Avaya Solution & Interoperability Test Lab

Application Notes for NMS Communications Vision Media Gateway VG2000 and Vision Signaling Server VS5000 with Avaya Voice Portal and Avaya Aura™ Session Manager - Issue 1.0

Abstract

These Application Notes describe the configuration steps required to enable calls between Avaya Voice Portal and the PSTN through a SIP infrastructure consisting of Avaya Aura™ Session Manager, SIP Endpoints registered with Avaya Aura™ SIP Enablement Services, H.323 Endpoints registered with Avaya Aura™ Communication Manager and the NMS Communications Vision Media Gateway VG2000. This solution allows Avaya Voice Portal to receive calls from the PSTN and transfer calls to Avaya SIP telephones or the PSTN. The Vision Media Gateway VG2000 combines signaling and media gateway functions providing PSTN access to SIP-based telephony networks.

Information in these Application Notes has been obtained through DevConnect compliance testing and additional technical discussions. Testing was conducted via the DevConnect Program at the Avaya Solution and Interoperability Test Lab.

Table of Contents

1. Introduction

These Application Notes present a sample configuration for a network that uses Avaya Voice Portal and the PSTN through a SIP infrastructure consisting of Avaya Aura™ Session Manager R1.1, Avaya Aura™ Communication Manager, Avaya Voice Portal and the NMS Communications Vision Media Gateway VG2000. This solution allows Voice Portal to receive/and transfer calls from/to the PSTN and SIP. The Vision Media Gateway VG2000 combines signaling and media gateway functions providing PSTN access to SIP-based telephony networks. In this configuration, the VG2000 connects to Voice Portal through SIP trunks on Session Manager. The VG2000 has separate network connections for IP-based call signaling (SIP) and for IP-based media (RTP). The Vision Signaling Server or VS5000 provides an interface to the SS7 network using the ISUP signaling protocol and terminates the SS7 signaling link. Refer to **Figure 1** for an illustration of the test configuration. Session Manager using its SM-100 (Security Module) network interface, routes the calls between the different entities using SIP Trunks. All inter-system calls are carried over these SIP trunks. Session Manager supports flexible inter-system call routing based on the dialed number, the calling number and the system location; it can also provide protocol adaptation to allow multi-vendor systems to interoperate. Session Manager is managed by Avaya Aura™ System Manager via the management network interface. Configurations supporting SIP telephones for the version here presented require Avaya Aura™ SIP Enablement Services with endpoints configured as OPTIM extensions on Communication Manager.

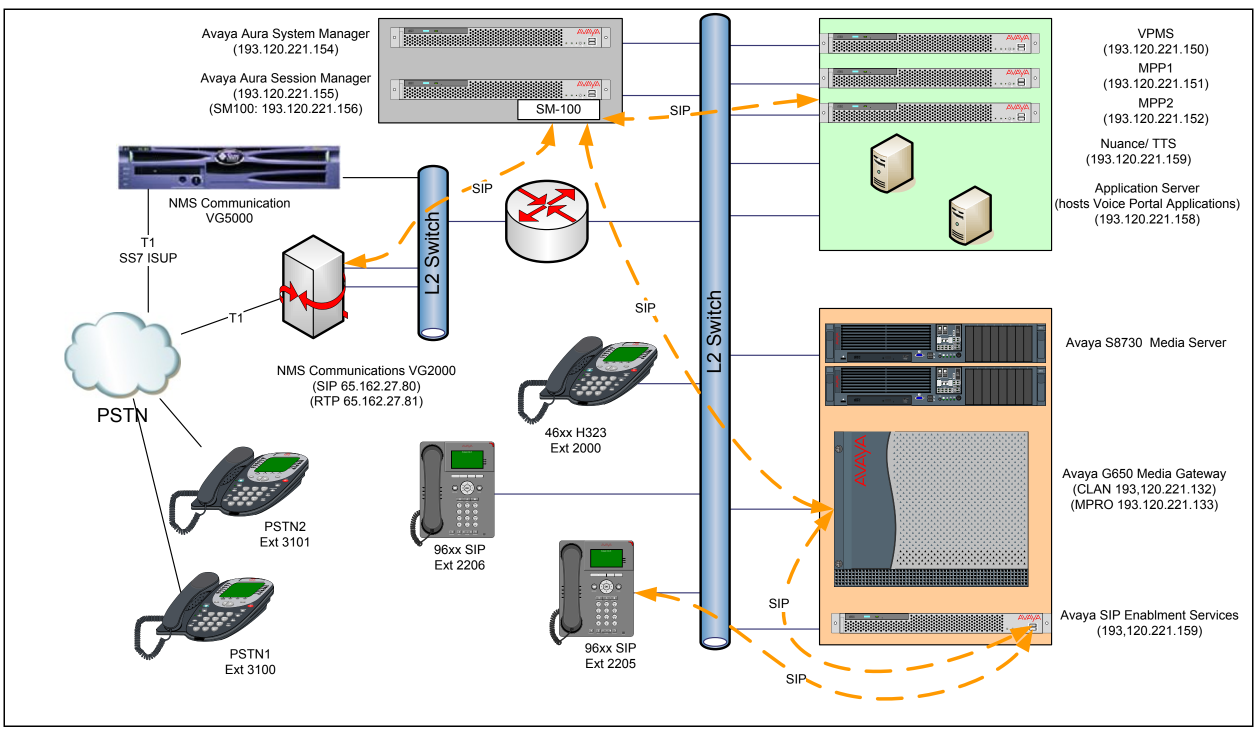


Figure 1 – Test Configuration of VG2000, Voice Portal, and Session Manager

For the sample configuration shown in **Figure 1**, Session Manager runs on an Avaya S8510 Server, Communication Manager 5.2 runs on an Avaya S8730 Server with an Avaya G650 Media Gateway, and Voice Portal runs on an Avaya S8510 Server. The results in these Application Notes are applicable to other Communication Manager Server and Media Gateway combinations. A four digit Uniform Dial Plan (UDP) is used for dialing between systems. Unique extension ranges are associated with Communication Manager 5.2 (2xxx) and Voice Portal (6000) and the PSTN (3xxx). These Application Notes will focus on the configuration of the SIP trunks and call routing. Detailed administration of the endpoint telephones will not be described. Refer to the appropriate documentation in **Reference [1] and [2]** for more details.

2. Equipment and Software Validated

The following equipment and software were used for the sample configuration provided:

Product / Hardware Platform	Software Version
Avaya S8510 Server	Avaya Aura™ SIP Enablement Services Home/Edge 5.2 SES-5.2.0.0-947.3b
Avaya S8510 Server with SM-100 card	Avaya Aura™ Session Manager 1.1 (1.1.4.0.111005) PASS2.1
Avaya S8510 Server	Avaya Aura™ System Manager 1.1 (1.1.4.0.111005)
Avaya Aura™ Voice Portal • Avaya S8510 Server	5.1 SP1 (5.0.0.1.0701)
Avaya S8730 Media Server	Avaya Aura™ Communication Manager 5.2.0 (S8730-015-02.0.947.3) With patch 02.0.947.3-17534
Avaya G650 Media Gateway IPSI (TN2312BP) C-LAN (TN799DP) IP Media Resource 320 (TN2602AP) Digital line	 TN2312BP HW28 FW046 TN799DP HW16 FW032 TN2602AP HW08 FW048 TN2214CP HW10 FW015
Avaya IP Telephones: 9630 & 9620 (SIP) 9620 (H323) 1616 (H323) 4621 (H323) Avaya Digital Telephones (2420)	 Avaya one-X Deskphone SIP 2.4.1 Avaya one-X Deskphone S3.00 Release 1.2000 Release 2.9.1 N/A
NMS Communications SR 1500 Server	NMS Vision Gateway VG 2000 version 2.0 patch #6395
NMS Communications Vision Signaling Server Model VS5000 with TX4000 SS7 Signaling Board	VS5000 Version 1.0 patch #6112

3. Configure Communication Manager

This section provides the procedures for configuring Communication Manager. The procedures include the following areas:

- Verify Communication Manager license
- Configure IP node names
- Verify/List IP interfaces
- Configure Codec Set
- Configure Network Region
- Administer a SIP Trunk to Session Manager
- Configure Route Pattern
- Configure Location and Public Unknown Numbering
- Administer Uniform Dial Plan and AAR Analysis

Throughout this section the administration of Communication Manager is performed using a System Access Terminal (SAT), the following commands are entered on the system with the appropriate administrative permissions. Some administration screens have been abbreviated for clarity. These instructions assume that the Communication Manager has been installed, configured, licensed and provided with a functional dial plan. Refer to the appropriate documentation as described in **Reference [1] and [2]** for more details**.** In these Application Notes the system was configured with a 4 digit uniform dialplan, in which number as **2xxx** are assigned to station and **3xxx** (the PSTN stations) **6xxx** (Voice Portal) to **aar** table. Dialplan analisys can be verified with the **display dialplan analysis** command**.**

```
display dialplan analysis                                   Page   1 of  12
                                DIAL PLAN ANALYSIS TABLE
                                      Location:  all          Percent Full:    1

       Dialed    Total  Call     Dialed    Total  Call      Dialed    Total  Call
       String    Length Type     String    Length Type      String    Length Type
     2             4    ext
     3             4    aar
     6             4    aar
     8             3    dac
     *9            3    fac
```

The SIP endpoints and the integration with SIP Enablement Server are configured as described in documents available from **Reference [1] and [2].**

3.1 Verify Communication Manager License

Use the **display system-parameters customer-options** command. Navigate to **Page 2** and verify that there is sufficient remaining capacity for SIP trunks by comparing the **Maximum Administered SIP Trunks** field value with the corresponding value in the **USED** column. The difference between the two values needs to be greater than or equal to the desired number of simultaneous SIP trunk connections. Verify highlighted value, as shown below.

```
display system-parameters customer-options                      Page   2 of  10
                                OPTIONAL FEATURES

IP PORT CAPACITIES                                              USED
                     Maximum Administered H.323 Trunks: 100     0
           Maximum Concurrently Registered IP Stations: 18000   2
             Maximum Administered Remote Office Trunks: 0       0
Maximum Concurrently Registered Remote Office Stations: 0       0
              Maximum Concurrently Registered IP eCons: 0       0
  Max Concur Registered Unauthenticated H.323 Stations: 100     0
                        Maximum Video Capable Stations: 100     0
                   Maximum Video Capable IP Softphones: 100     9
                       Maximum Administered SIP Trunks: 1000    300
```

If there is insufficient capacity of SIP Trunks or a required feature is not enabled, contact an authorized Avaya sales representative to make the appropriate changes.

3.2 Configure IP Node Names

As SIP interaction with Session Manager is carried through the security module SM100 IP interface, in configuring the SIP Trunk refer to its IP address. Use the **change node-names ip** command to add the **Name** and **IP Address** for the Session Manager, in the example **SM100** and **193.120.221.156** .

```
change node-names ip                                            Page   1 of   2
                                  IP NODE NAMES
    Name                 IP Address
Gateway001             193.120.221.129
SM100                  193.120.221.156
clan                   193.120.221.132
default                0.0.0.0
mpro                   193.120.221.133
procr                  0.0.0.0
ses                    193.120.221.159
```

Note that in the example some other values (CLAN, MedPro, SES) have been already created as per installation and configuration of Communication Manager and SIP Enablement Services.

3.3 Verify/List IP Interfaces

Use the **list ip-interface all** command and note the **C-LAN** to be used for SIP trunks between the Communication Manager and the Session Manager.

```
list ip-interface all
                                 IP INTERFACES
                                                                        Net
ON Type   Slot  Code/Sfx    Node Name/       Mask  Gateway Node     Rgn VLAN
                            IP-Address
-- ------ ----- --------    ---------------  ----  ---------------  ---  ----
 y C-LAN  01A02 TN799  D     clan             /25   Gateway001       1    n
                             193.120.221.132
 y MEDPRO 01A03 TN2602       mpro             /25   Gateway001       1    n
                             193.120.221.133
```

3.4 Configure IP Codec Sets

Use the **change ip-codec-set n** command where **n** is codec set used in the configuration. The VG2000 supports both G.711 and G.729 to have both available in the network region configure as it follows:

- **Audio Codec** set for **G.729AB** as first codec and **G.711MU** as second
- **Silence Suppression**: Retain the default value **n**
- **Frames Per Pkt:** Enter **2**
- **Packet Size (ms):** Enter **20**

Retain the default values for the remaining fields, and submit these changes.

```
change ip-codec-set 1                                      Page   1 of   2
                          IP Codec Set

    Codec Set: 1

    Audio        Silence      Frames   Packet
    Codec        Suppression  Per Pkt  Size(ms)
 1: G.729AB           n          2        20
 2: G.711MU           n          2        20
 3:
```

3.5 Configure IP Network Region

Use the **change ip-network-region n** command where **n** is the number of the network region used and set the **Intra-region IP-IP Direct Audio**, and **Inter-region IP-IP Direct Audio** fields to **yes**. For the **Codec Set** enter the corresponding audio codec set configured in **Section 3.4**. Set the **Authoritative Domain** to the SIP domain. Retain the default values for the remaining fields, and submit these changes.

Note. In the test configuration, **network region 1** was used. If you are creating a new network region or modifying another one, ensure to configure it with the correct parameters.

```
change ip-network-region 1                                    Page   1 of  19
                               IP NETWORK REGION
  Region: 1
Location: 1       Authoritative Domain: avaya.com
    Name: CallCenter
MEDIA PARAMETERS                Intra-region IP-IP Direct Audio: yes
      Codec Set: 1              Inter-region IP-IP Direct Audio: yes
   UDP Port Min: 2048                      IP Audio Hairpinning? n
   UDP Port Max: 3329
```

3.6 Administer a SIP Trunk to Session Manager

To administer a SIP Trunk on Communication Manger, two intermediate steps are required, creation of a signaling group and trunk group.

3.6.1 Add SIP Signaling group

Use the **add signaling-group n** command, where **n** is an available signaling group number, for one of the SIP trunks to the Session Manager, and fill in the indicated fields. Default values can be used for the remaining fields:

- **Group Type:** **sip**
- **Transport Method:** **tcp**
- **Near-end Node Name:** C-LAN node name from **Section 3.2** (i.e., **clan**)
- **Far-end Node Name:** Session Manager node name from **Section 3.2** (i.e., **SM100**)
- **Near-end Listen Port:** **5060**
- **Far-end Listen Port:** **5060**
- **Far-end Domain:** **avaya.com**
- **DTMF over IP:** **rtp-payload**

```
add signaling-group 20                                          Page   1 of   1
                                SIGNALING GROUP

 Group Number: 20             Group Type: sip
                         Transport Method: tcp
  IMS Enabled? n

   Near-end Node Name: clan                   Far-end Node Name: SM100
 Near-end Listen Port: 5060                 Far-end Listen Port: 5060
                                         Far-end Network Region: 1
Far-end Domain: avaya.com

                                            Bypass If IP Threshold Exceeded? n

         DTMF over IP: rtp-payload           Direct IP-IP Audio Connections? y
Session Establishment Timer(min): 3                    IP Audio Hairpinning? n
         Enable Layer 3 Test? n                     Direct IP-IP Early Media? n
H.323 Station Outgoing Direct Media? n          Alternate Route Timer(sec): 6
```

3.6.2 Configure a SIP Trunk Group

Add the corresponding trunk group controlled by this signaling group via the **add trunk-group n** command, where **n** is an available trunk group number and fill in the indicated fields.

- **Group Type:** **sip**
- **Group Name:** A descriptive name (i.e**., to AuraSM**)
- **TAC:** An available trunk access code (i.e., **820**)
- **Service Type:** **tie**
- **Signaling Group:** The number of the signaling group added in **Section 3.6.1** (i.e. **20**)
- **Number of Members:** The number of SIP trunks to be allocated to calls routed to Session Manager (must be within the limits of the total trunks available from licensed verified in **Section 3.1**)

Note: The number of members determines how many simulataneous calls can be processed by the trunk through Session Manager.

```
add trunk-group 20                                            Page   1 of  21
                                TRUNK GROUP

Group Number: 20                   Group Type: sip          CDR Reports: y
  Group Name: to AuraSM                   COR: 1        TN: 1      TAC: 820
   Direction: two-way       Outgoing Display? y
 Dial Access? n                                  Night Service:
Queue Length: 0
Service Type: tie                    Auth Code? n
                                                    Signaling Group: 20
                                                  Number of Members: 200
```

Navigate to **Page 3** and change **Numbering Format** to **public.** Use default values for all other fields. Submit these changes.

```
add trunk-group 20                                            Page   3 of  21
TRUNK FEATURES
          ACA Assignment? n            Measured: none
                                                        Maintenance Tests? y

                     Numbering Format: public
                                              UUI Treatment: service-provider

                                                Replace Restricted Numbers? n
                                               Replace Unavailable Numbers? n
```

Configure Route Pattern

Configure a route pattern to correspond to the newly added SIP trunk group. Use **change route pattern n** command, where **n** is an available route pattern. Enter the following values for the specified fields, and retain the default values for the remaining fields. Submit these changes.

- **Pattern Name:** A descriptive name (i.e., **to AuraSM**)
- **Grp No:** The trunk group number from **Section 3.6.2**
- **FRL:** Enter a level that allows access to this trunk, with **0** being least **restrictive**.

```
change route-pattern 20                                       Page   1 of   3
                    Pattern Number: 20  Pattern Name: to AuraSM
                             SCCAN? n     Secure SIP? n
    Grp FRL NPA Pfx Hop Toll No.  Inserted                            DCS/ IXC
    No          Mrk Lmt List Del  Digits                              QSIG
                             Dgts                                     Intw
 1: 20   0                                                             n   user
 2:                                                                    n   user
 3:                                                                    n   user
 4:                                                                    n   user
 5:                                                                    n   user
 6:                                                                    n   user

     BCC VALUE  TSC CA-TSC   ITC BCIE Service/Feature PARM  No. Numbering LAR
    0 1 2 M 4 W     Request                                 Dgts Format
                                                            Subaddress
 1: y y y y y n  n            rest                               none
```

3.7 Configure Location and Public Unknown Numbering

Use the **change locations** command to assign the SIP route pattern for Avaya SIP endpoints to a location corresponding to the **Main** site. Add an entry for the Main site if one does not exist already. Enter the following values for the specified fields, and retain default values for the remaining fields. Submit these changes.

- **Name:** A descriptive name to denote the Main site.
- **Timezone:** An appropriate time zone offset.
- **Rule:** An appropriate daylight savings rule (i.e., **0**)
- **Proxy Sel. Rte. Pat.:** The route pattern number from **Section 3.7**. (i.e., **20**)

```
change locations                                                Page   1 of   1
                                  LOCATIONS

                ARS Prefix 1 Required For 10-Digit NANP Calls? y

 Loc  Name           Timezone Rule  NPA                             Proxy Sel
 No                   Offset                                          Rte Pat
 1:  Main            + 00:00   0                                         20
```

Use the **change public-unknown-numbering 0** command, to define the calling party number to be sent to Voice Portal. Add an entry for the trunk group defined in **Section 3.6.2** to reach the Voice Portal application (see **Section 4.6**). In the example shown below, all calls originating from a **4-digit** extension beginning with **6** and routed to trunk group **20** will result in a **4-digit calling** number. The calling party number will be in the SIP "From" header. Submit these changes.

```
change public-unknown-numbering 0                               Page   1 of   2
                      NUMBERING - PUBLIC/UNKNOWN FORMAT
                                          Total
Ext Ext            Trk       CPN           CPN
Len Code           Grp(s)    Prefix        Len
                                                   Total Administered: 1
 4  2              20                       4         Maximum Entries: 9999
```

3.8 Administer Uniform Dial Plan and AAR Analysis

This section provides sample Automatic Alternate Routing (AAR) used for routing calls with dialed digits **6**xxx to Voice Portal. Note that other methods of routing may be used. Use the **change uniform-dialplan 0** command and add an entry to specify use of AAR for routing of digits 6xxx. Enter the following values for the specified fields, and retain the default values for the remaining fields. Submit these changes.

- **Matching Pattern:** Dialed prefix digits to match on, in this case **6**
- **Len:** Length of the full dialed number (i.e., **4**)
- **Del:** Number of digits to delete (i.e., **0**)
- **Net:** **aar**

```
change uniform-dialplan 0                                        Page   1 of   2
                       UNIFORM DIAL PLAN TABLE
                                                                 Percent Full: 0

  Matching                     Insert              Node
  Pattern          Len Del     Digits      Net Conv Num
 6                 4   0                  aar  n
```

Use the **change aar analysis 0** command and add an entry to specify how to route the calls to **3xxx** (PSTN through Session Manager and VG2000) and **6xxx** (for Voice Portal). Enter the following values for the specified fields and retain the default values for the remaining fields. Submit these changes.

