(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0)
ITSP Profile B	Msp2	(04xxxxxxxxS0)
ITSP Profile C	Msp3	(13[1-9]xxxS0|1[38]xxxxxxxxS0)
ITSP Profile D	Msp4	(0011xx.)
Line Port	Mli	(000S0 |11xxS0)
Phone Port 1	Mph1	((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)|**[1-46](Msp1)|**[1-46](Msp2)|**[1-46](Msp3)|**[1-46](Msp4)|**[1-46](Mli))
User Defined DM 2	Mselrt	((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli))

Now we amend the PH1 DM by deleting **[1-46](Msp1)|**[1-46](Msp2)|**[1-46](Msp3)|**[1-46](Msp4)|**[1-46](Mli) that we had recently entered and replace it with the new (Mselrt) digitmap we just created, leaving the OBCR as is :

DM – ((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)~~|**[1-46](Msp1)|**[1-46](Msp2)|**[1-46](Msp3)|**[1-46](Msp4)|**[1-46](Mli)~~)

DM – ((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)|**[1-46](Mselrt))

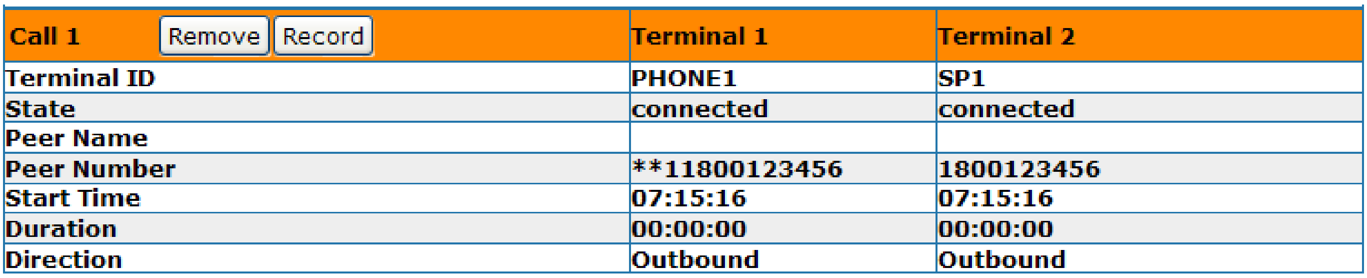
OBCR – {(Msp1):SP1},{(Msp2):SP2},{(Msp3):SP3},{(Msp4):SP4},{(Mli):Li},{(<**1:>xx.):SP1},{(<**2:>xx.):SP2},{(<**3:>xx.):SP3},{(<**4:>xx.):SP4},{(<**6:>xx.):LI}

With the dial plan database now looking like:

<table>
<tr><th>Interface</th><th>Ref As</th><th>Dial Plan</th></tr>
<tr><td>ITSP Profile A</td><td>Msp1</td><td>(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0)</td></tr>
<tr><td>ITSP Profile B</td><td>Msp2</td><td>(04xxxxxxxxS0)</td></tr>
<tr><td>ITSP Profile C</td><td>Msp3</td><td>(13[1-9]xxxS0|1[38]xxxxxxxxS0)</td></tr>
<tr><td>ITSP Profile D</td><td>Msp4</td><td>(0011xx.)</td></tr>
<tr><td>Line Port</td><td>Mli</td><td>(000S0 |11xxS0)</td></tr>
<tr><td>Phone Port 1</td><td>Mph1</td><td>((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)|**[1-46](Mselrt))</td></tr>
<tr><td>User Defined DM 2</td><td>Mselrt</td><td>((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli))</td></tr>
</table>

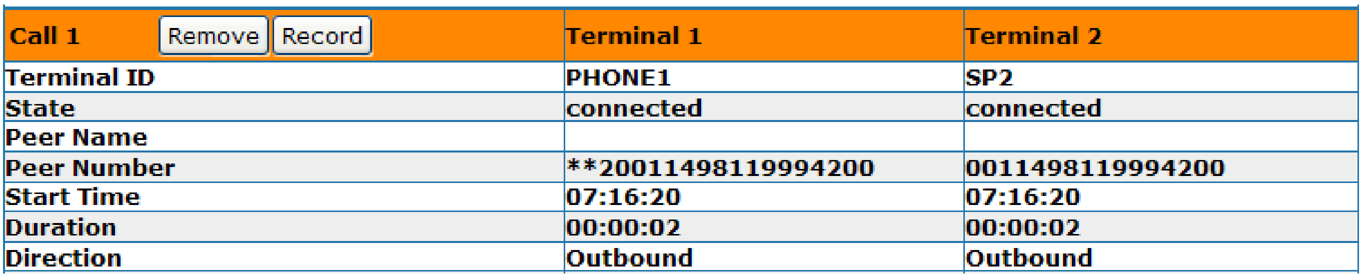
Let's make some test calls forcing them out via specific interfaces:

1800 via SP1:



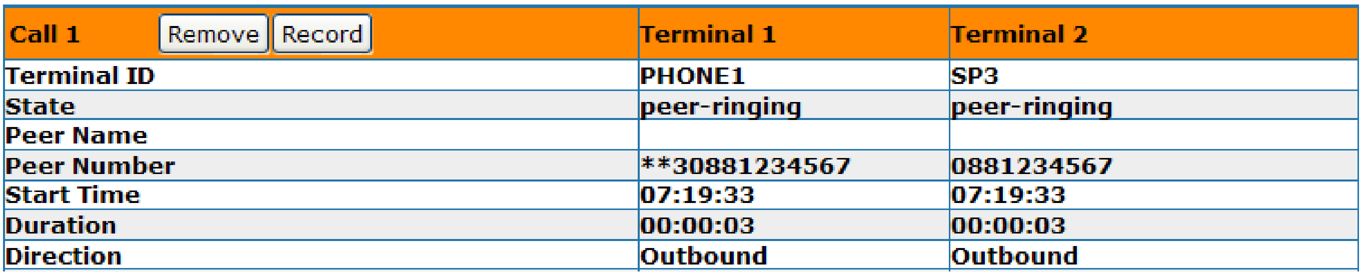
Call 1 Remove Record	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP1
State	connected	connected
Peer Name		
Peer Number	**11800123456	1800123456
Start Time	07:15:16	07:15:16
Duration	00:00:00	00:00:00
Direction	Outbound	Outbound

International via SP2:



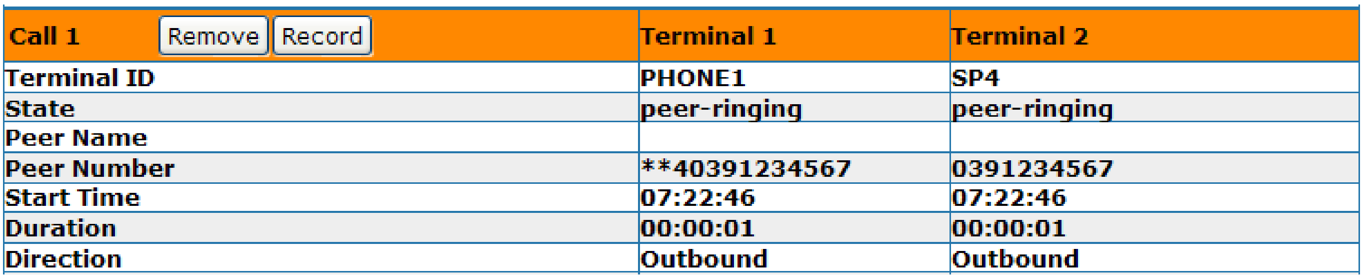
Call 1 Remove Record	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP2
State	connected	connected
Peer Name		
Peer Number	**20011498119994200	0011498119994200
Start Time	07:16:20	07:16:20
Duration	00:00:02	00:00:02
Direction	Outbound	Outbound

Local via SP3:



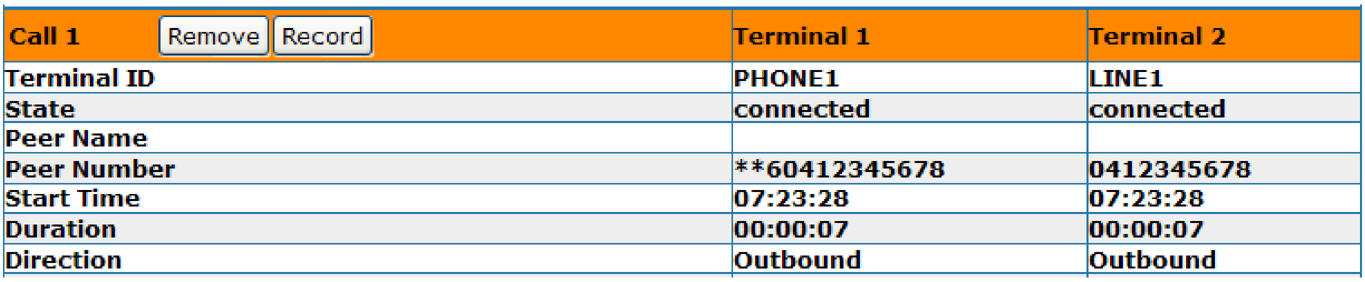
Call 1 Remove Record	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP3
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	**30881234567	0881234567
Start Time	07:19:33	07:19:33
Duration	00:00:03	00:00:03
Direction	Outbound	Outbound

National via SP4:



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	SP4
State	peer-ringing	peer-ringing
Peer Name		
Peer Number	**40391234567	0391234567
Start Time	07:22:46	07:22:46
Duration	00:00:01	00:00:01
Direction	Outbound	Outbound

Mobile via PSTN:



Call 1	Terminal 1	Terminal 2
Terminal ID	PHONE1	LINE1
State	connected	connected
Peer Name		
Peer Number	**60412345678	0412345678
Start Time	07:23:28	07:23:28
Duration	00:00:07	00:00:07
Direction	Outbound	Outbound

Part 3

Summary

We can now look back over how we configured the Obi – the first method using the SPA style approach and the second method using the Obi approach to achieve the same end results in both cases.

With the first method there was quite a lot of typing required both for the DM & OBCR fields to achieve the desired result. The PH1 DM & OBCR looked like this:

DM

```
(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0|04xxxxxxxxS0|13[1-9]xxxS0|1[38]xxxxxxxxS0|0011xx.|000S0|11xxS0|
**1[78]xxxxxxxS0|**10[2378]xxxxxxxxS0|**104xxxxxxxxS0|**113[1-9]xxxS0|**11[38]xxxxxxxxS0|**10011xx.|
**1000S0|**111xxS0|**2[78]xxxxxxxS0|**20[2378]xxxxxxxxS0|**204xxxxxxxxS0|**213[1-9]xxxS0|
**21[38]xxxxxxxxS0|**20011xx.|**2000S0|**211xxS0|**3[78]xxxxxxxS0|**30[2378]xxxxxxxxS0|
**304xxxxxxxxS0|**313[1-9]xxxS0|**31[38]xxxxxxxxS0|**30011xx.|**3000S0|**311xxS0|**4[78]xxxxxxxS0|
**40[2378]xxxxxxxxS0|**404xxxxxxxxS0|**413[1-9]xxxS0|**41[38]xxxxxxxxS0|**40011xx.|**4000S0|
**411xxS0|**6[78]xxxxxxxS0|**60[2378]xxxxxxxxS0|**604xxxxxxxxS0|**613[1-9]xxxS0|**61[38]xxxxxxxxS0|
**60011xx.|**6000S0|**611xxS0)
```

OBCR

```
{(0[2378]xxxxxxxx):SP1},{(04xxxxxxxx):SP2},{(13[1-9]xxx|1[38]xxxxxxxx):SP3},{(0011xx.):SP4},{(000|11xx):LI},
{(<**1:08>[78]xxxxxxxS0|<**1:>0[2378]xxxxxxxxS0|<**1:>04xxxxxxxxS0|<**1:>13[1-9]xxxS0|
<**1:>1[38]xxxxxxxxS0|<**1:>0011xx.|<**1:>000S0|<**1:>11xxS0):SP1},{(<**2:08>[78]xxxxxxxS0|
<**2:>0[2378]xxxxxxxxS0|<**2:>04xxxxxxxxS0|<**2:>13[1-9]xxxS0|<**2:>1[38]xxxxxxxxS0|<**2:>0011xx.|
<**2:>000S0|<**2:>11xxS0):SP2},{(<**3:08>[78]xxxxxxxS0|<**3:>0[2378]xxxxxxxxS0|<**3:>04xxxxxxxxS0|
<**3:>13[1-9]xxxS0|<**3:>1[38]xxxxxxxxS0|<**3:>0011xx.|<**3:>000S0|<**3:>11xxS0):SP3},
{(<**4:08>[78]xxxxxxxS0|<**4:>0[2378]xxxxxxxxS0|<**4:>04xxxxxxxxS0|<**4:>13[1-9]xxxS0|
```