Calls to PSTN:

- **Dialed String:** Dialed prefix digits to match on, in this case **3**
- **Total Min:** Minimum number of digits, in this case **4**
- **Total Max:** Maximum number of digits, in this case **4**
- **Route Pattern:** The route pattern number from **Section 3.7**. (i.e., **20**)
- **Call Type:** **aar**

Repeat with a second line for calls for Voice Portal:

- **Dialed String:** Dialed prefix digits to match on, in this case **6**
- **Total Min:** Minimum number of digits, in this case **4**
- **Total Max:** Maximum number of digits, in this case **4**
- **Route Pattern:** The route pattern number from **Section 3.7**. (i.e., **20**)
- **Call Type:** **aar**

```
change aar analysis 0                                           Page   1 of   2
                             AAR DIGIT ANALYSIS TABLE
                                 Location:  all          Percent Full:    1

      Dialed          Total     Route    Call   Node  ANI
      String         Min  Max  Pattern   Type   Num   Reqd

      3              4    4     20       aar
      6              4    4     20       aar          n
```

3.9 Save Translations

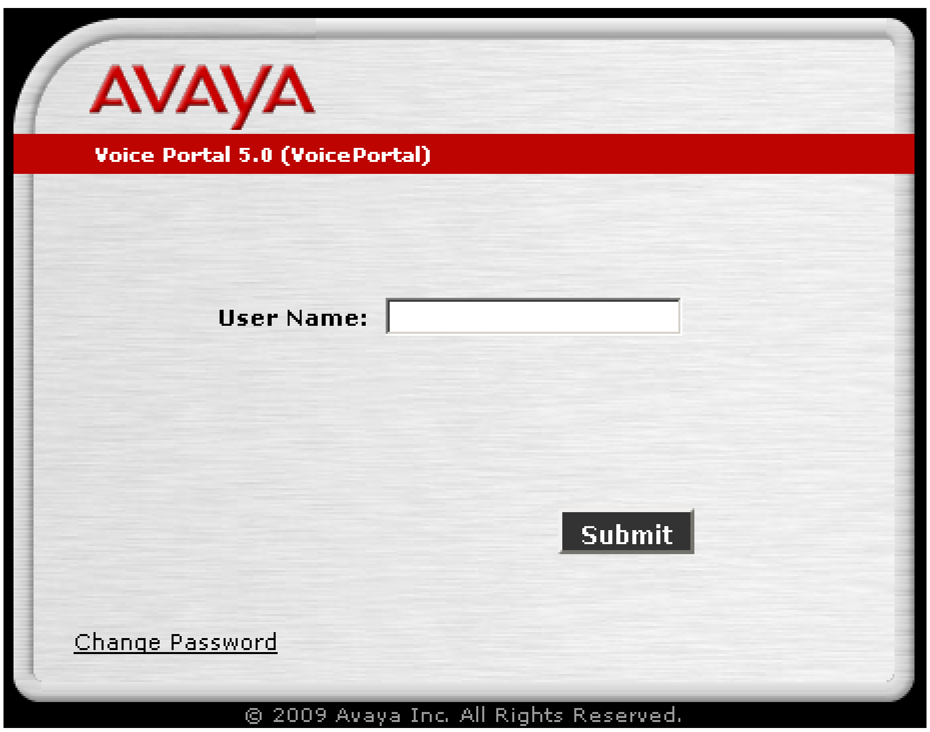
Configuration of Communication Manager is complete. Use the **save translations** command to save these changes.

4. Configure Avaya Aura™ Voice Portal

This section covers the administration of Voice Portal. Voice Portal solution is build on several components such as: Voice Portal Management System (VPMS), one or more Media Processing Platform (MPP) servers, one or more external speech servers. This section covers the administration of Voice Portal. The installation steps are defined in **Reference [7].** In this configuration, Voice Portal connected to the IP network via a SIP interface. Voice Portal configuration required:

- Configuring a SIP connection for Session Manager
- Adding MPP servers
- Configuring the VoIP audio format (mu-law or a-law)
- Adding a speech server
- Adding applications
- Starting the MPP servers

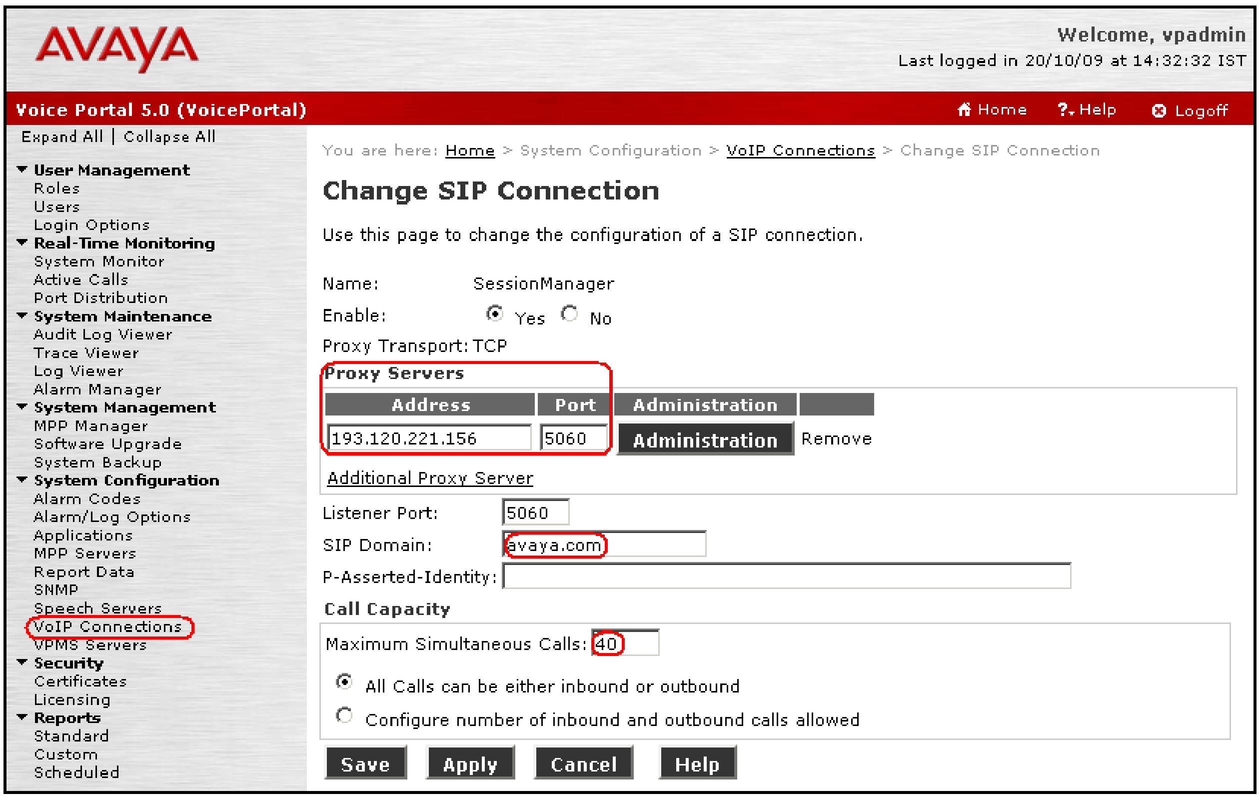
Voice Portal is configured via the Voice Portal Management System (VPMS) web interface. To access the web interface, enter http://<ip-addr>/VoicePortal as the URL in an Internet browser, where *<ip-addr>* is the IP address of the VPMS. The screen shown below is displayed. Log in using the Administrator user role.



4.1 Configuring a SIP Connection for Session Manager

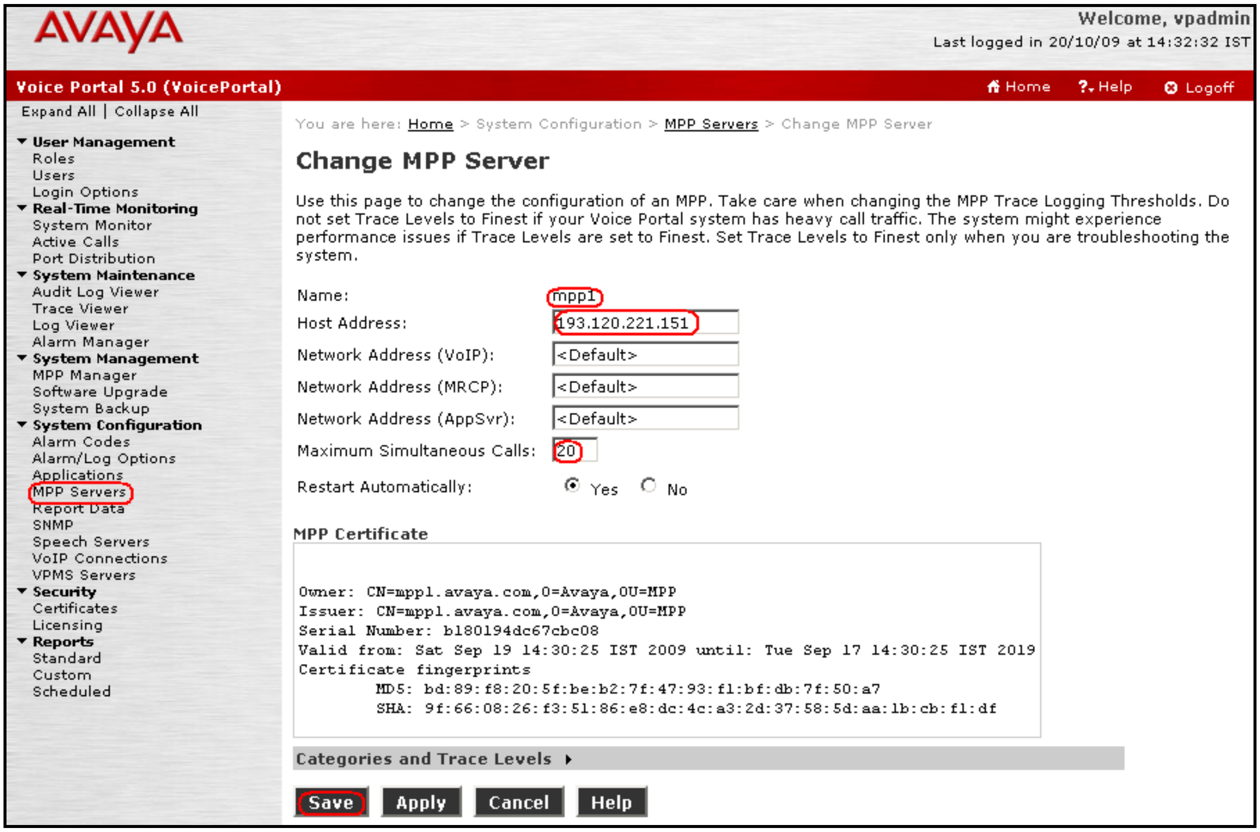
To configure a SIP connection for Session Manager, navigate to the **VoIP Connections** page and then click on the **SIP** tab. In the SIP tab shown in the **figure below**, the following parameters must be configured:

- Specify the IP address of Session Manager IP interface in the **Proxy Server Address** field
- Set the **Proxy Server Port** and **Listener Port** fields to **5060** for TCP.
- Set the **SIP Domain** (e.g., **avaya.com**)
- Set the **Maximum Simultaneous Calls** and **Number of Outbound Calls Allowed**. In this example, a maximum of **40** calls is supported between two MPP servers.
- Accept the default values for the other fields.

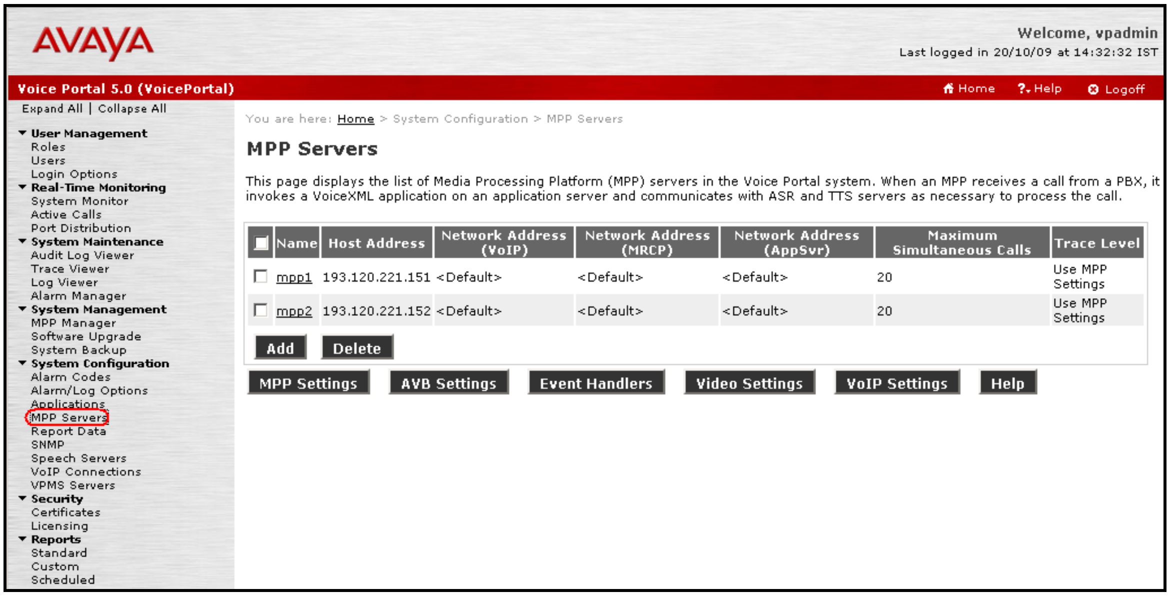


4.2 Add the MPP Servers

Add the required MPP servers (two in these Application Notes) by navigating to the **MPP Servers** screen. In the MPP Server configuration page, specify a descriptive name and the **Host Address** of each MPP server. Also, specify the **Maximum Simultaneous Calls** supported by each MPP server. The figure below shows the configuration for the first MPP server. Repeat these steps for the second MPP server.

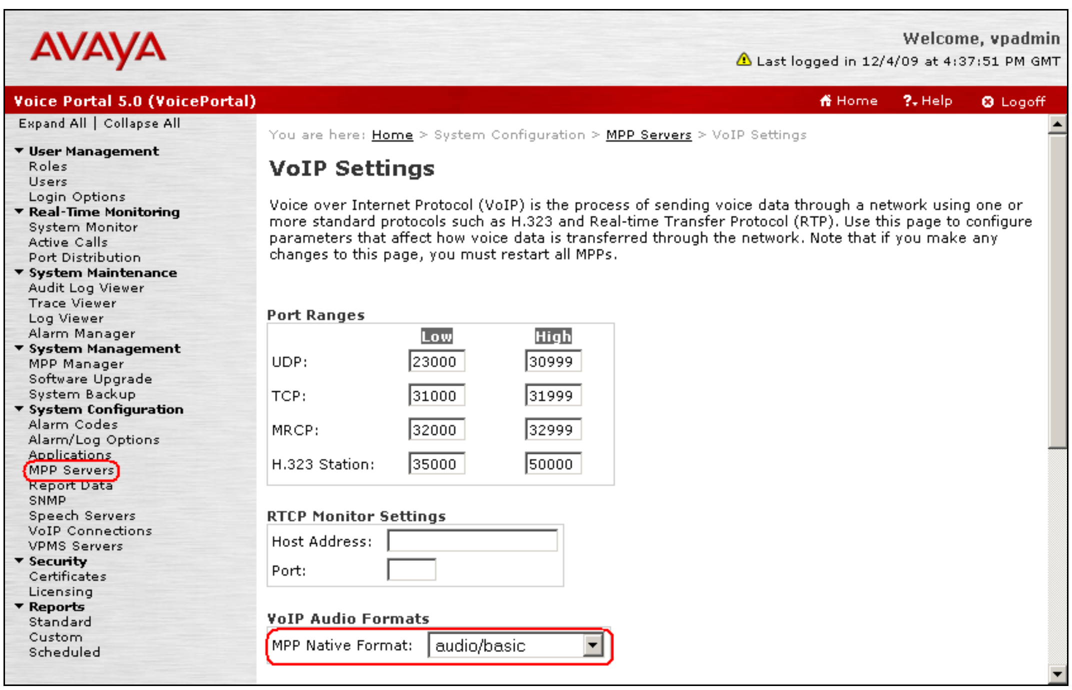


The following picture summarizes the process after the second MPP server is added into the system.



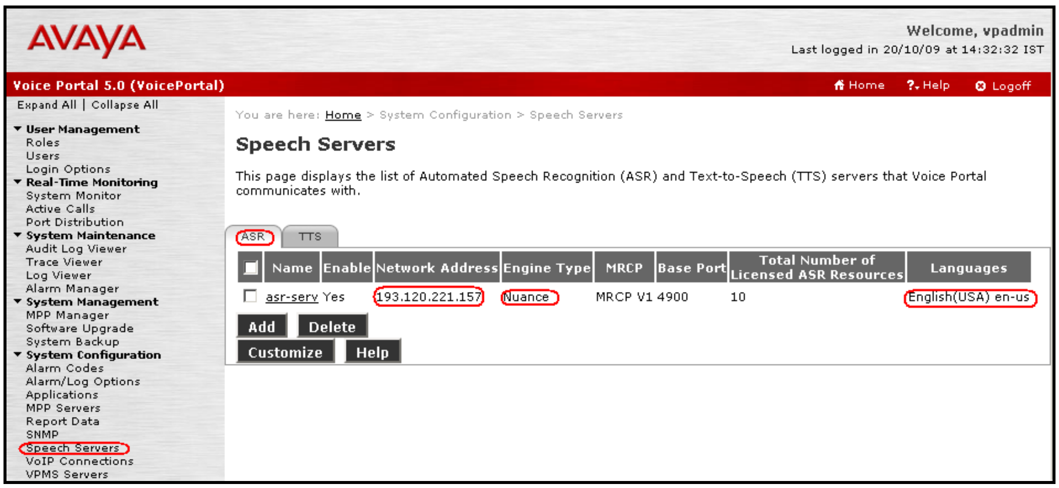
4.3 Configuring the VoIP Audio Format

The **VoIP Audio Format** for the MPP servers is configured in the **VoIP Settings** screen. The VG2000 supports both mu-law and a-law. The **MPP Native Format** field in the following figure is set to **audio/basic** for mu-law.



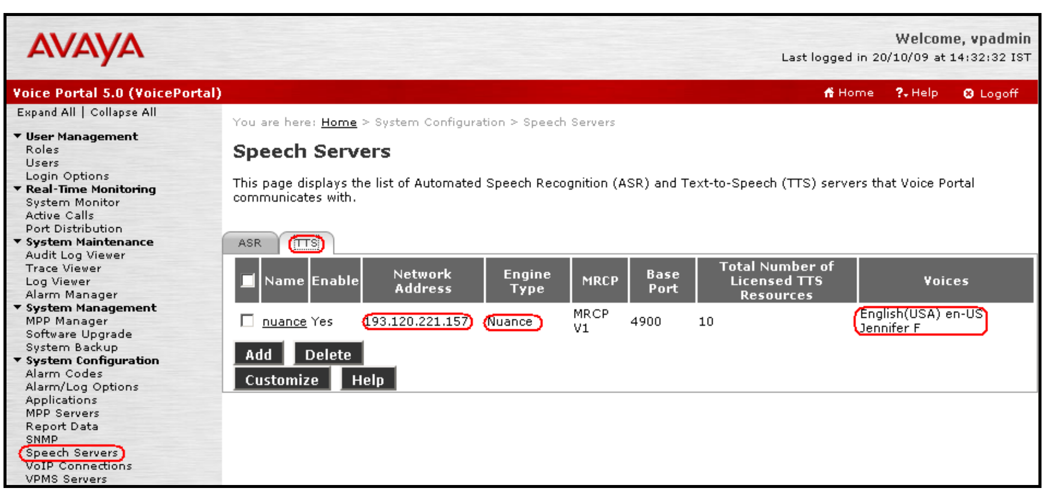
4.4 Add an ASR Server

To configure the ASR server, click on **Speech Servers** in the left pane, select the **ASR** tab, and then click **Add**. The following figure shows the screen after the ASR server has already been configured. Set the **Engine Type** to the appropriate value. In this configuration, a Nuance ASR server was used so the engine type was set to **Nuance.** Set the **Network Address** field to the IP address of the speech server and select the desired **Languages** to be supported. The other fields were set to their default values.



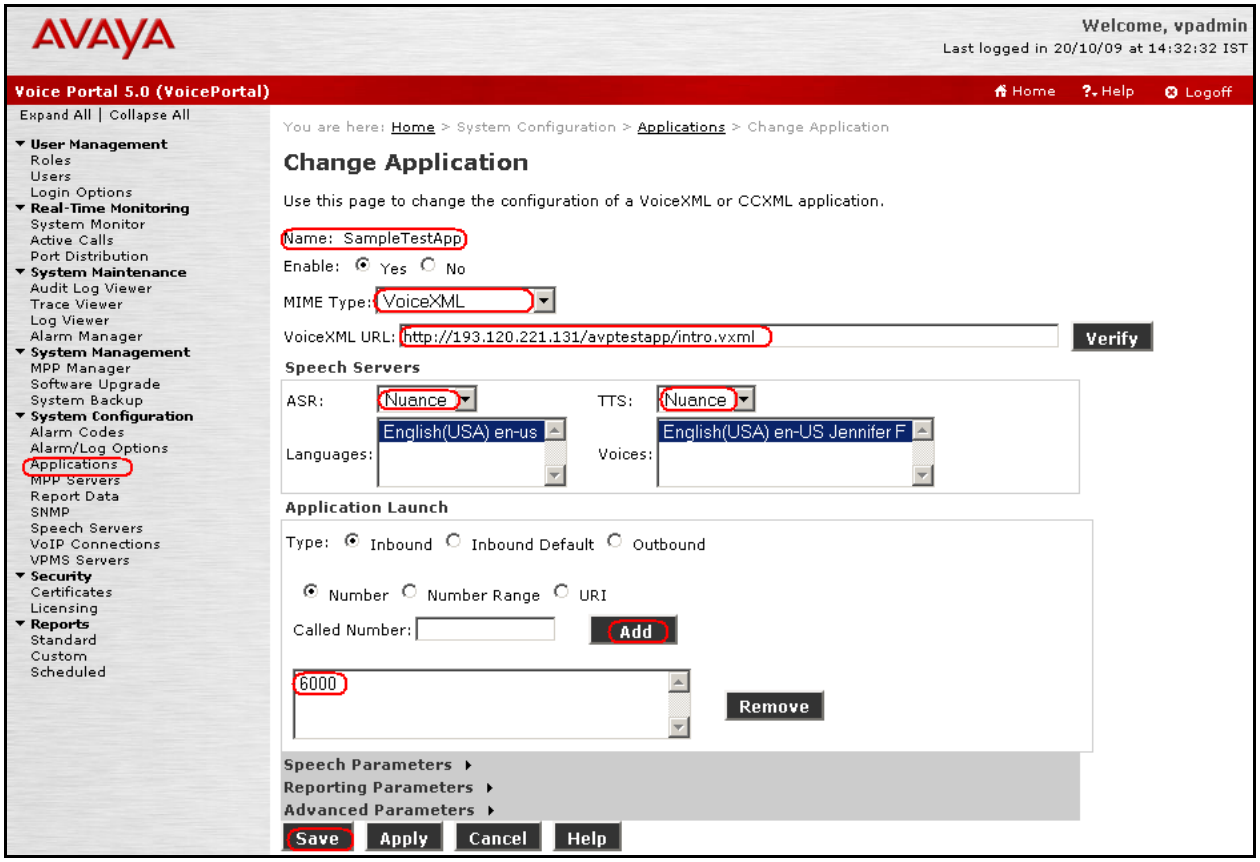
4.5 Add a TTS Server

To configure the TTS server, click on **Speech Servers** in the left pane, select the **TTS** tab, and then click **Add**. The following figure shows the screen after the TTS server has already been configured. Set the **Engine Type** to the appropriate value. In this configuration, a Nuance TTS server was used so the engine type was set to **Nuance**. Set the **Network Address** field to the IP address of the speech server and select the desired **Languages** to be supported. The other fields were set to their default values.



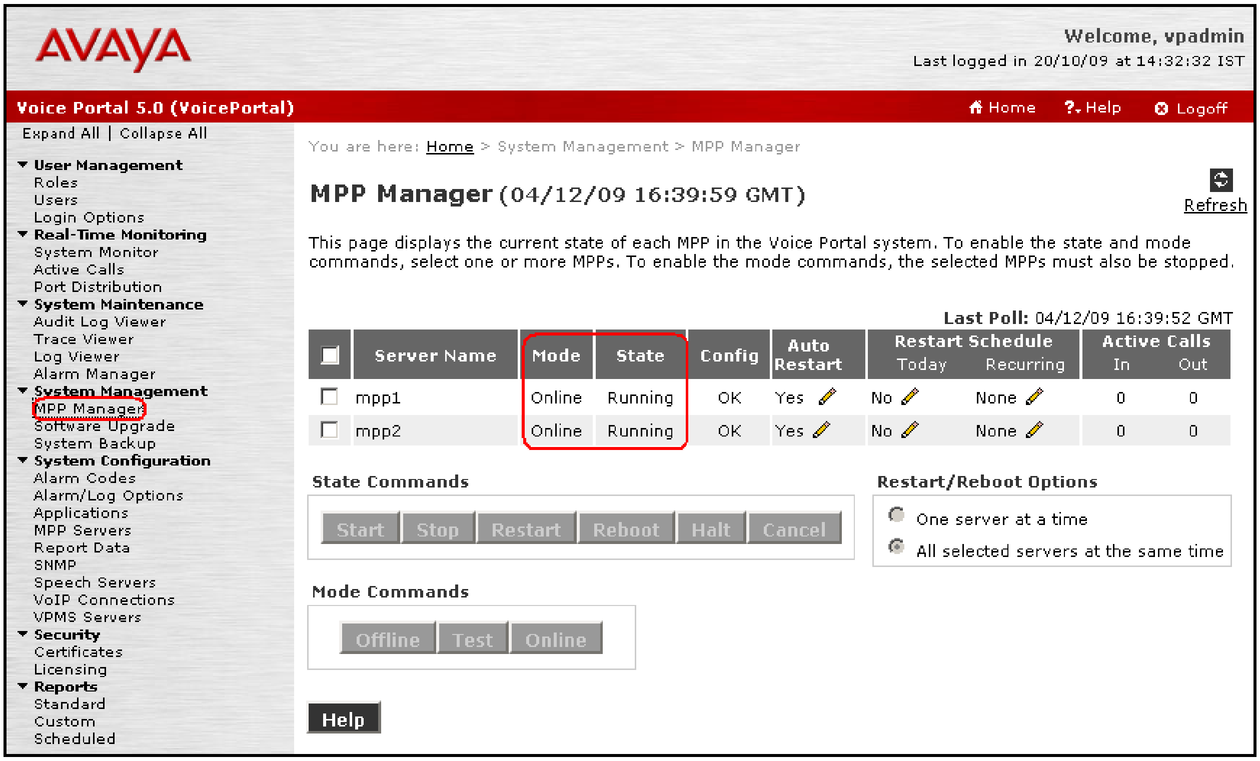
4.6 Add an Application

On the **Change Applications** page, add a Voice Portal application. Specify a **Name** for the application, set the **MIME Type** field to the appropriate value (e.g., **VoiceXML**), and set the **VoiceXML URL** field to point to a VoiceXML application on the application server. Next, specify the type of ASR and TTS servers to be used by the application and the called number that invokes the application (**Nuance** in these Application Notes). The called number is entered into the **Called Number** field and then the **Add** button is clicked. Click on Save once completed. The following screenshot summarizes the process.



4.7 Start the MPP Servers

Start the MPP servers from the **MPP Manager** page as shown in the figure below. Select each MPP and then click the **Start** button. The **Mode** of each MPP should be **Online** and the **State** should be **Running.**

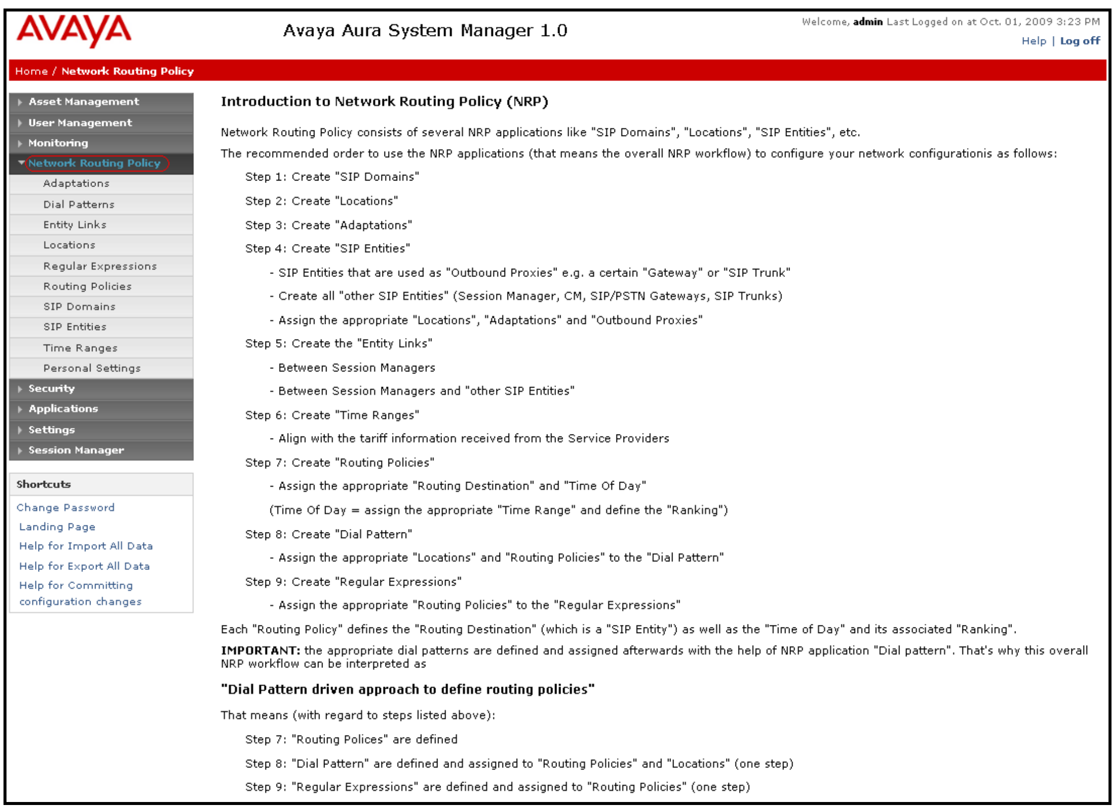


5. Configure Avaya Aura™ Session Manager

This section provides the procedures for configuring Session Manager, assuming it has been installed and licensed as described [5], [6], [7] and [8]. The procedures include adding the following items:

- Specify SIP Domain
- Add Adaptation
- Logical/physical Locations that can be occupied by SIP Entities
- SIP Entities corresponding to the SIP telephony systems and Session Manager
- Entity Links, which define the SIP trunk parameters used by Session Manager when routing calls to/from SIP Entities
- Routing Policies, which control call routing between the SIP Entities
- Dial Patterns, which govern to which SIP Entity a call is routed
- Session Manager, corresponding to the Session Manager Server to be managed by System Manager.

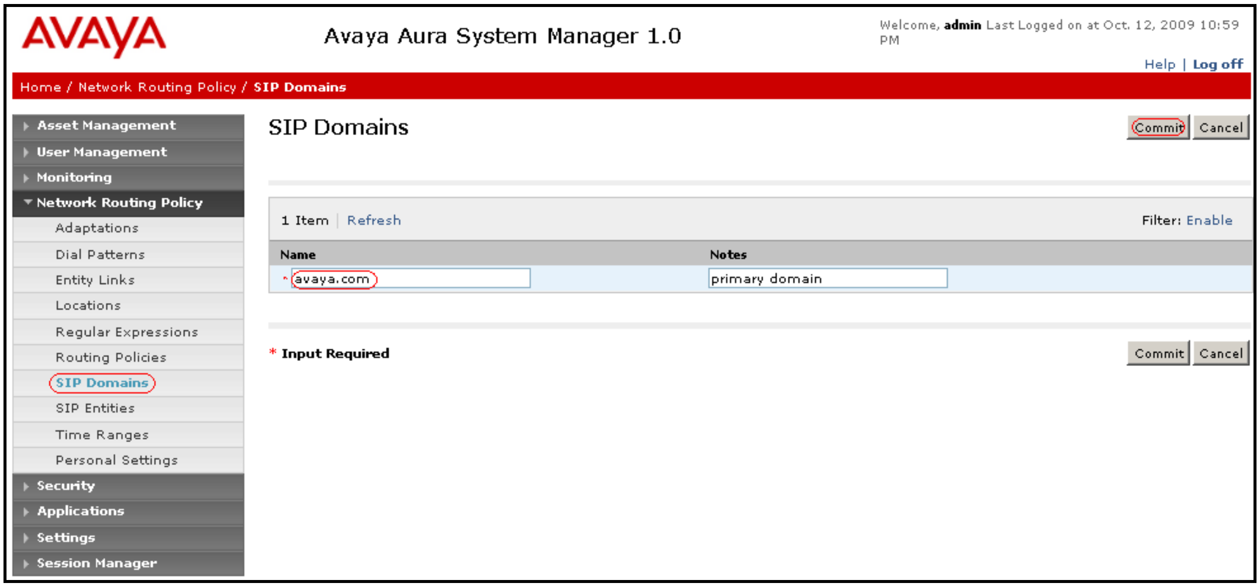
Configuration is accomplished by accessing the browser-based GUI of System Manager, using the URL **http://<ip-address>/IMSM**, where **<ip-address>** is the IP address of System Manager. Log in with the appropriate credentials and accept the Copyright Notice. The menu shown below is displayed. Expand the **Network Routing Policy** Link on the left side as shown.



5.1 Specify SIP Domain

Add the SIP domain for which the communications infrastructure will be authoritative. Do this by selecting **SIP Domains** on the left and clicking the **New** button on the right. The following screen will then be shown. Fill in the following fields and click **Commit**.

- **Name:** The authoritative domain name (e.g., **avaya.com**)
- **Notes:** Descriptive text (optional).



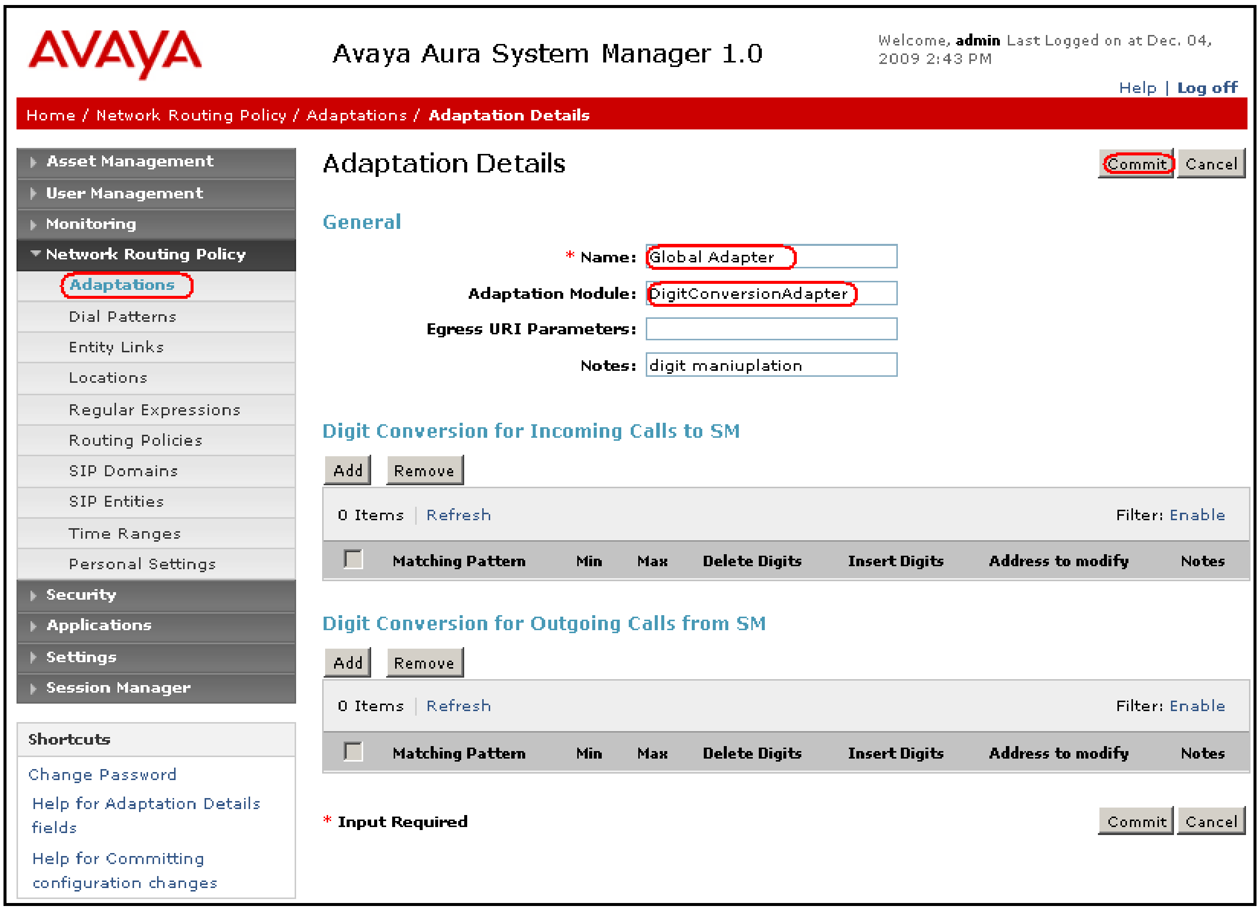
5.2 Add Adaptations

If required by the field configuration, digit manipulation can be done with adaptation module. To add an adaptation, under the Network Routing Policy select **Adaptations** on the left and click on the **New** button on the right. The following screen will then be shown. Fill in the following:

Under **General**:

- **Name:** A descriptive name.
- **Adaptation Module:** Enter the appropriate module name, refer to **[5]** for additional details.

The screen below illustrates the sample configuration. Click **Commit** to save the changes.



5.3 Add Locations

Locations can be used to identify logical and/or physical locations where SIP Entities reside, for purposes of bandwidth management. A single location is added to the configuration for Communication Manager, Voice Portal and VG2000. To add a location, select **Locations** on the left and click on the **New** button on the right. The following screen will then be shown. Fill in the following:

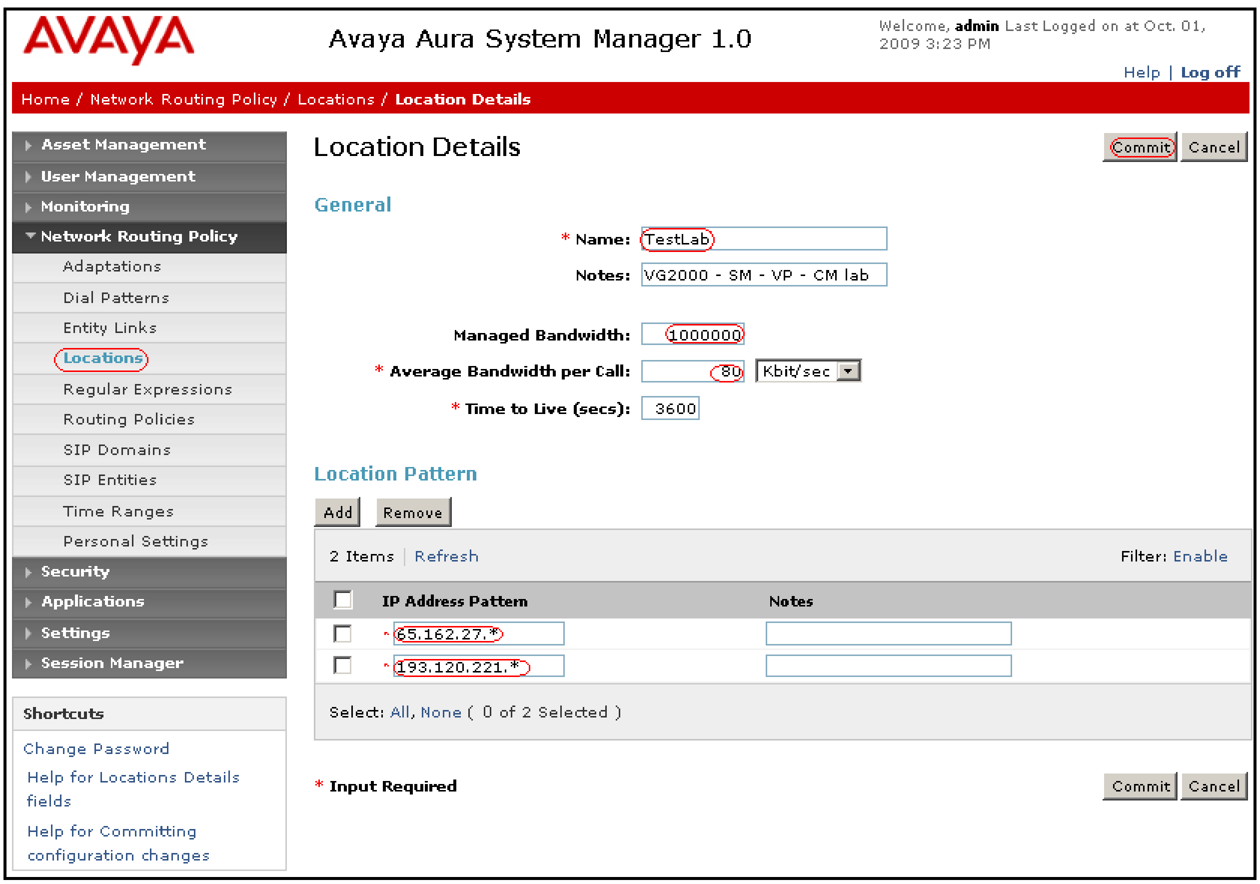
Under **General**:

- **Name:** A descriptive name.
- **Notes:** Descriptive text (optional).
- **Managed Bandwidth:** Leave the default or customize as described in **[5]**

Under **Location Pattern**:

- **IP Address Pattern:** A pattern used to logically identify the location. In these Application Notes, the pattern selected defined the networks involved. Other patterns can be used.
- **Notes:** Descriptive text (optional).

The screen below shows addition of the **TestLab** location, which includes all the components of the compliance test lab. Click **Commit** to save.

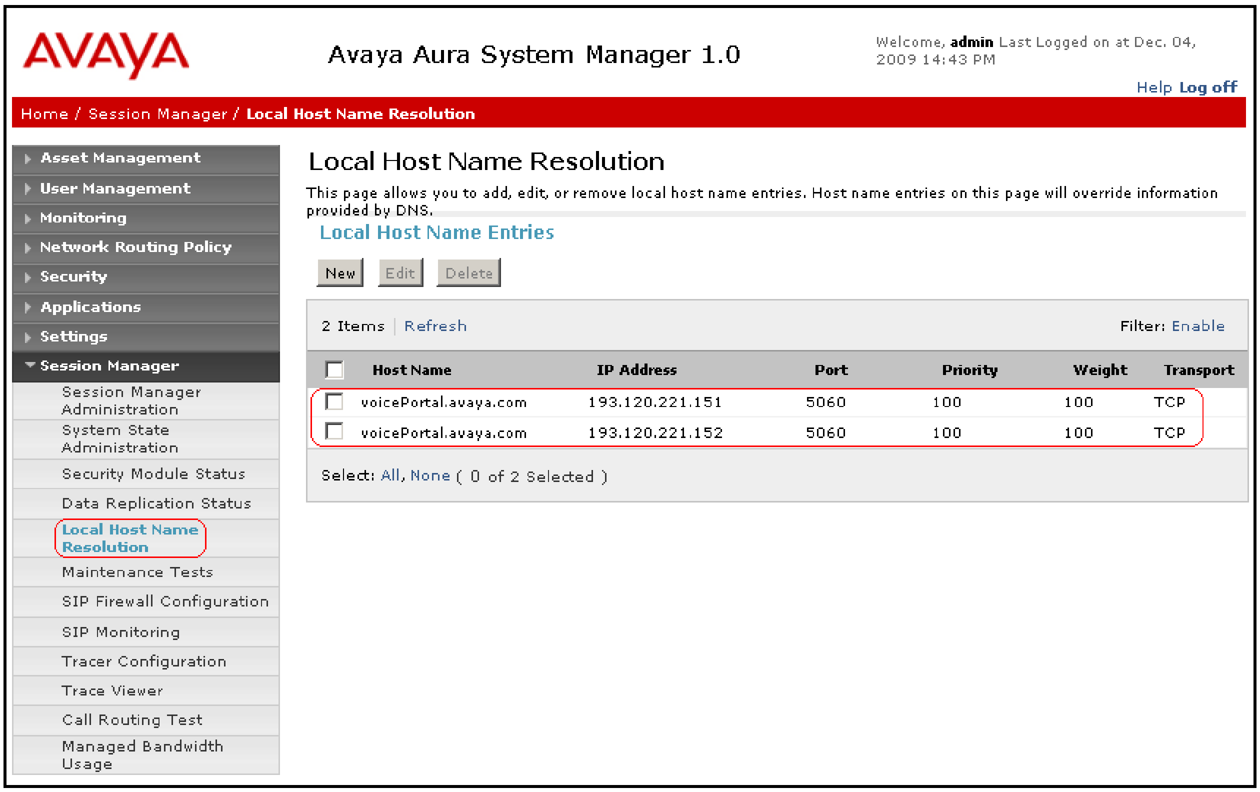


5.4 Add Voice Portal MPPs as Local Host Entries

Session Manager can connect calls to an entity with multiple IP interfaces to perform failover/load sharing. In order to configure load sharing among different MPP servers, a Host Name must be defined as it follows. Expand **Session Manager** in the web interface

- Select **Local Host Name Resolution** to enter the details of the MPPs with their respective IP addresses.
  - Click **New**
  - Under **Host Name,** add an identifier followed by the SIP domain (added in **Section 4.1**). This Host Name is going to be common for the 2 of MPPs being added.
  - Under **IP Address,** enter the IP address of the respective MPPs.
  - Under **Port,** enter **5060**.
  - Enter the appropriate **Priority** and **Weight** as required.
  - Under **Transport,** select **TCP**.
  - Click **Commit** to save.