<**4:>1[38]xxxxxxxxS0|<**4:>0011xx.|<**4:>000S0|<**4:>11xxS0):SP4},{(<**6:08>[78]xxxxxxxS0|<**6:>0[2378]xxxxxxxxS0|<**6:>04xxxxxxxxS0|<**6:>13[1-9]xxxS0|<**6:>1[38]xxxxxxxxS0|<**6:>0011xx.|<**6:>000S0|<**6:>11xxS0):LI}

Now compare all those rules to the second method we used to configure the Obi:

DM – ((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)|**[1-46](Mselrt))

OBCR – {(Msp1):SP1},{(Msp2):SP2},{(Msp3):SP3},{(Msp4):SP4},{(Mli):Li},{(<**1:>xx.):SP1},{(<**2:>xx.):SP2},{(<**3:>xx.):SP3},{(<**4:>xx.):SP4},{(<**6:>xx.):LI}

Interface	Ref As	Dial Plan
ITSP Profile A	Msp1	(<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0)
ITSP Profile B	Msp2	(04xxxxxxxxS0)
ITSP Profile C	Msp3	(13[1-9]xxxS0|1[38]xxxxxxxxS0)
ITSP Profile D	Msp4	(0011xx.)
Line Port	Mli	(000S0 |11xxS0)
Phone Port 1	Mph1	((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)|**[1-46](Mselrt))
User Defined DM 2	Mselrt	((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli))

There is definitely a lot less typing involved and probably easier to spot a potential error. The ability to utilise multiple digitmaps as well as reference them makes it easier to configure the Obi in 'bite size' pieces combining multiple digitmaps as building blocks in order to construct powerful dialling capabilities.

When it comes to how the DMP 'looks' at the underlying structure of our configuration, where we as a user see a string of rules for the PH1 DM as **((Msp1)|(Msp2)|(Msp3)|(Msp4)|(Mli)|**[1-46](Mselrt))**, the DMP sees those rules as:

DM

((<:08>[78]xxxxxxxS0|0[2378]xxxxxxxxS0|04xxxxxxxxS0|13[1-9]xxxS0|1[38]xxxxxxxxS0|0011xx.|000S0|11xxS0|**1<:08>[78]xxxxxxxS0|**10[2378]xxxxxxxxS0|**104xxxxxxxxS0|**113[1-9]xxxS0|**11[38]xxxxxxxxS0|**10011xx.|**1000S0|**111xxS0|**2<:08>[78]xxxxxxxS0|**20[2378]xxxxxxxxS0|**204xxxxxxxxS0|**213[1-9]xxxS0|**21[38]xxxxxxxxS0|**20011xx.|**2000S0|**211xxS0|**3<:08>[78]xxxxxxxS0|**30[2378]xxxxxxxxS0|**304xxxxxxxxS0|**313[1-9]xxxS0|**31[38]xxxxxxxxS0|**30011xx.|**3000S0|**311xxS0|**4<:08>[78]xxxxxxxS0|**40[2378]xxxxxxxxS0|**404xxxxxxxxS0|**413[1-9]xxxS0|**41[38]xxxxxxxxS0|**40011xx.|**4000S0|**411xxS0|**6<:08>[78]xxxxxxxS0|**60[2378]xxxxxxxxS0|**604xxxxxxxxS0|**613[1-9]xxxS0|**61[38]xxxxxxxxS0|**60011xx.|**6000S0|**611xxS0)

which is essentially what we had entered as our dial plan for the first method in Part 1.

Finally, just to reiterate, the dialling rules used in the scenario presented here were fairly limited and made no allowance for dialling *** to access the IVR, # for calling phone 2 or even ## for directly accessing the PSTN line. Remember this when crafting your own dialling rules.

Hopefully, this document has helped somewhat in de-mystifying the concepts of DigitMaps & Outbound CallRoutes and is meant to complement the tutorial already presented by Mark W.