Shown below is the updated screen for the sample configuration.



5.5 Add SIP Entities

A SIP Entity must be added for Session Manager and for each SIP-based telephony system supported by it using SIP trunks. In the sample configuration, a SIP Entity is added for the Session Manager, the C-LAN board in the Avaya G650 Media Gateway, the Voice Portal and VG2000. To add a SIP Entity, select **SIP Entities** on the left and click on the **New** button on the right. The following screen is displayed. Fill in the following:

Under **General**:

- **Name:** A descriptive name.
- **FQDN or IP Address:** IP address of the Session Manager or the signaling interface on the telephony system.
- **Type:** Select between **Session Manager** for Session Manager, **CM** for Communication Manager, and **VoicePortal** for VoicePortal.

Gateway for VG2000

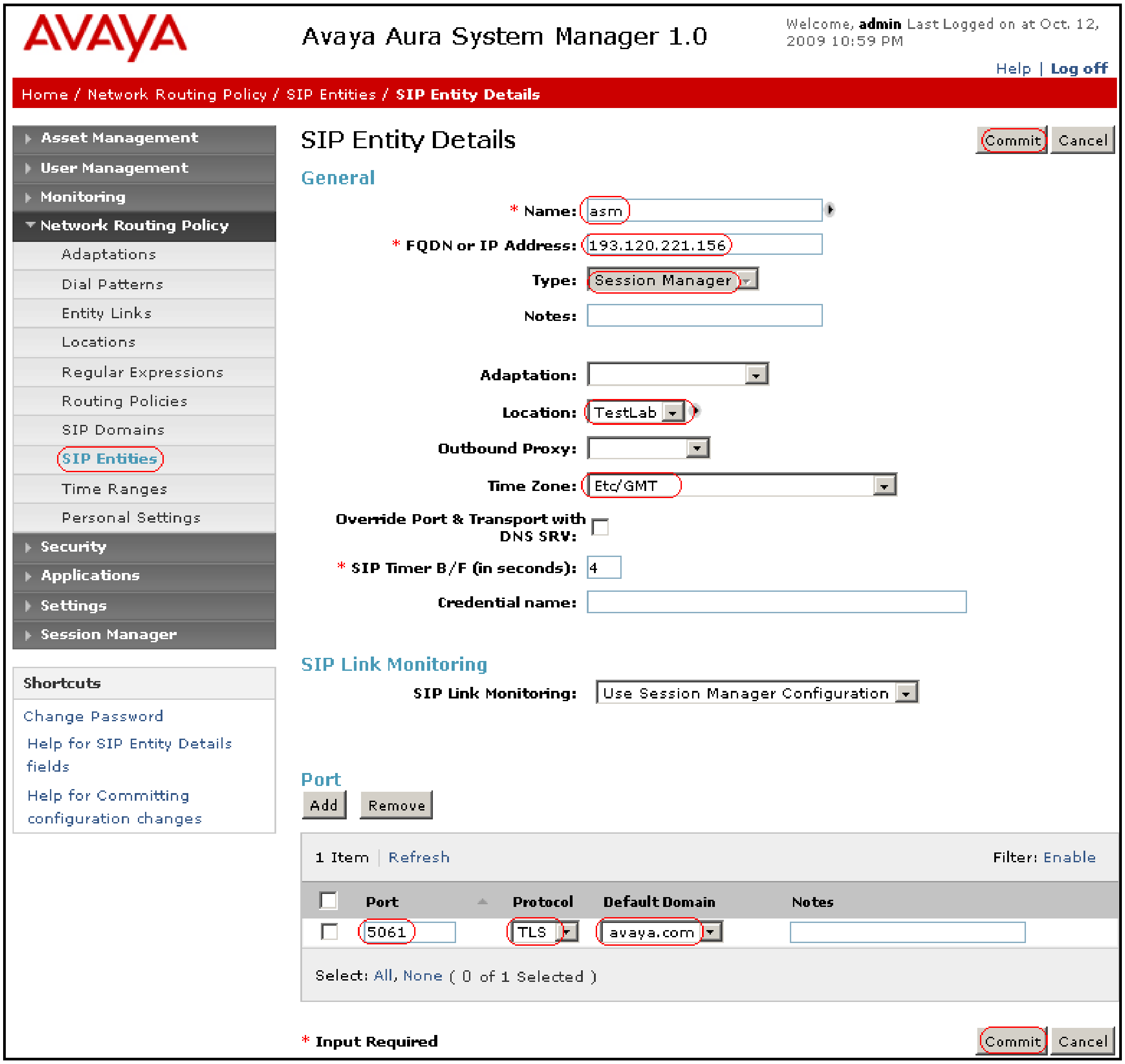
- **Adaptation:** Select the previously created Adaptation if needed.
- **Location:** Select one of the locations defined previously.
- **Time Zone:** Time zone for this entity.

Under **Port**, click **Add**, and then edit the fields in the resulting new row as shown below:

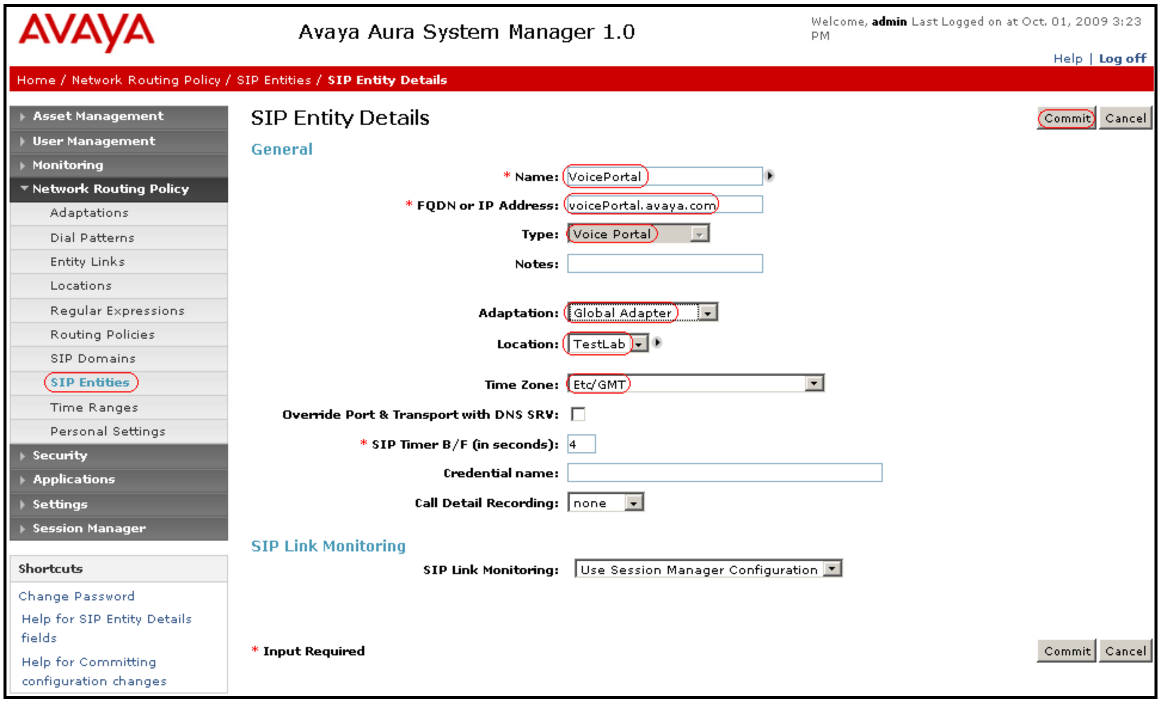
- **Port:** Port number on which the system listens for SIP requests.
- **Protocol:** Transport protocol to be used to send SIP requests.
- **Default Domain** The domain used for the enterprise (e.g., **avaya.com**).

Defaults can be used for the remaining fields. Click **Commit** to save each SIP Entity definition.

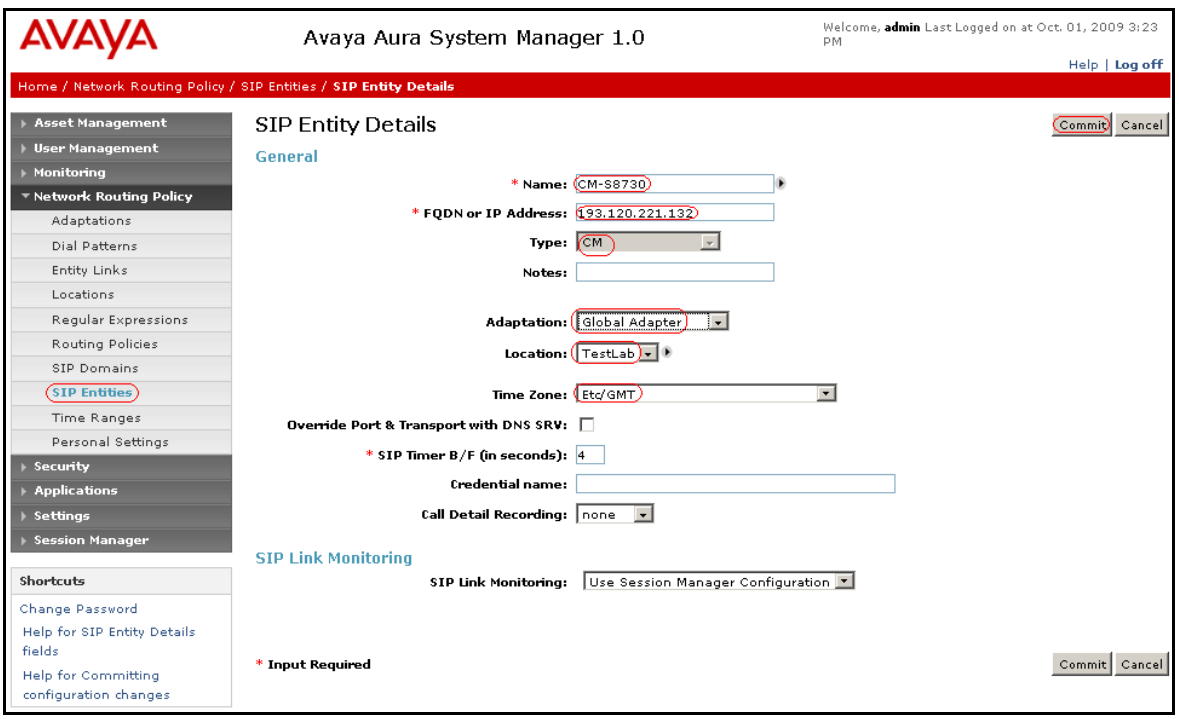
The following screen shows the addition of Session Manager. The IP address used, **193.120.221.156,** is the SM-100 Security Module.



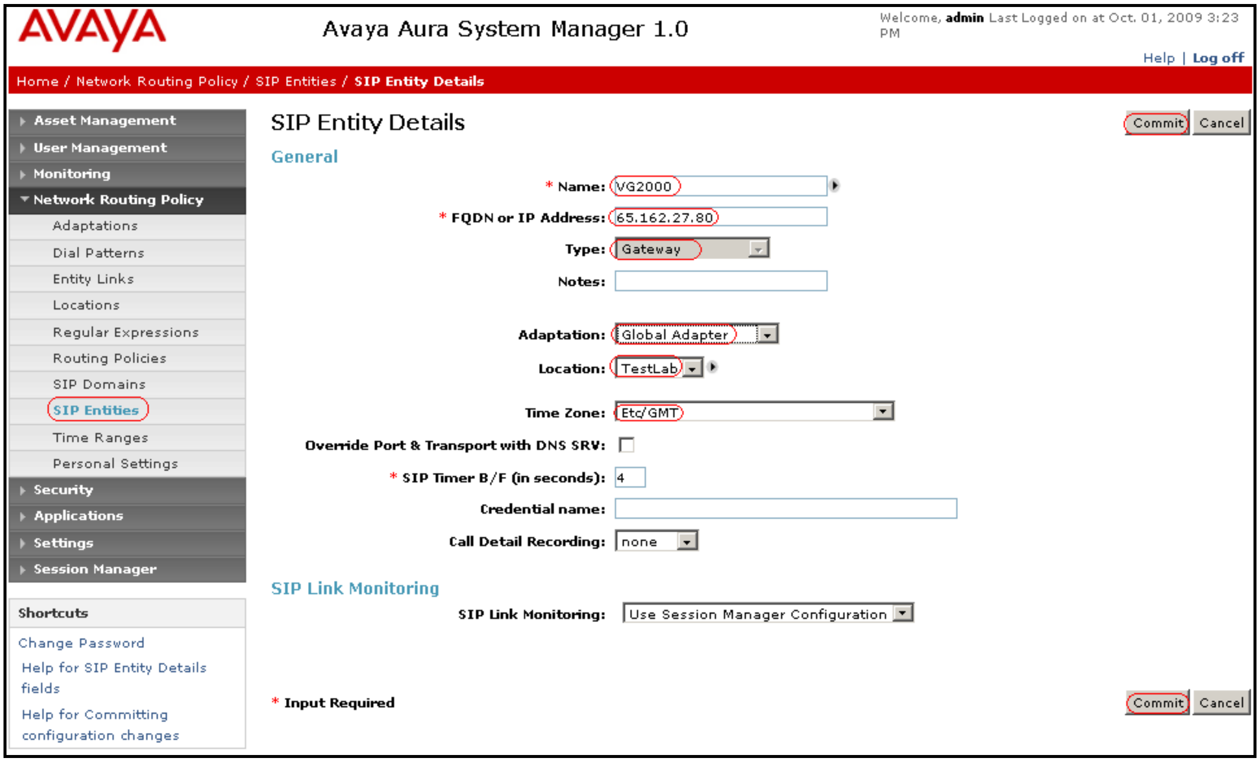
The following screen shows the addition of Voice Portal, as **FQDN voicePortal.avaya.com** is used. This FQDN was defined on the Local Host Name Resolution screen shown in **Section 5.4.**



The following screen shows addition of Communication Manager. The IP address for defining the SIP Entity used is that of the C-LAN board in the Avaya G650 Media gateway, **193.120.221.132** in this test configuration.



The picture below shows the configuration of the SIP Entity related to the NMS VG2000, the **IP Address**, **65.162.27.80** is the signaling interface of the VG2000, refer to **Section 5.4**.



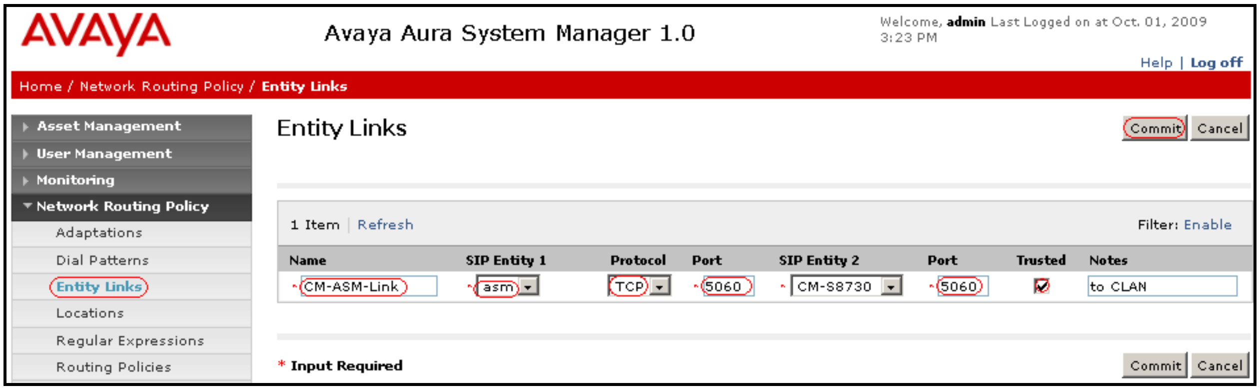
5.6 Add Entity Links

A SIP trunk between Session Manager and a telephony system is described by an Entity link. To add an Entity Link, select **Entity Links** on the left and click on the **New** button on the right. Fill in the following fields in the new row that is displayed:

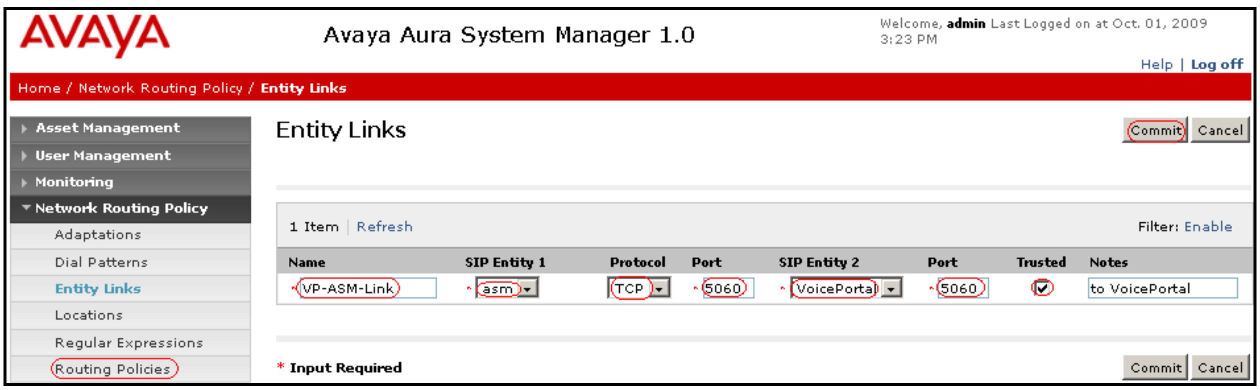
- **Name:** A descriptive name.
- **SIP Entity 1:** Select the Session Manager entity.
- **Port:** Port number to which the other system sends SIP requests
- **SIP Entity 2:** Select the name of the other system.
- **Port:** Port number on which the other system receives SIP requests
- **Trusted:** Check this box.
  Note: If this box is not checked, calls from the associated SIP Entity specified in **Section 4.4** will be denied.
- **Protocol:** Select the transport protocol between **UDP/TCP/TLS** as **long they** are aligned with the definition on the **other end o**f the link. In these application notes **TCP** was used.

Click **Commit** to save each Entity Link definition**.**

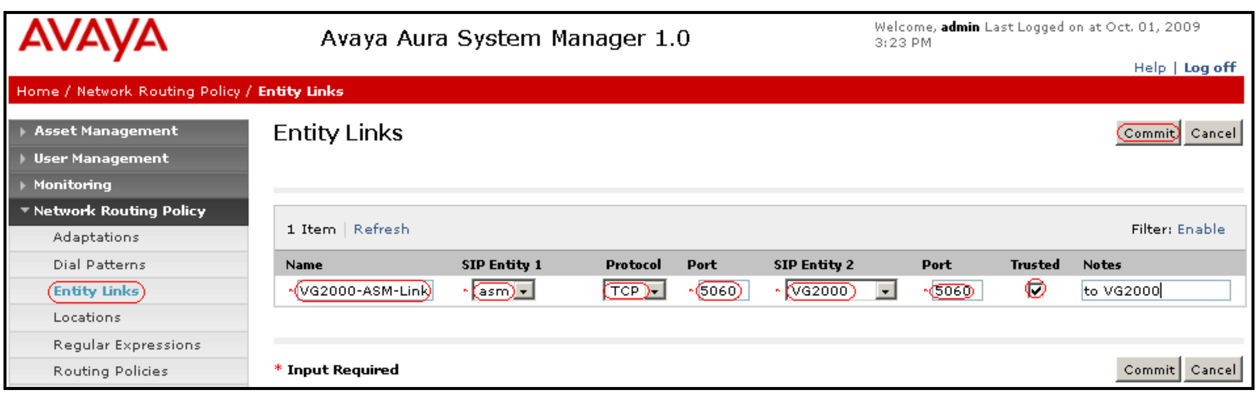
The following screens illustrate adding the Entity Link for Communication Manager.



Below it's illustrated adding the Entity Link for Voice Portal.



Below it's illustrated adding the Entity Link for VG2000.



5.7 Add Routing Policies

Routing policies describe the conditions under which calls will be routed to the SIP Entities specified in **Section 4.4**. Three routing policies must be added: for Communication Manager, Voice Portal and one for NMS VG2000. To add a routing policy, select **Routing Policies** on the left and click on the **New** button on the right. The following screen is displayed. Fill in the following:

Under **General:**

- Enter a descriptive name in **Name**.

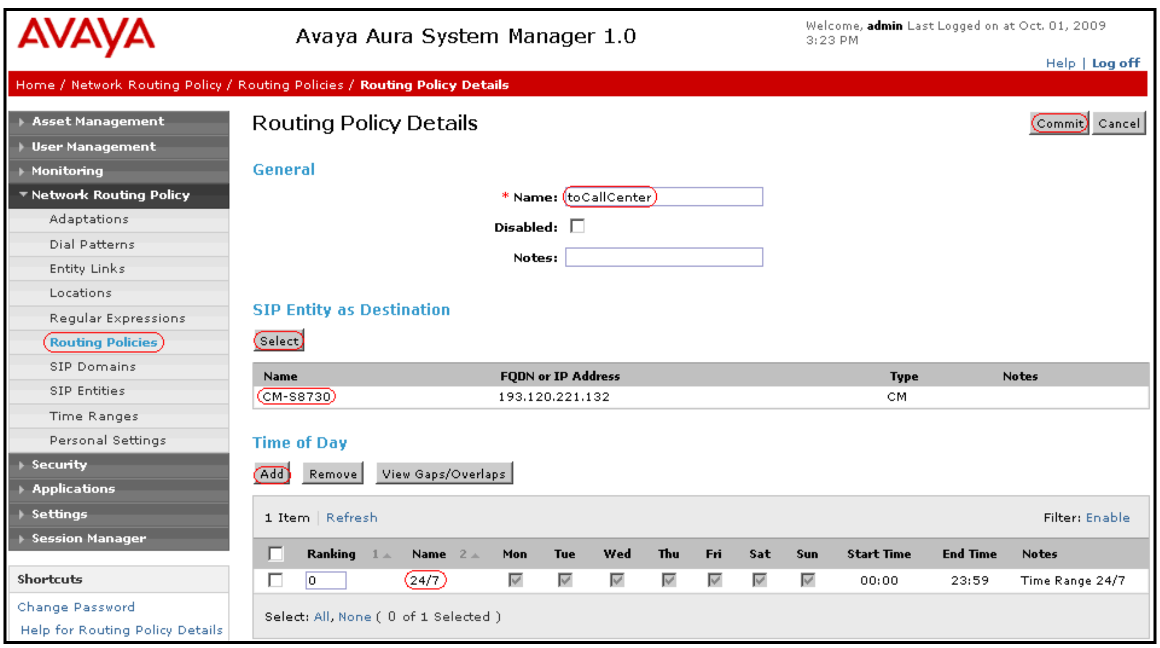
Under **SIP Entity as Destination:**

- Click **Select**, and then select the appropriate SIP entity to which this routing policy applies.

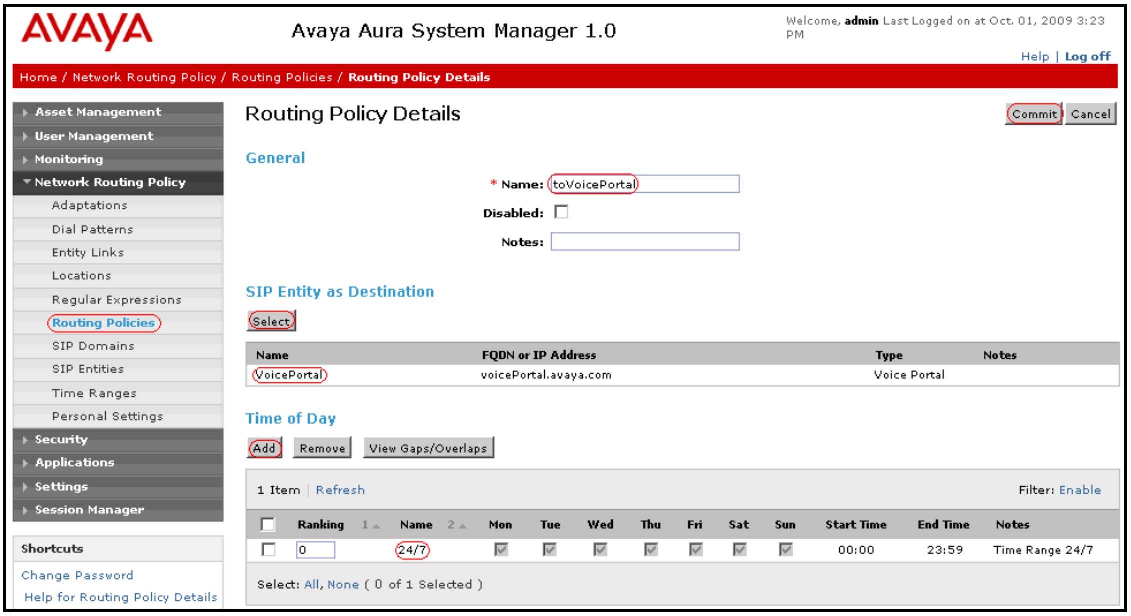
Under **Time of Day:**

- Click **Add**, and select the time range configured. In these Application Notes the predefined 24/7 Time Range is used.

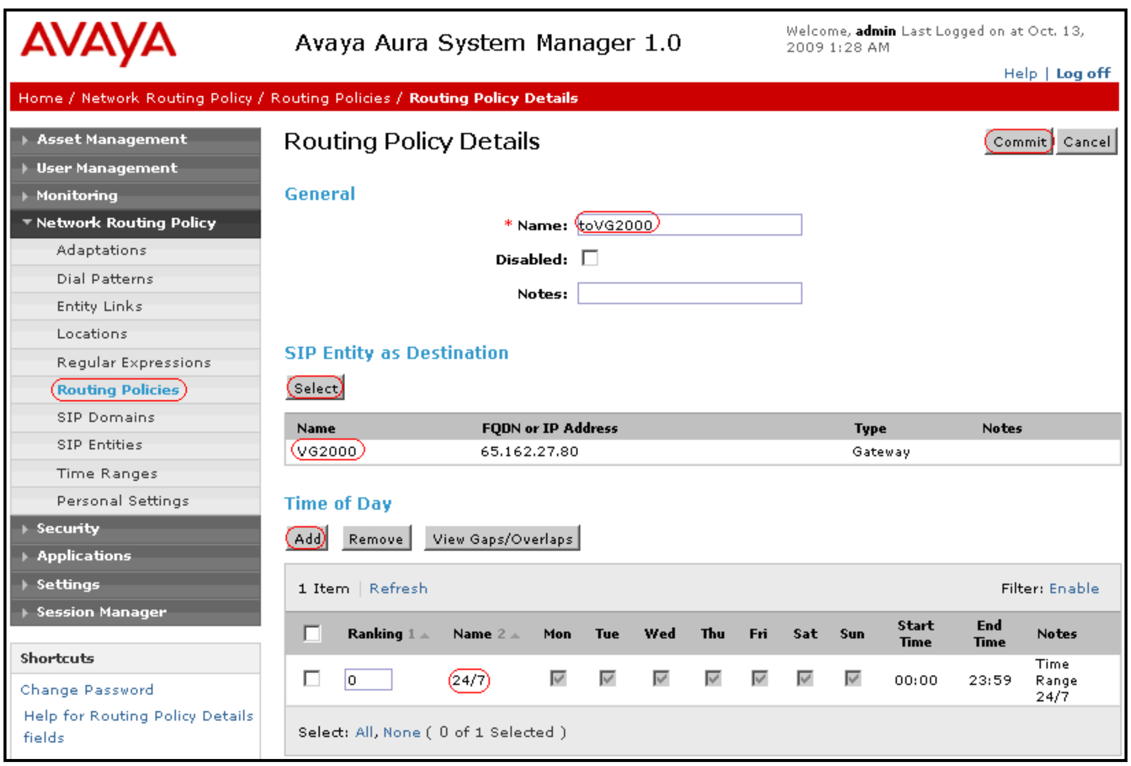
Defaults can be used for the remaining fields. Click **Commit** to save each Routing Policy definition. The following picture shows the Routing Policy for Communication Manager.



The following picture shows the Routing Policy for Voice Portal.



The following picture shows the Routing Policy for NMS VG2000 Gateway.



5.8 Add Dial Patterns

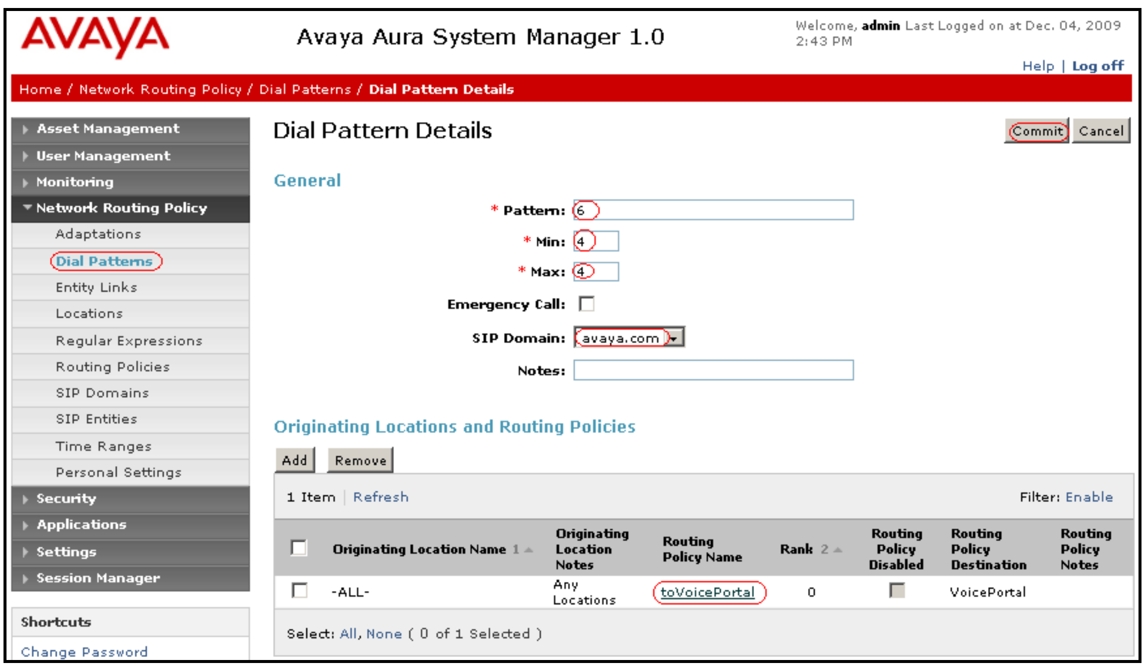
Dial patterns must be defined that will direct calls to the appropriate SIP Entity. In the sample configuration, 4-digit extensions beginning with **2** reside on Communication Manager and 4-digit DDI beginning with **6** reside on Voice Portal, while numbers beginning with 3 resides on PSTN therefore are associated with VG2000 gateway. To add a dial pattern, select **Dial Patterns** on the left and click on the **New** button on the right. Fill in the following, as shown in the screen below, which corresponds to the dial pattern for routing calls to Communication Manager:

Under **General**:

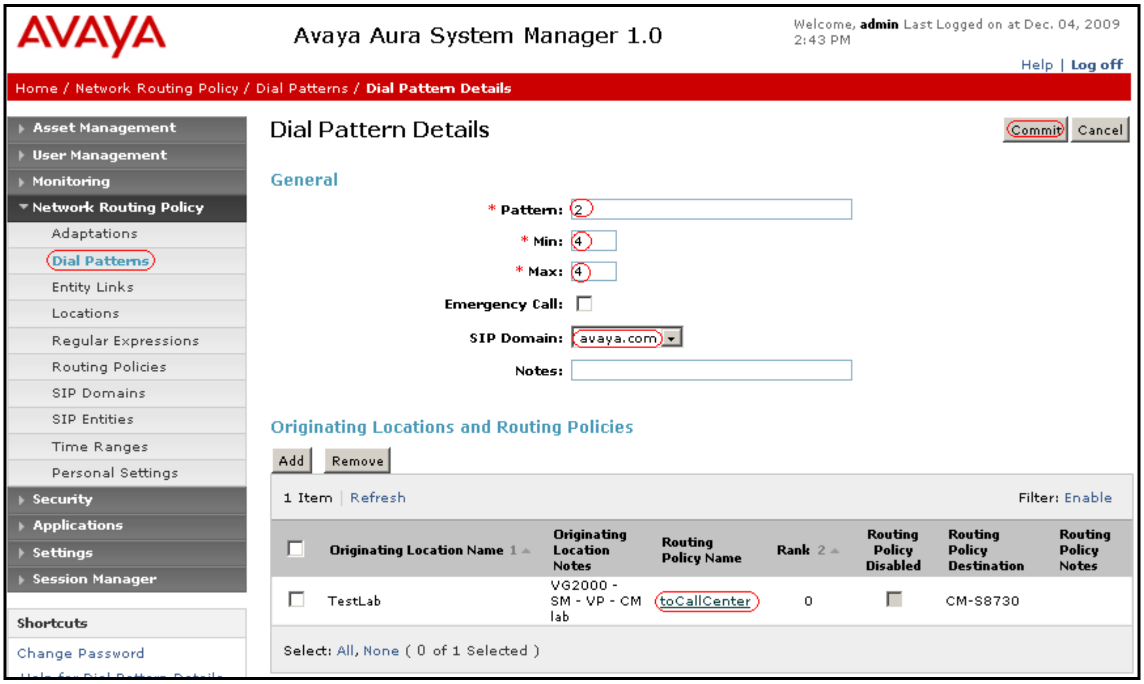
- **Pattern:** Dialed number or prefix.
- **Min** Minimum length of dialed number.
- **Max** Maximum length of dialed number.
- **Notes** Comment on purpose of dial pattern.

Under **Originating Locations and Routing Policies**, click **Add**, and then select the appropriate location and routing policy from the list. Default values can be used for the remaining fields. Click **Commit** to save this dial pattern.

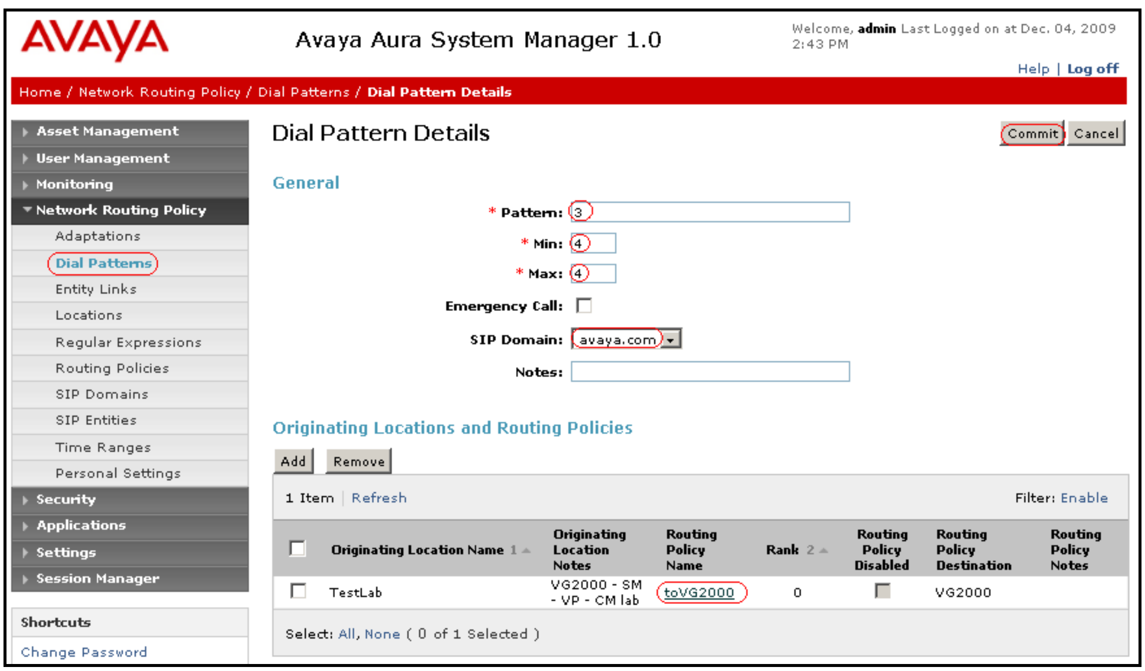
The following screen shows the dial pattern definitions for Voice Portal.



The following screen shows the dial pattern definitions for Communication Manager.



The following screen shows the dial pattern definitions for NMS VG2000 Gateway



5.9 Add Session Manager

To complete the configuration, adding the Session Manager will provide the linkage between System Manager and Session Manager. Expand the **Session Manager** menu on the left and select **Session Manager Administration**. Then click **Add**, and fill in the fields as described below and shown in the following screen:

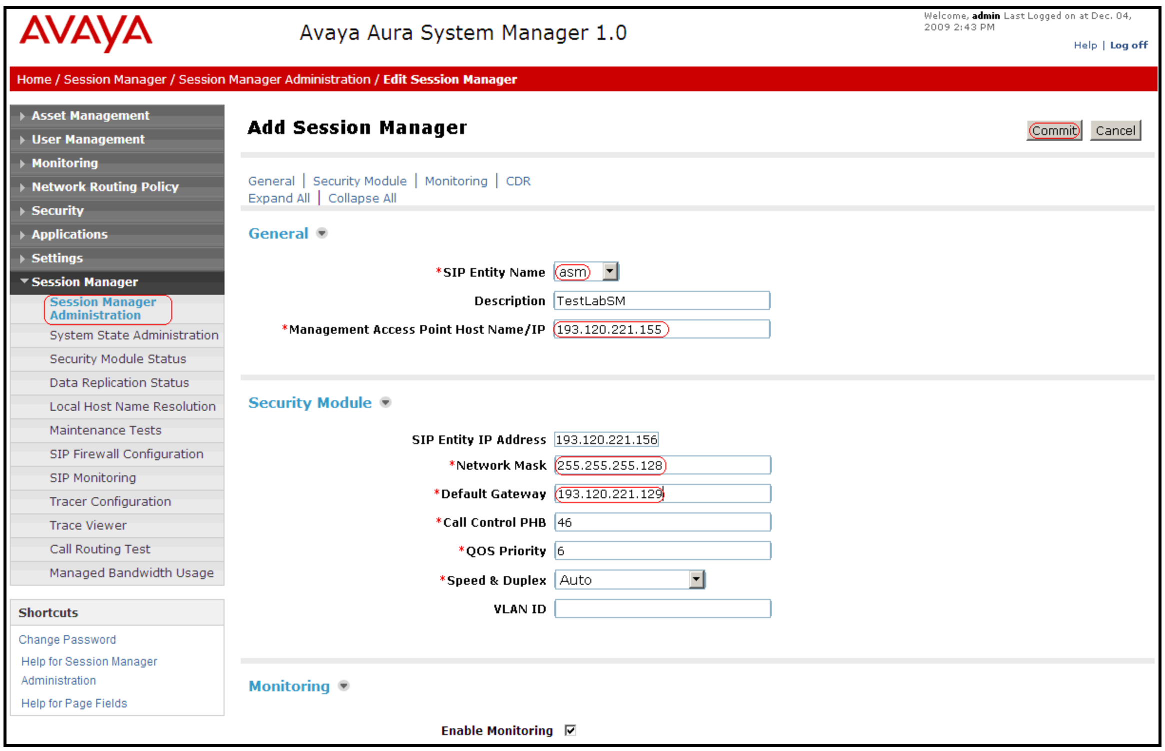
Under **General**:

- **SIP Entity Name:** Select the name of the SIP Entity added for Session Manager.
- **Description**: Descriptive comment (optional)
- **Management Access Point Host Name/IP**
  Enter the IP address of the Session Manager management interface.

Under **Security Module**:

- **Network Mask:** Enter the network mask corresponding to the IP address of the SM100 interface (i.e., **255.255.255.128**)
- **Default Gateway**: Enter the IP address of the default gateway for SM100 interface (i.e. **193.120.221.129**)

Use default values for the remaining fields. Click **Save** to add this Session Manager.



6. NMS VG2000 Configuration

This section provides the procedures for configuring the Vision Media Gateway (VG2000). The procedures require two distinct operations:

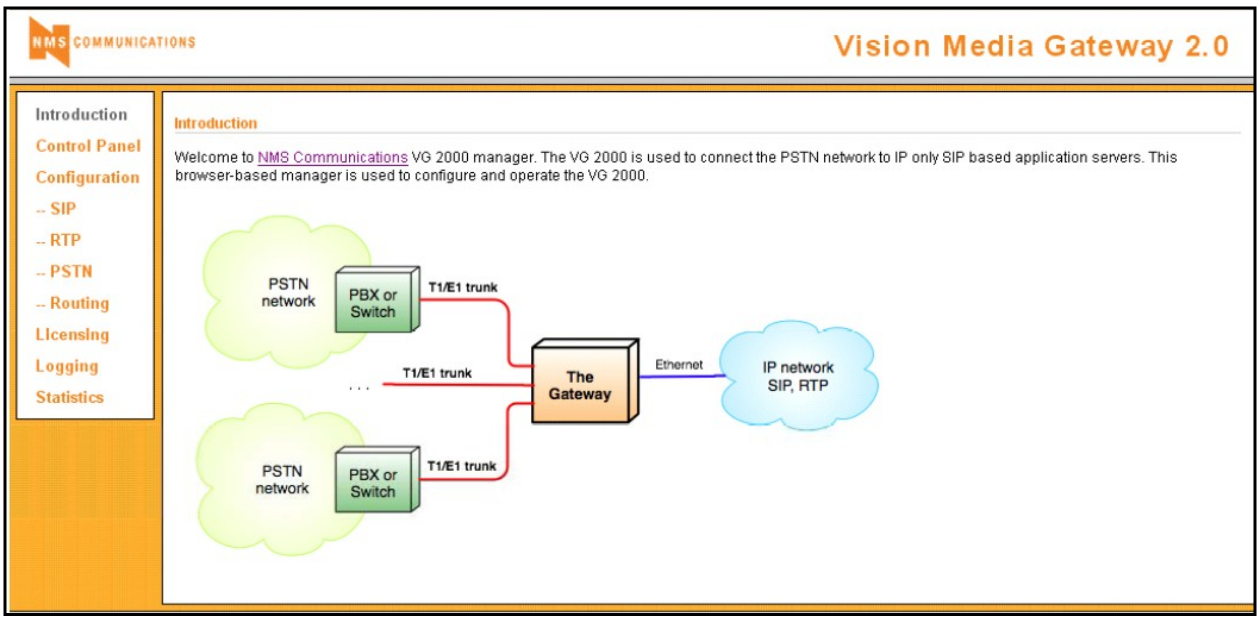
- Configuring the Vision Media Gateway host IP network parameters
- Configuring the Vision Media Gateway telephony network interfaces

6.1 Configure the IP Network Parameters

The IP network parameters of the Vision Media Gateway are configured using standard Linux administration tools. Refer to **[8]** and **[9]** for instructions.

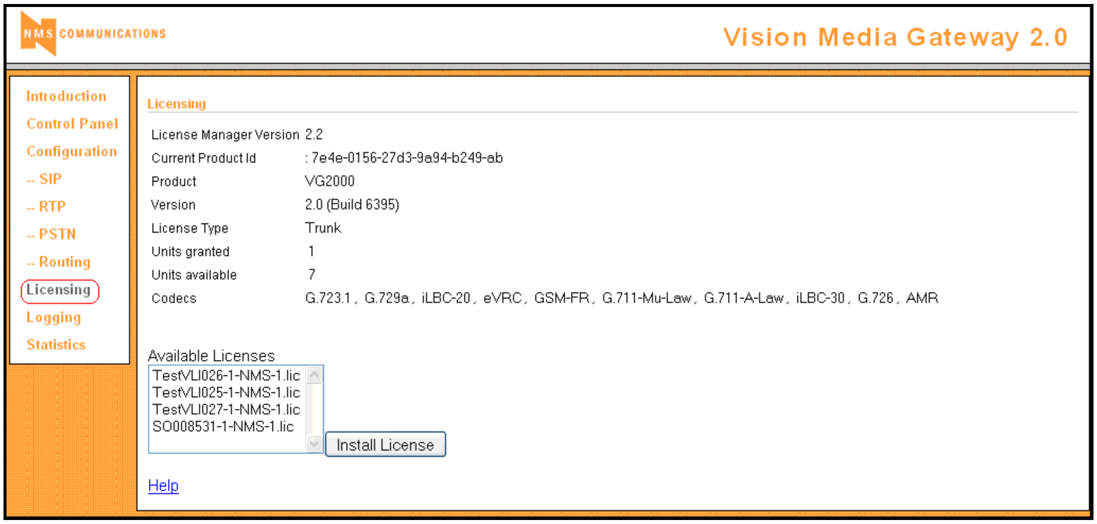
6.2 Configure Telephony Network Interfaces

To configure the telephony network interfaces of the Vision Media Gateway, access the Configuration Manager by launching Mozilla Firefox browser version 2.0 or later. Browse to the IP address of the Vision Media Gateway using port 8080 by specifying **http://<ip-address>:8080** in the URL. Log in using administrator credentials. The Vision Media Gateway welcome screen is shown below.



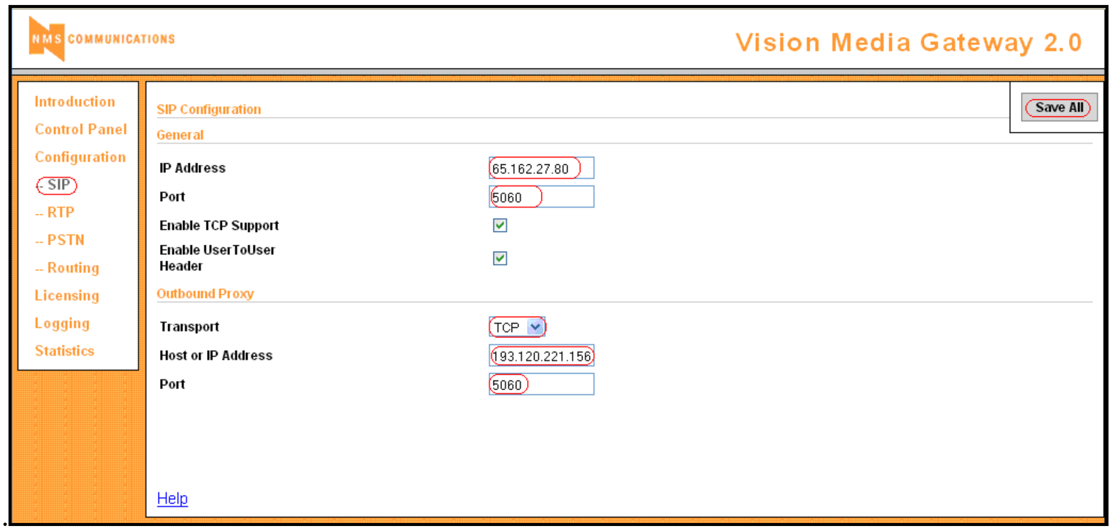
6.2.1 Verify licensing

The top portion of the Licensing window displays current licensing information. The bottom portion of the window shows the available licenses and allows you to install additional licenses. **Note:** G.729a codec require a license. Contact an NMS Communications representative for more information.



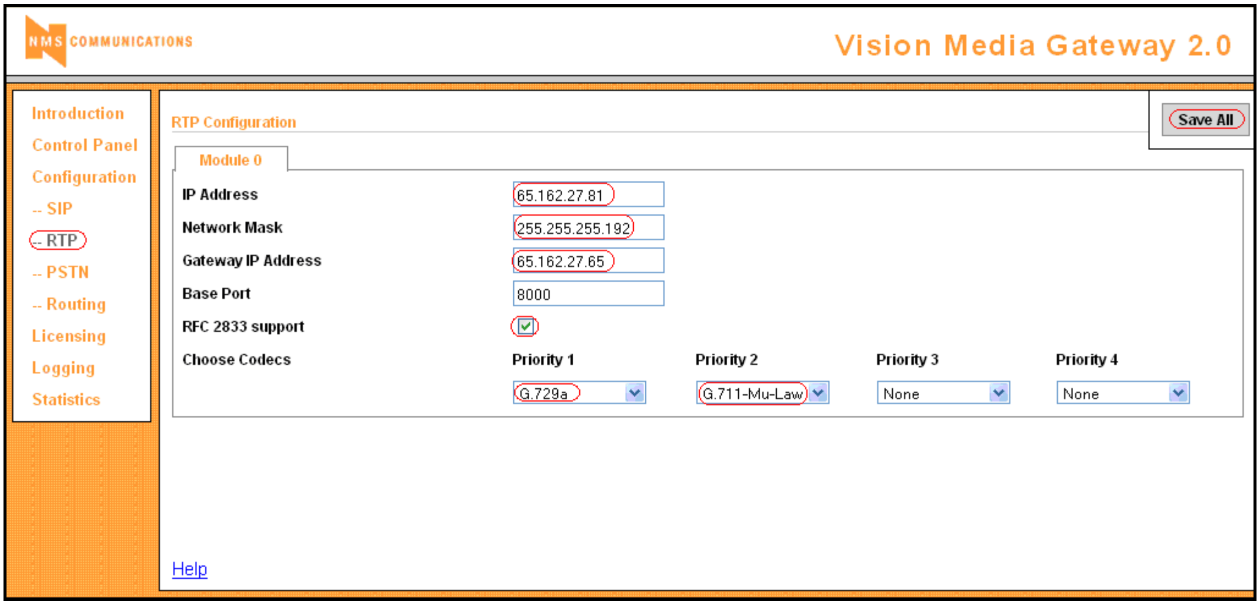
6.2.2 Configure the SIP Interface.

SIP messages are exchanged between the VG2000 and Avaya Session Manager using the TCP transport protocol and **Port 5060**. Navigate to the **SIP Configuration screen** and configure the parameters as shown in the figure below. Under the **General** section, specify the IP address of the VG2000 in the **IP Address** field, set the **Port** field to **5060**, and enable **TCP** support. Under the **Outbound Proxy** section, set the **Transport** field to **TCP**, configure the **Host or IP Address** field to the IP address of SM-100 interface on the SES, and set the **Port** field to **5060**. Click **Save All**.



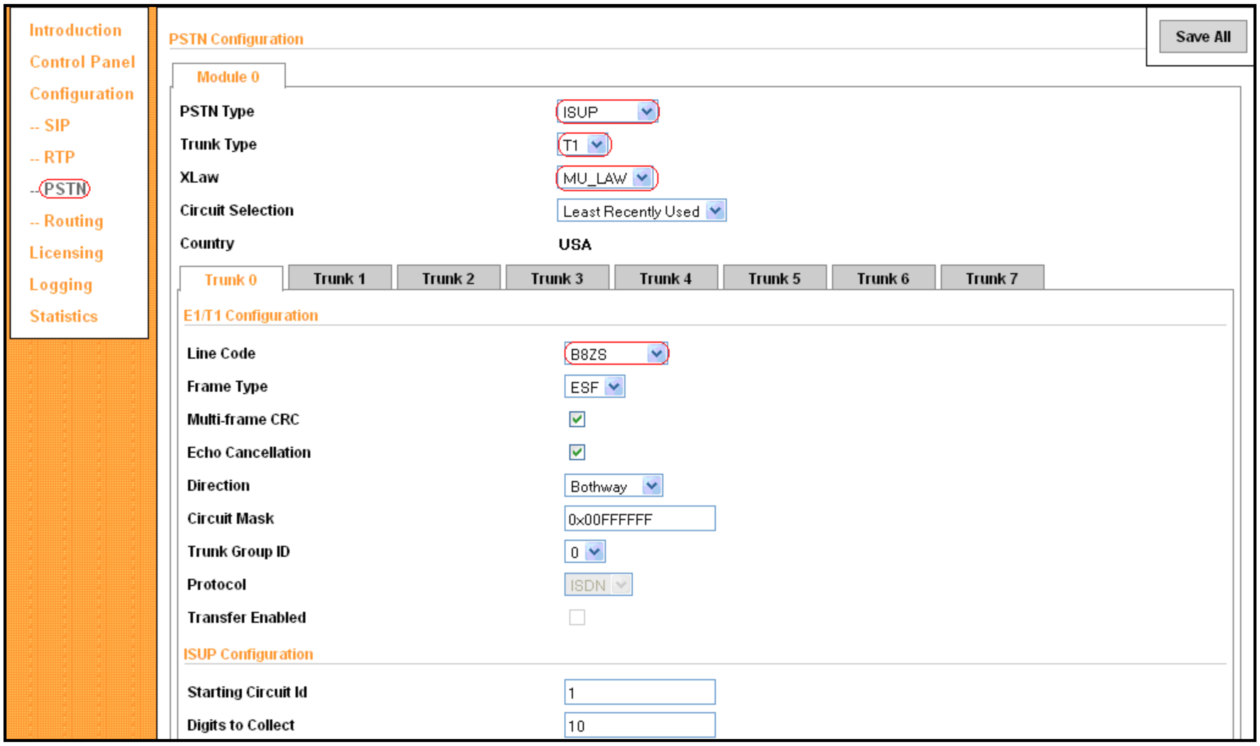
6.2.3 Configure the RTP Interface.

Navigate to the **RTP Configuration** screen. Provide the IP network configuration associated with the RTP interface of the Vision Media Gateway. Accept the default **Base Port** of **8000** and enable **RFC 2833** support. Select codecs with the preferred priority. In these Application Notes **G.729a** and **G.711-Mu** have been used. Click **Save All**.



6.2.4 Configure the SS7 Interface.

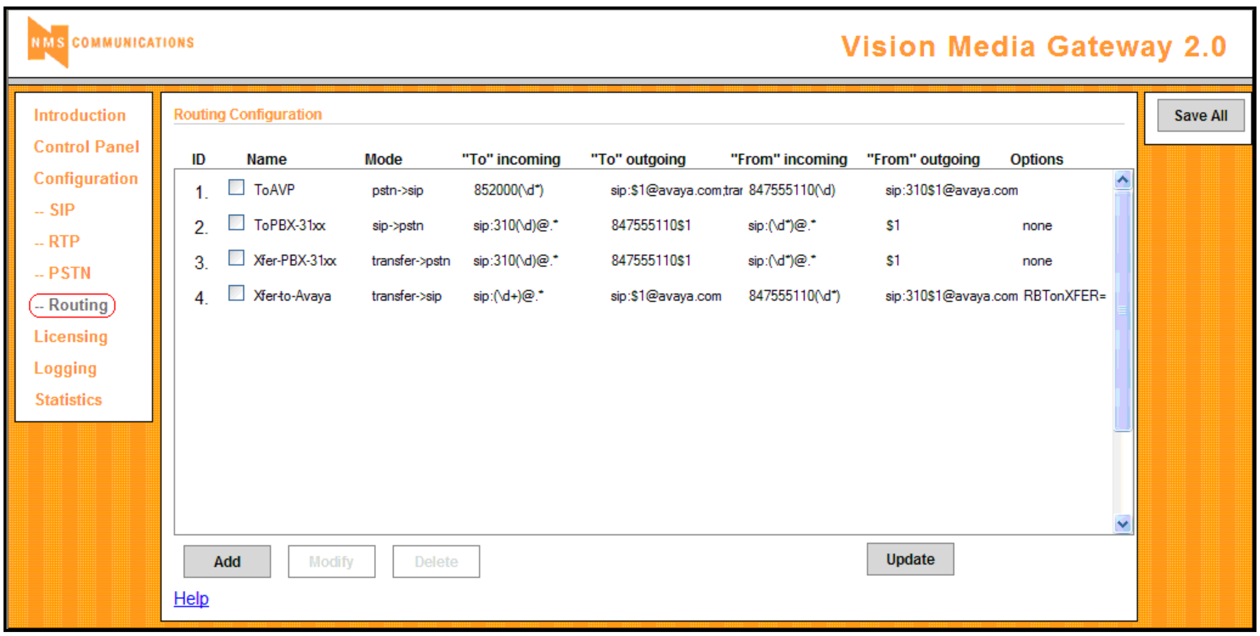
Once the SIP and RTP interfaces are configured, configure the SS7 interface to the PSTN. Navigate in the **PSTN Configuration** screen; configure the SS7 facility as shown in the figure below. Configure **ISUP** in the **PSTN Type** field**.** In this configuration, the **Trunk Type** was configured for **T1** with **B8ZS** as **Line Coding** and the ESF frame type. Click **Save All**. Refer to **[9]** for additional information on configuring the SS7 interface.



6.2.5 Configure Call Routing.

Use the Vision Media Gateway **Routing** Configuration screen to create a set of routing rules for the gateway. Routing table entries include from and to (or calling and called) patterns that are matched against the calling and called addresses received for an incoming call. For each matched pattern set, the routing table specifies the outgoing call mode (for example, SIP to PSTN or PSTN to SIP) and the substitution pattern for constructing the outgoing from and to addresses from the incoming addresses for that call. Separate routing rules are also required for transferred calls. Patterns are specified as JavaScript regular expressions.

In this configuration, two routes were used to route calls between the SIP and PSTN networks and two routes were used to route transferred calls. Below is a brief description of each call route. When multiple routes exist with the same Mode value, the order of the routes is important. The routes with the lower IDs take precedence over the ones with a higher ID.



Route ID #1 To AVP (Avaya Voice Portal), PSTN → SIP:

- The calling party dials 852 000 XXXX, the VG2000 will strip off the last 4 digits as stated in the "To Incoming" field. The VG2000 will then applies the rule of the "To outgoing" field and place the last 4 digits as a sip:XXXX@avaya.com message.
- For the called party 847 555 110X, the VG2000 will strip off the last digit as stated in the "From incoming" field. The VG2000 will then applies the rule of the "From outgoing" field and place the digit in a sip:310X@avaya.com message.

Route ID #2 To PBX-31xx, SIP → PSTN:

- The calling party dials 310X, the VG2000 will strip off the last digit as stated in the "To Incoming" field. The VG2000 will then applies the rule of the "To outgoing" field and re-place the 310X with 110X, to have a PSTN calling party of 847 555 110X.

- For the called party, any # that is sent by the AVP, what will be used as the PSTN called party as it is.

Route ID #3 Xfer-PBX-31xx, TRANSFER → PSTN:

- In the SIP Refer message from the AVP, The calling party field is 310X, the VG2000 will strip off the last digit as stated in the "To Incoming" field. The VG2000 will then applies the rule of the "To outgoing" field and re-place the 310X with 110X, to have a PSTN calling party of 847 555 110X.
- For the called party, any # that is sent by the AVP, what will be used as the PSTN called party as it is.

Route ID #4 Xfer-to-Avaya, TRANSFER → SIP:

- The transfer party is to AVP, any 4 digits that is sent to the VG2000 in the "To Incoming" field, will be copied and pasted into "To outgoing" field and as the 4 digits in the sip:XXXX@avaya.com message.
- For the called party 847 555 110X, the VG2000 will strip off the last digit as stated in the "From incoming" field. The VG2000 will then applies the rule of the "From outgoing" field and place the digit in a sip:310X@avaya.com message. The option **RBTonXFER**=, allows for ring back to played to the sip phone during call progress.

7. Configure NMS Communications Vision Signaling Server

This section provides the procedures for configuring the Vision Signaling Server (VS5000). The procedures require two distinct operations:

- Configuring the Vision Signaling Server IP network parameters
- Configuring the Vision Signaling Server SS7 network interface

7.1 Configure the IP Network Parameters

The Visions Signaling Server IP network configuration is managed by scripts and the procedures are described in **[8]**.

7.2 Configure SS7 Interface

The Vision Signaling Server network configuration is based on two configuration files:

- ***txcfg1.txt***: TDM configuration file, which defines the physical characteristics of the T1/E1 trunks.
- ***ss7_config_default.xml***: SS7 configuration, which defines the SS7 network configuration including signaling links, link sets, routes, and circuit groups.

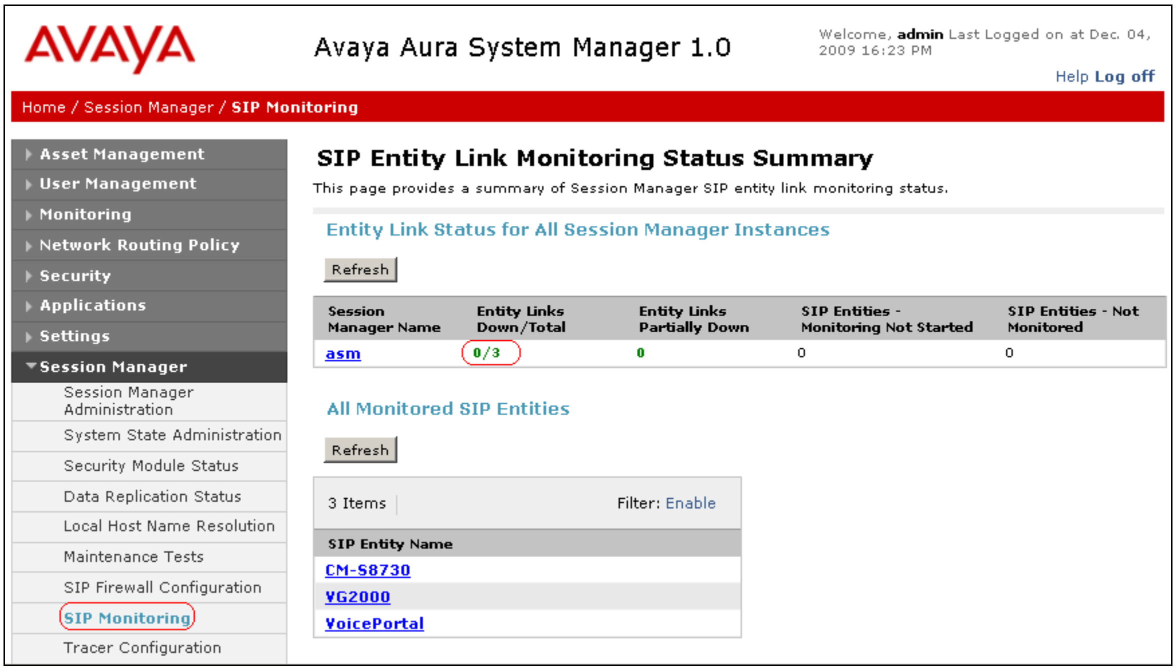
The configuration files used during testing are contained in the appendix of this document. Refer to **[10]** for information on how to configure these files.

8. Verification Steps

This section provides the verification steps that may be performed to verify that the Voice Portal can establish calls to the PSTN through the Vision Media Gateway.

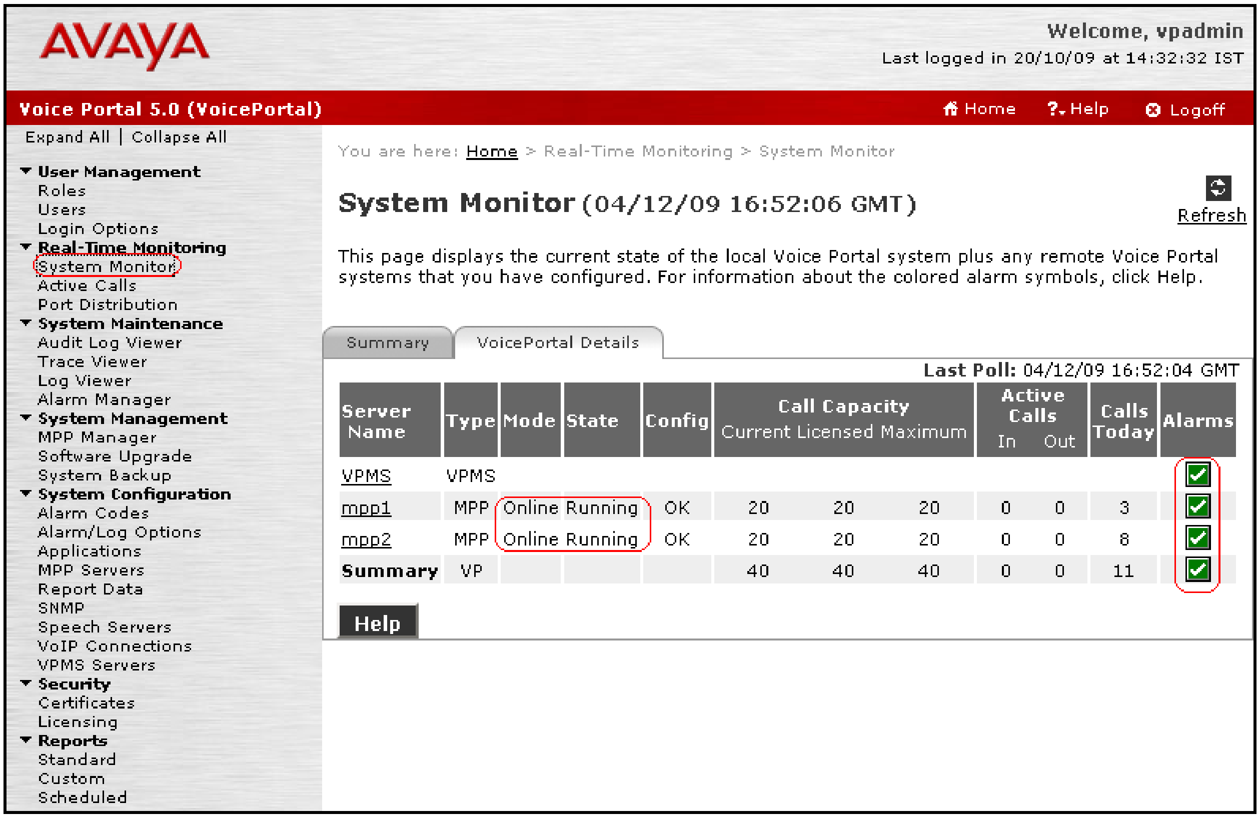
8.1 SIP Monitoring on Session Manager

Expand the Session Manager menu on the left and click SIP Monitoring. Verify that none of the links to the defined SIP entities are down, indicating that they are all reachable for call routing.



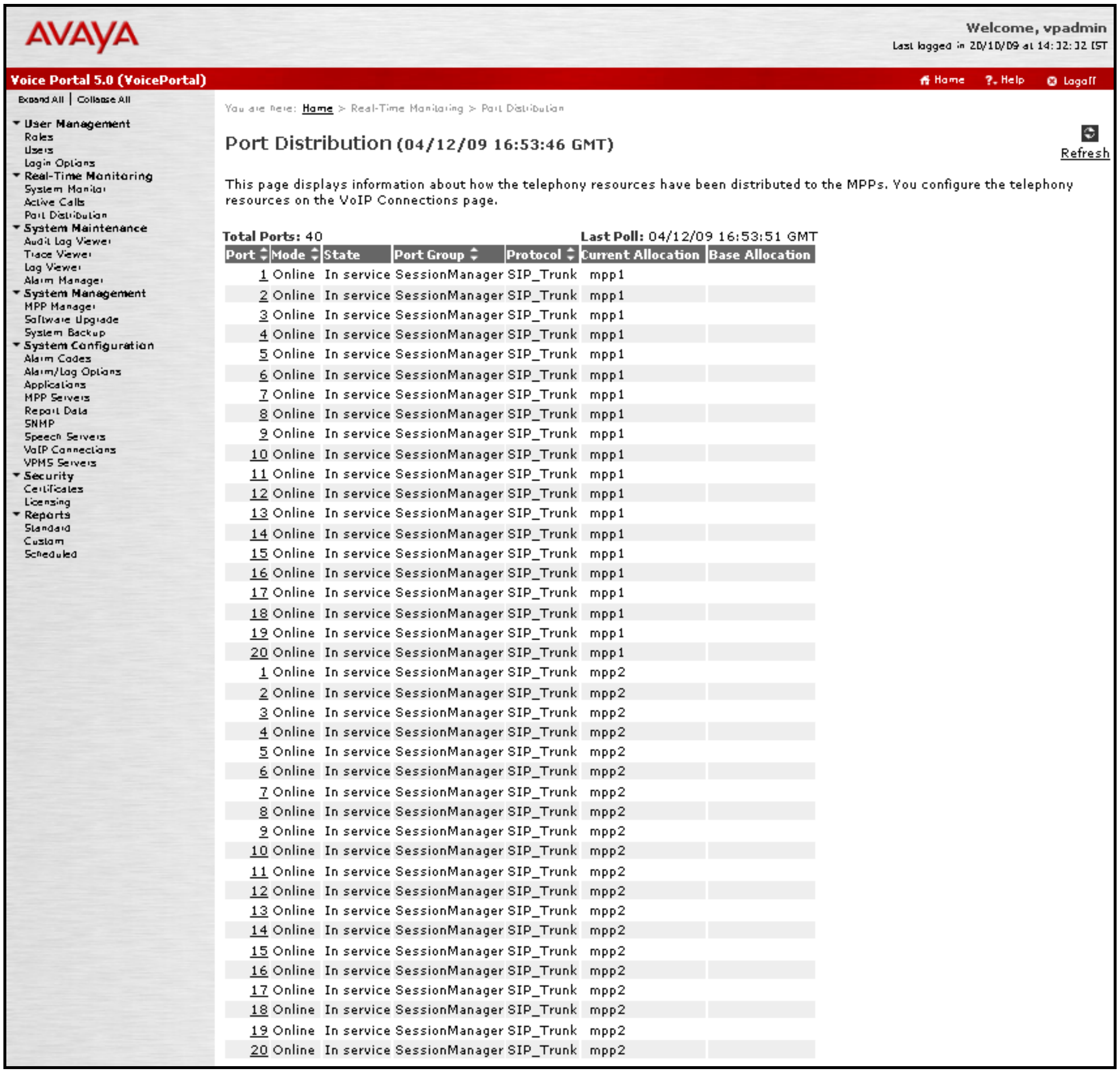
8.2 Voice Portal Monitoring – System Monitor

From the VPMS web interface, verify that the MPP servers are online and running and there are no alarms associated with VP system, in the **System Monitor** page shown **below**.



8.3 Voice Portal Monitoring – Port Distribution

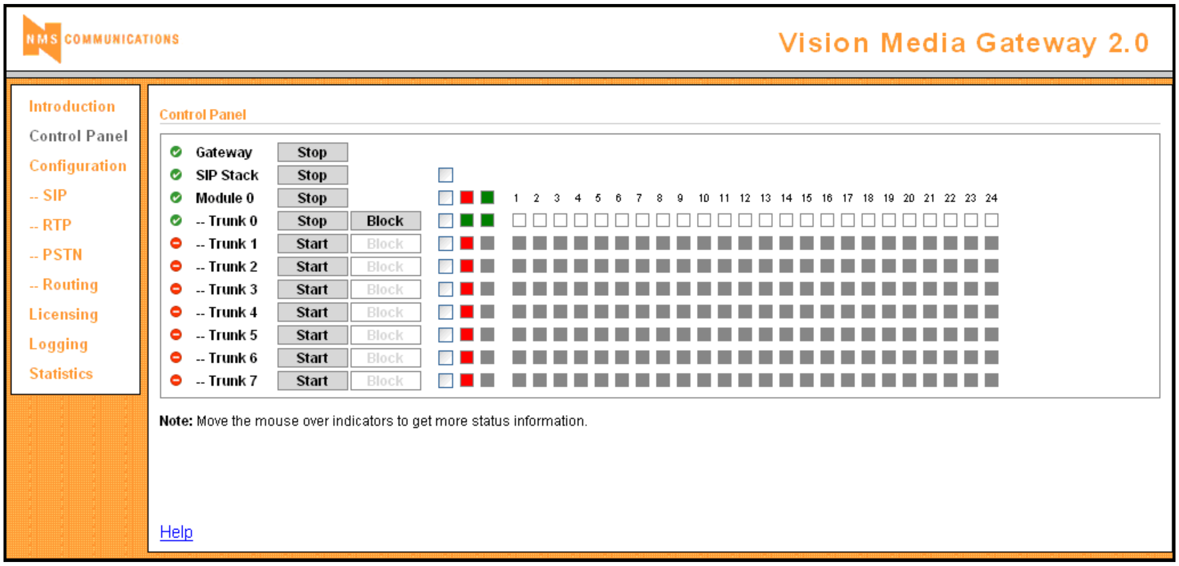
From the VPMS web interface, verify that the ports on the MPP servers are in-service in the **Port Distribution** page shown below.



Port	Mode	State	Port Group	Protocol	Current Allocation	Base Allocation
1	Online	In service	SessionManager	SIP_Trunk	mpp1	
2	Online	In service	SessionManager	SIP_Trunk	mpp1	
3	Online	In service	SessionManager	SIP_Trunk	mpp1	
4	Online	In service	SessionManager	SIP_Trunk	mpp1	
5	Online	In service	SessionManager	SIP_Trunk	mpp1	
6	Online	In service	SessionManager	SIP_Trunk	mpp1	
7	Online	In service	SessionManager	SIP_Trunk	mpp1	
8	Online	In service	SessionManager	SIP_Trunk	mpp1	
9	Online	In service	SessionManager	SIP_Trunk	mpp1	
10	Online	In service	SessionManager	SIP_Trunk	mpp1	
11	Online	In service	SessionManager	SIP_Trunk	mpp1	
12	Online	In service	SessionManager	SIP_Trunk	mpp1	
13	Online	In service	SessionManager	SIP_Trunk	mpp1	
14	Online	In service	SessionManager	SIP_Trunk	mpp1	
15	Online	In service	SessionManager	SIP_Trunk	mpp1	
16	Online	In service	SessionManager	SIP_Trunk	mpp1	
17	Online	In service	SessionManager	SIP_Trunk	mpp1	
18	Online	In service	SessionManager	SIP_Trunk	mpp1	
19	Online	In service	SessionManager	SIP_Trunk	mpp1	
20	Online	In service	SessionManager	SIP_Trunk	mpp1	
1	Online	In service	SessionManager	SIP_Trunk	mpp2	
2	Online	In service	SessionManager	SIP_Trunk	mpp2	
3	Online	In service	SessionManager	SIP_Trunk	mpp2	
4	Online	In service	SessionManager	SIP_Trunk	mpp2	
5	Online	In service	SessionManager	SIP_Trunk	mpp2	
6	Online	In service	SessionManager	SIP_Trunk	mpp2	
7	Online	In service	SessionManager	SIP_Trunk	mpp2	
8	Online	In service	SessionManager	SIP_Trunk	mpp2	
9	Online	In service	SessionManager	SIP_Trunk	mpp2	
10	Online	In service	SessionManager	SIP_Trunk	mpp2	
11	Online	In service	SessionManager	SIP_Trunk	mpp2	
12	Online	In service	SessionManager	SIP_Trunk	mpp2	
13	Online	In service	SessionManager	SIP_Trunk	mpp2	
14	Online	In service	SessionManager	SIP_Trunk	mpp2	
15	Online	In service	SessionManager	SIP_Trunk	mpp2	
16	Online	In service	SessionManager	SIP_Trunk	mpp2	
17	Online	In service	SessionManager	SIP_Trunk	mpp2	
18	Online	In service	SessionManager	SIP_Trunk	mpp2	
19	Online	In service	SessionManager	SIP_Trunk	mpp2	
20	Online	In service	SessionManager	SIP_Trunk	mpp2	

8.4 VG2000 PSTN Channel Verification

From the Configuration Manager of the Vision Media Gateway, navigate to the **Control Panel** and verify that SS7 interface and channels are in-service as shown in the following figure. The checkboxes are enabled so that the system comes up automatically after rebooting or cycling power.



8.5 Functional Verification

From a phone on the PSTN, place a call to an application on Voice Portal. Verify that the call is established successfully and that the proper greeting is provided. Transfer the call to another user.

9. Interoperability Compliance Testing

This section describes the interoperability compliance testing used to verify calls between Avaya Voice Portal and the PSTN through the NMS Communications Vision Media Gateway, which served as a SIP-to-PSTN gateway. This section covers the general test approach and the test results.

9.1 General Test Approach

The interoperability compliance test included feature, serviceability, and performance load testing. The feature testing focused on verifying the following:

- Placing calls from the Avaya SIP-based network to the PSTN
- Placing calls from the PSTN to Avaya Voice Portal
- Call transfers from Voice Portal to users on the PSTN and the Avaya SIP-based network
- Performing Blind, Supervised, and Bridged call transfers from Voice Portal to users on the PSTN and Avaya SIP-based network
- Sending UUI during call transfers from Voice Portal to the PSTN
- Receiving UUI from the PSTN to Voice Portal

The serviceability testing focused on verifying the ability of the VG2000 and VS5000 to recover from adverse conditions, such as power failures and disconnecting cables from the IP network.

9.2 Test Results

All test cases passed. Avaya Voice Portal was successful in establishing and transferring calls to users on the PSTN through the NMS Communications Vision Media Gateway.

10. Conclusion

As illustrated in these Application Notes, NMS Vision Media Gateway VG2000 can interoperate with Avaya Aura™ Voice Portal with Avaya Aura™ Session Manager using SIP trunks. The test used G711 and G729 codecs as media encoding.

11. Additional References

Avaya references, available at http://support.avaya.com

Avaya Aura™ Communication Manager:

1. *Administering Avaya Aura™ Communication Manager*, Doc ID 03-300509
2. *SIP Support in Avaya Aura™ Communication Manager Running on Avaya S8xxx Servers, Doc ID 555-245-206*

Avaya Aura™ Session Manager & Aura™ System Manager:

3. *Installing and Administering Avaya Aura™ Session Manager document id* 03-603324
4. *Avaya Aura™ Session Manager Overview, Doc ID 03-603323*
5. *Maintaining and Troubleshooting Avaya Aura™ Session Manager, Doc ID 03-603325*
6. *Installing the Avaya S8510 Server Family and Its Components*

Avaya Aura™ Voice Portal:

7. *Administering Voice Portal*

NMS Communications references available at http://www.nmscommunications.com

8. *Installing the Vision Server SR1500* Version 1.1, July 2008, Document ID 9000-62494-11.
9. *Vision Media Gateway Configuration and Administration Manual* Version 1.1, June 2008 Document ID 9000-62701-11.
10. *Installing the Vision VS 5000 Signaling Server*, Document ID 9000-62672-11, Version 1.1, June 2007.

12. APPENDIX

In this section are presented the relevant configuration files for the devices used in the DevConnect compliance testing.

12.1 VG2000 configuration file

Here follows the sample configuration file for the NMS VG2000.

```
#
#Wed Sep 30 11:33:41 CDT 2009
mod.0.trunk.1.immediatestart=0
mod.0.trunk.3.timewaitdial=500
route.4.mode=4
mod.0.trunk.5.isdn.side=user
mod.0.trunk.1.isdn.side=user
log.Isup.Debug=0
mod.0.trunk.5.isdn.primary=5
mod.0.trunk.3.cas.variant=ss50
route.1.opt=
log.Session.ObjState=1
mod.0.trunk.2.RTCdigitnumber=9
log.Controller.CtaApi=0
log.IpTrunk.Timer=0
mod.0.trunk.7.signalingmethod=0
mod.0.trunk.7.isdn.backup=0
route.3.to.out=847555$1
mod.0.trunk.1.linecode=B8ZS
route.6.from.in=847555110(\\d*)
mod.0.trunk.5.direction=Bothway
mod.0.trunk.1.direction=Bothway
mod.0.trunk.length=8
mod.0.trunk.4.decadicsignalmethod=0
route.4.from.out=$1
log.Session.Error=1
mod.0.trunk.4.isup.waitdigits=2000
log.Gateway.Component=0
log.Timer.Info=1
log.IpTrunk.NmsApi=0
mod.0.trunk.1.cas.varparam.length=0
route.4.to.in=sip\:310(\\d)@.*
route.6.name=Xfer-to-Avaya
sip.tcp=1
log.Cas.Info=1
mod.0.trunk.2.transferEnabled=0
mod.0.trunk.4.direction=Bothway
mod.0.trunk.2.trunkgrpId=0
log.Port.Error=1
route.0.to.in=sip\:(847555\\d+)@.*
mod.0.trunk.5.autostart=0
route.0.from.out=$1
mod.0.trunk.1.autostart=0
log.Session.Info=1
mod.0.trunk.0.dialpulsemethod=0
route.length=7
log.Timer.Warn=1
route.0.opt=
sip.outproxy.transport=tcp
mod.0.trunk.4.isup.digits=10
log.Trunk.Timer=0
```

```
route.4.to.out=847555110$1
log.Cas.Warn=1
log.Timer.ObjCmd=0
log.Board.Timer=0
log.Port.ObjEvt=1
mod.0.trunk.4.ANINumber=8
mod.0.trunk.0.ANINumber=8
mod.0.trunk.3.signalingmethod=0
log.Session.Warn=1
mod.0.trunk.7.trunkgrpId=0
mod.0.trunk.3.dialpulsemethod=0
mod.0.trunk.7.isup.circstart=169
mod.0.trunk.0.answerGroupA=0
mod.0.rtp.codec4=-1
mod.0.rtp.codec3=-1
mod.0.trunk.7.RTCdigitnumber=9
mod.0.trunk.7.ANINumber=8
mod.0.rtp.codec2=0
mod.0.trunk.7.sendanididwink=0
mod.0.rtp.codec1=18
mod.0.trunk.5.mode=ISUP
log.Cas.Error=1
mod.0.trunk.7.linecode=B8ZS
log.Isup.Info=1
mod.0.trunk.7.isdn.primary=7
log.Gateway.CtaApi=0
mod.0.trunk.2.echo=1
route.3.mode=4
mod.0.trunk.7.decadicsignalmethod=0
mod.0.trunk.1.circmask=0x00FFFFFF
route.3.to.in=sip\:(110\\d)@.*
mod.0.trunk.2.CIDsupport=0
log.Trunk.ObjCmd=0
log.Isup.Warn=1
mod.0.trunk.3.linecode=B8ZS
mod.0.trunk.3.isup.waitdigits=2000
mod.0.trunk.5.cas.varparam.length=0
mod.0.trunk.0.timewaitdial=500
mod.0.trunk.2.mfcrc=1
mod.0.trunk.1.isdn.intid=1
route.6.from.out=sip\:310$1@avaya.com
mod.0.trunk.2.isdn.primary=2
log.Port.NmsApi=1
mod.0.trunk.1.transferEnabled=0
mod.0.trunk.3.immediatestart=0
log.Isup.Error=1
log.Isdn.Info=1
mod.0.trunk.4.networkside=0
log.Gateway.Debug=0
route.4.from.in=sip\:(\\d*)@.*
mod.0.trunk.4.RTCdigitnumber=9
route.5.name=Xfer-PBX-31xx
mod.0.trunk.4.sendanididwink=0
mod.0.trunk.7.CIDsupport=0
log.Board.ObjCmd=0
log.Cas.NmsEvent=0
log.Timer.ObjEvt=0
log.Session.CtaApi=1
route.0.from.in=sip\:(\\d*)@.*
route.2.from.out=sip\:310$1@avaya.com
sip.outproxy.addr=193.120.221.156
mod.0.trunk.6.isdn.intid=6
```

```
log.Isdn.Warn=1
mod.0.trunk.0.isup.circstart=1
mod.0.trunk.6.isdn.variant=20
mod.0.trunk.2.signalingmethod=0
mod.0.trunk.2.expectanididwink=0
mod.0.trunktype=T1
mod.0.trunk.0.decadicsignalmethod=0
mod.0.trunk.0.isdn.backup=0
mod.0.trunk.2.answerGroupA=0
route.6.to.in=sip\:(\\d+)@.*
log.Session.Timer=0
mod.0.trunk.0.cas.variant=ss50
sip.outproxy.port=5060
mod.0.trunk.0.isdn.variant=20
mod.0.trunk.1.DIDNumber=7
mod.0.trunk.4.isdn.backup=0
mod.0.trunk.5.mfcrc=1
mod.0.trunk.0.immediatestart=0
log.Isdn.Debug=0
mod.0.trunk.1.frametype=ESF
log.IpTrunk.CtaApi=0
log.Trunk.CtaApi=0
mod.0.trunk.4.mode=ISUP
log.Controller.Info=1
mod.0.trunk.4.cas.variant=ss50
log.Port.Timer=0
mod.0.trunk.1.expectanididwink=0
log.SipStack.ObjCmd=0
mod.0.xlaw=MU_LAW
log.Trunk.ObjEvt=0
mod.0.trunk.1.mfcrc=1
mod.0.trunk.1.RTCdigitnumber=9
log.IpTrunk.NmsEvent=0
mod.0.trunk.4.DIDNumber=7
mod.0.trunk.0.DIDNumber=7
log.Port.Info=1
mod.0.trunk.1.sendanididwink=0
log.Board.NmsEvent=0
mod.0.trunk.1.echo=1
log.Timer.NmsApi=0
mod.0.trunk.4.frametype=ESF
route.2.mode=1
mod.0.trunk.0.frametype=ESF
mod.0.trunk.2.timewaitdial=500
mod.0.trunk.5.trunkgrpId=0
mod.0.trunk.3.circmask=0x00FFFFFF
log.Trunk.Info=1
log.Cas.ObjState=0
log.MediaPort.Info=1
log.MediaPort.NmsEvent=0
log.Controller.Warn=1
mod.0.trunk.4.isdn.primary=4
route.1.to.out=$1
log.Port.Warn=1
log.Board.ObjEvt=0
mod.0.trunk.2.signalingtype=0
log.Cas.Timer=0
log.Trunk.Warn=1
mod.0.trunk.7.dialpulsemethod=0
log.MediaPort.Warn=1
log.SipStack.Component=0
mod.0.trunk.4.isdn.side=user
```

```
mod.0.trunk.0.isdn.side=user
mod.0.trunk.3.decadicsignalmethod=0
mod.0.pstntype=ISUP
route.4.name=ToPBX-31xx
mod.0.trunk.5.isup.digits=10
log.Timer.Debug=0
mod.0.trunk.0.CIDsupport=0
mod.0.trunk.0.cas.varparam.length=0
log.Isup.ObjCmd=0
log.Trunk.NmsApi=0
route.2.from.in=847555110(\\d)
mod.0.trunk.7.isdn.side=user
mod.0.trunk.3.isdn.side=user
mod.0.trunk.0.direction=Bothway
mod.0.trunk.0.isup.digits=10
mod.0.trunk.1.signalingtype=0
log.IpTrunk.ObjState=0
log.Board.ObjState=0
mod.0.trunk.6.isup.circstart=145
mod.0.trunk.5.answerGroupA=0
log.Gateway.Error=1
log.SipStack.ObjEvt=0
mod.0.trunk.6.sendanididwink=0
mod.0.trunk.7.isup.waitdigits=2000
mod.0.trunk.5.CIDsupport=0
mod.0.trunk.7.direction=Bothway
mod.0.trunk.3.direction=Bothway
mod.0.trunk.3.isdn.variant=20
mod.0.trunk.7.expectanididwink=0
log.Isup.Timer=0
log.MediaPort.ObjState=0
log.Board.NmsApi=0
log.Port.CtaApi=1
mod.0.trunk.5.transferEnabled=0
mod.0.trunk.4.isdn.intid=4
mod.0.trunk.4.autostart=0
mod.0.trunk.0.autostart=0
log.Isdn.NmsEvent=0
mod.0.trunk.6.immediatestart=0
mod.0.trunk.3.mode=ISUP
mod.0.trunk.0.signalingtype=0
mod.0.trunk.7.autostart=0
mod.0.trunk.0.echo=1
mod.0.trunk.3.autostart=0
mod.0.trunk.3.ANINumber=8
mod.0.trunk.1.networkside=0
mod.0.trunk.5.circmask=0x00FFFFFF
mod.0.trunk.5.timewaitdial=500
mod.0.trunk.4.linecode=B8ZS
mod.0.trunk.7.echo=1
route.1.mode=4
log.IpTrunk.Component=0
mod.0.trunk.6.expectanididwink=0
mod.0.trunk.6.signalingmethod=0
log.Isdn.Error=1
mod.0.rtp.port=8000
mod.0.trunk.6.decadicsignalmethod=0
mod.0.trunk.6.ANINumber=8
mod.0.trunk.5.networkside=0
mod.0.trunk.3.isup.circstart=73
mod.0.trunk.2.ANINumber=8
mod.0.trunk.6.dialpulsemethod=0
```

```
route.5.to.out=847555110$1
mod.0.trunk.3.RTCdigitnumber=9
mod.0.trunk.3.sendanididwink=0
mod.0.trunk.3.cas.varparam.length=0
mod.0.trunk.0.linecode=B8ZS
log.SipStack.NmsApi=0
log.Isup.ObjEvt=0
mod.0.trunk.4.cas.varparam.length=0
mod.0.trunk.7.signalingtype=0
mod.0.trunk.1.isdn.primary=1
mod.0.rtp.gateway=65.162.27.65
mod.0.trunk.0.networkside=0
log.MediaPort.Debug=0
mod.0.trunk.3.trunkgrpId=0
mod.0.rtp.mask=255.255.255.192
mod.0.trunk.5.expectanididwink=0
route.3.name=ToPBX-4
mod.0.trunk.1.isdn.backup=0
log.Cas.ObjCmd=0
mod.0.trunk.7.answerGroupA=0
sip.ip=65.162.27.80
log.Isdn.ObjState=0
mod.0.trunk.1.cas.variant=ss50
log.Timer.CtaApi=0
mod.0.trunk.6.isup.waitdigits=2000
log.MediaPort.ObjCmd=0
mod.0.trunk.5.isdn.variant=20
mod.0.trunk.5.isdn.backup=0
log.Timer.Component=0
log.SipStack.Info=1
mod.0.trunk.6.signalingtype=0
mod.0.trunk.4.transferEnabled=0
log.Timer.Error=1
mod.0.trunk.5.cas.variant=ss50
mod.0.trunk.0.RTCdigitnumber=9
mod.0.trunk.0.sendanididwink=0
mod.0.trunk.2.dialpulsemethod=0
sys.version=2.0
log.Isup.NmsApi=0
log.SipStack.Warn=1
mod.0.trunk.7.timewaitdial=500
mod.0.trunk.2.mode=ISUP
mod.0.boardtype=CG_6565
mod.0.trunk.7.circmask=0x00FFFFFF
log.Isdn.ObjCmd=0
mod.0.trunk.5.signalingmethod=0
mod.0.trunk.6.linecode=B8ZS
route.2.to.in=852000(\\d*)
mod.0.trunk.3.CIDsupport=0
mod.0.trunk.6.echo=1
mod.0.trunk.7.cas.varparam.length=0
route.0.mode=7
log.Gateway.Timer=0
mod.0.trunk.2.isdn.intid=2
mod.0.trunk.3.isdn.primary=3
mod.0.trunk.2.isup.waitdigits=2000
log.SipStack.Debug=0
log.Controller.Debug=0
mod.0.trunk.6.isup.digits=10
mod.0.trunk.2.linecode=B8ZS
mod.0.trunk.7.DIDNumber=7
mod.0.trunk.3.DIDNumber=7
```

```
log.Cas.ObjEvt=0
mod.0.trunk.0.transferEnabled=0
log.Controller.ObjCmd=0
route.5.from.out=$1
log.Board.CtaApi=0
mod.0.trunk.7.frametype=ESF
mod.0.trunk.3.frametype=ESF
route.5.from.in=sip\:(\\d*)@.*
log.MediaPort.ObjEvt=0
mod.0.trunk.0.expectanididwink=0
log.Session.NmsEvent=1
mod.0.trunk.1.isup.digits=10
mod.0.trunk.6.DIDNumber=7
mod.0.trunk.7.isdn.intid=7
route.2.name=ToAVP
mod.0.trunk.5.isup.circstart=121
route.1.from.in=sip\:(\\d*)@.*
mod.0.trunk.2.decadicsignalmethod=0
mod.0.trunk.6.frametype=ESF
mod.0.trunk.3.transferEnabled=0
log.IpTrunk.Debug=0
route.1.from.out=$1
log.Isdn.Timer=0
mod.0.trunk.1.signalingmethod=0
route.5.to.in=sip\:310(\\d)@.*
mod.0.trunk.1.dialpulsemethod=0
log.MediaPort.Error=1
mod.0.trunk.1.trunkgrpId=0
mod.0.trunk.4.answerGroupA=0
mod.0.trunk.4.mfcrc=1
mod.0.trunk.5.immediatestart=0
route.2.to.out=sip\:$1@avaya.com;transport\=tcp
route.1.to.in=sip\:(847555110\\d)@.*
route.6.opt=RBTonXFER\=1
log.Isdn.ObjEvt=0
mod.0.trunk.2.isdn.variant=20
log.Controller.NmsEvent=0
mod.0.trunk.0.mfcrc=1
log.SipStack.CtaApi=0
mod.0.trunk.6.RTCdigitnumber=9
mod.0.trunk.2.networkside=0
mod.0.trunk.6.isdn.side=user
mod.0.trunk.2.isdn.side=user
log.Cas.NmsApi=0
log.Board.Component=0
sip.autostart=0
mod.0.trunk.1.mode=ISUP
log.MediaPort.NmsApi=0
mod.0.trunk.6.trunkgrpId=0
mod.0.trunk.6.networkside=0
log.Port.NmsEvent=1
mod.0.trunk.2.isup.circstart=49
log.Trunk.Component=0
log.Trunk.Debug=0
mod.0.trunk.1.isup.waitdigits=2000
mod.0.trunk.6.direction=Bothway
mod.0.trunk.5.echo=1
mod.0.trunk.2.direction=Bothway
log.Gateway.Info=1
mod.0.trunk.4.timewaitdial=500
log.Board.Debug=0
mod.0.trunk.2.sendanididwink=0
```

```
log.Controller.ObjEvt=0
route.6.mode=6
mod.0.trunk.6.isdn.primary=6
mod.0.trunk.2.circmask=0x00FFFFFF
log.Timer.Timer=0
log.Gateway.ObjCmd=0
mod.0.trunk.7.cas.variant=ss50
route.5.opt=none
mod.0.trunk.2.immediatestart=0
mod.0.trunk.5.decadicsignalmethod=0
mod.0.trunk.2.isdn.backup=0
log.Gateway.Warn=1
mod.0.trunk.7.mfcrc=1
mod.0.trunk.6.autostart=0
mod.0.trunk.2.autostart=0
mod.0.trunk.1.CIDsupport=0
log.Isdn.NmsApi=0
log.SipStack.NmsEvent=0
mod.0.trunk.0.isdn.primary=0
mod.0.trunk.2.cas.varparam.length=0
mod.0.rtp.rfc2833=1
mod.0.trunk.2.cas.variant=ss50
log.Isup.CtaApi=0
log.Board.Info=1
mod.0.trunk.3.mfcrc=1
mod.0.trunk.0.signalingmethod=0
mod.0.trunk.6.isdn.backup=0
route.1.name=ToIsdnPbx
mod.0.trunk.0.isdn.intid=0
log.Controller.ObjState=0
log.SipStack.Error=1
mod.0.trunk.5.ANINumber=8
route.3.from.in=sip\:(\\d*)@.*
mod.0.trunk.1.ANINumber=8
log.Gateway.NmsEvent=0
log.Controller.Error=1
route.3.from.out=$1
mod.0.trunk.6.cas.variant=ss50
log.Trunk.NmsEvent=0
mod.0.trunk.6.answerGroupA=0
mod.0.autostart=0
mod.0.trunk.6.CIDsupport=0
log.Board.Warn=1
mod.0.trunk.4.isdn.variant=20
sip.enableUUIhdr=1
log.Port.ObjState=1
log.Session.ObjCmd=1
log.Controller.NmsApi=0
route.4.opt=none
mod.0.trunk.5.isdn.intid=5
route.6.to.out=sip\:$1@avaya.com
log.Timer.NmsEvent=0
log.IpTrunk.Error=1
mod.0.trunk.1.answerGroupA=0
log.Isup.Component=0
mod.0.trunk.0.isup.waitdigits=2000
mod.0.trunk.4.expectanididwink=0
mod.0.trunk.3.isup.digits=10
log.Controller.Component=0
log.MediaPort.Component=0
log.Gateway.ObjEvt=0
mod.0.trunk.6.timewaitdial=500
```

```
mod.0.trunk.0.mode=ISUP
mod.0.circuit.algorithm=0
mod.0.trunk.7.mode=ISUP
mod.0.trunk.7.immediatestart=0
log.SipStack.ObjState=0
mod.0.trunk.6.mfcrc=1
mod.0.trunk.7.isup.digits=10
mod.0.trunk.4.echo=1
log.MediaPort.Timer=0
mod.0.trunk.7.transferEnabled=0
mod.0.trunk.4.circmask=0x00FFFFFF
log.IpTrunk.ObjCmd=0
route.5.mode=7
log.Session.Debug=1
mod.0.trunk.5.signalingtype=0
log.Gateway.ObjState=0
mod.0.trunk.6.cas.varparam.length=0
mod.0.trunk.5.dialpulsemethod=0
log.Trunk.ObjState=0
route.3.opt=none
mod.0.trunk.3.expectanididwink=0
mod.0.trunk.2.isup.digits=10
mod.0.trunk.1.timewaitdial=500
mod.0.trunk.4.trunkgrpId=0
mod.0.trunk.4.isup.circstart=97
log.Trunk.Error=1
log.Port.Debug=1
log.IpTrunk.Info=1
mod.0.trunk.0.circmask=0x00FFFFFF
log.Session.Component=1
log.Board.Error=1
log.Cas.CtaApi=0
log.Isup.NmsEvent=0
log.Timer.ObjState=0
log.Session.ObjEvt=1
log.Cas.Component=0
route.0.name=XferToPBX
log.MediaPort.CtaApi=0
mod.0.trunk.4.signalingtype=0
log.Gateway.NmsApi=0
log.IpTrunk.Warn=1
mod.0.trunk.4.immediatestart=0
mod.0.trunk.2.DIDNumber=7
mod.0.trunk.5.isup.waitdigits=2000
mod.0.trunk.1.decadicsignalmethod=0
mod.0.trunk.7.isdn.variant=20
mod.0.trunk.3.networkside=0
log.Isdn.Component=0
mod.0.trunk.5.RTCdigitnumber=9
mod.0.trunk.2.frametype=ESF
mod.0.trunk.5.sendanididwink=0
route.0.to.out=$1
log.Cas.Debug=0
route.2.opt=
mod.0.trunk.5.DIDNumber=7
mod.0.trunk.3.answerGroupA=0
mod.0.trunk.0.trunkgrpId=0
sip.port=5060
mod.0.trunk.7.networkside=0
log.Port.Component=1
mod.0.trunk.5.frametype=ESF
mod.0.trunk.1.isup.circstart=25
```

```
mod.0.trunk.1.isdn.variant=20
mod.0.trunk.3.signalingtype=0
log.IpTrunk.ObjEvt=0
log.Isdn.CtaApi=0
log.SipStack.Timer=0
mod.0.trunk.6.transferEnabled=0
mod.0.trunk.4.CIDsupport=0
log.Controller.Timer=0
mod.0.trunk.6.mode=ISUP
mod.0.trunk.4.signalingmethod=0
log.Port.ObjCmd=1
mod.0.rtp.ip=65.162.27.81
log.Session.NmsApi=1
mod.0.trunk.4.dialpulsemethod=0
mod.0.trunk.6.circmask=0x00FFFFFF
mod.0.trunk.3.isdn.intid=3
mod.0.trunk.5.linecode=B8ZS
mod.0.trunk.3.echo=1
log.Isup.ObjState=0
mod.0.trunk.3.isdn.backup=0
```

12.2 VS5000 configuration files

Here follows the sample configuration file ***ss7_config_default.xml*** for the NMS VS5000.

```
<Properties xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="ss7_config.xsd">
  <Version>1.0</Version>
  <AutoStart>Yes</AutoStart>
  <State>OutOfService</State>
  <Config>
    <MtpConfig>
      <GenConfig>
        <OPC>1</OPC>
        <DefaultVariant>ANSI</DefaultVariant>
        <DefaultDPC>2</DefaultDPC>
        <NodeType>SP</NodeType>
        <MaxLinks>4</MaxLinks>
        <MaxUsers>2</MaxUsers>
        <MaxLinksets>2</MaxLinksets>
        <MaxRoutes>32</MaxRoutes>
      </GenConfig>
      <NsapConfig Index="1">
        <!--  Nsap #1 used by ISUP -->
      </NsapConfig>
      <NsapConfig Index="2">
        <!--  Nsap #2 reserved for future use by SCCP or other MTP user -->
      </NsapConfig>
      <LinkConfig Index="1">
        <!--  Link #1: first link to DPC 2 -->
       <Server>SS701</Server>
        <PortNumber>1</PortNumber>
        <LinkSLC>0</LinkSLC>
        <Linkset>1</Linkset>
        <Ssf>National</Ssf>
      </LinkConfig>
      <LinkConfig Index="2">
        <!--  Link #2: second link to DPC 2 -->
        <Server>SS701</Server>
        <PortNumber>2</PortNumber>
        <LinkSLC>1</LinkSLC>
        <Linkset>1</Linkset>
```

```
        <Ssf>National</Ssf>
      </LinkConfig>
      <LinksetConfig Index="1">
        <!--  Linkset #1: direct link set to DPC 2 (DefaultDpc) -->
        <TargetNmbActLinks>2</TargetNmbActLinks>
        <Route Index="1">
          <RouteNumber>2</RouteNumber>
          <Priority>0</Priority>
        </Route>
      </LinksetConfig>
      <RouteConfig Index="1">
        <!--  NOTE: This first route is the UP (inbound) route, therefore its DPC is
the local point code -->
        <DPC>1</DPC>
        <Ssf>National</Ssf>
        <Direction>Up</Direction>
      </RouteConfig>
      <RouteConfig Index="2">
        <!--  Direct route to attached DPC 2, via link set #1 -->
        <DPC>2</DPC>
        <Ssf>National</Ssf>
        <Direction>Down</Direction>
        <AdjRoute>true</AdjRoute>
      </RouteConfig>
    </MtpConfig>
    <IsupConfig>
      <GenConfig>
        <DefaultVariant>ANSI95</DefaultVariant>
        <MaxCircuits>1920</MaxCircuits>
        <MaxGroups>32</MaxGroups>
        <MaxCallRefs>1920</MaxCallRefs>
        <ExtElmts>true</ExtElmts>
      </GenConfig>
      <CircConfig Index="1">
        <!--  First voice T1 to attached DPC 2 -->
        <Circuit>1</Circuit>
        <CIC>1</CIC>
        <NumCircuits>24</NumCircuits>
        <Direction>Bothway</Direction>
        <UnusedCircuits>None</UnusedCircuits>
        <Ssf>National</Ssf>
      </CircConfig>
      <UsapConfig Index="1">
        <!--  Only 1 ISUP user, the ssp_server -->
      </UsapConfig>
      <NsapConfig Index="1">
        <!--  Matches the MTP Nsap reserved for use by ISUP -->
        <Ssf>National</Ssf>
      </NsapConfig>
    </IsupConfig>
    <SspConfig>
      <GenConfig>
        <Server1>SS701</Server1>
        <Server2>SS702</Server2>
      </GenConfig>
    </SspConfig>
  </Config>
</Properties>
```

Here follows the sample configuration file ***txcfg1.txt*** for the NMS VS5000.

```
#
# use clock recovered from trunk 1 as board's clock and drive H.100/H.110 A clocks
clock net=1 a
# use clock recovered from trunk 2 as network reference clock (drive NR1 signal)
netref 2 nr1
#
# Configure all 4 trunks as T1 mode (not loop master)
#-------------------------------------------------------------------------------
#     Trunk     Framing   Encoding   Buildout   Loop Master
t1cfg 1         esf       b8zs       0          false
t1cfg 2         esf       b8zs       0          false
t1cfg 3         esf       b8zs       0          false
t1cfg 4         esf       b8zs       0          false
# define ports that SS7 links will connect through
#-------------------------------------------------------------------------------
#    PortNum    L|H|E|T|J   Trunk   Channel     Speed
port 1          t1          1       0
port 2          t1          2       23
```

12.3 Web application server intro.vxml file

Here it is presented the **intro.vxml** file used in these Application Notes.

```
<?xml version="1.0" ?>
<vxml version="2.1" xmlns="http://www.w3.org/2001/vxml" xml:lang="en-US" >

<form id="form0">

        <field name="test_type">

             <prompt bargein="true" cond="session.connection.ccxml.values.test_page ==
'true'">
                    <audio src="prompts/introccxml.wav"/>
             </prompt>

             <prompt bargein="true" cond="session.connection.ccxml.values.test_page ==
undefined">
                    <audio src="prompts/introvxml.wav"/>
             </prompt>

                <grammar src="builtin:dtmf/digits" />

                <filled>
                    <if cond="test_type == 1">
                           <goto next="asrtest.vxml"/>
                    <elseif cond="test_type == 2"/>
                           <goto next="ttstest.vxml"/>
                    <elseif cond="test_type == 3"/>
                           <goto next="testbridgetransfer.vxml"/>
                    <elseif cond="test_type == 4"/>
                           <goto next="testblindtransfer.vxml"/>
                    <elseif cond="test_type == 5"/>
                           <goto next="testconsulttransfer.vxml"/>
                    <elseif cond="test_type == 6"/>
                           <goto next="playprompts.vxml"/>
                    <elseif cond="session.connection.ccxml.values.test_page ==
'true'"/>
                           <if cond="test_type > 9">
                                  <prompt bargein="false">
                                         <audio src="prompts/commonSorry.wav"/>
                                  </prompt>
                                  <clear namelist="test_type"/>
                           <elseif cond="test_type == 0"/>
                                  <prompt bargein="false">
                                         <audio src="prompts/Exit.wav"/>
                                  </prompt>
                           <else/>
                                  <exit namelist="test_type"/>
                           </if>
                    <else/>
                           <if cond="test_type == 7">
                                  <log expr="'Getting Ready To Exit'"/>
                                  <prompt bargein="false">
                                         <audio src="prompts/Exit.wav"/>
                                  </prompt>
                                  <exit/>
                           <else/>
                                  <prompt bargein="false">
                                         <audio src="prompts/commonSorry.wav"/>
```

```
                                </prompt>
                                <clear namelist="test_type"/>
                        </if>
                </if>

            </filled>

          <noinput>
                <prompt bargein="false">
                             <audio src="prompts/commonSorry.wav"/>
                    </prompt>
                <reprompt/>
          </noinput>

    </field>

</form>
</vxml>
```

12.4 Web application server testblindtransfer.vxml file

Here it is presented the **testblindtransfer.vxml** file used in these Application Notes

```
<?xml version="1.0" ?>
<vxml version="2.1" xmlns="http://www.w3.org/2001/vxml" xml:lang="en-US" >

	<var name="var1" expr="'tel:'"/>

	<form id="get_number">

		<field name="phone_number">

			<prompt bargein="true">
				<audio src="prompts/TransferGetNumber.wav"/>
			</prompt>

			<grammar src="builtin:dtmf/digits?minlength=1;maxlength=10" />

				<noinput>
				<prompt bargein="false">
					<audio src="prompts/TransferNoNumberSorry.wav"/>
				</prompt>
				<reprompt/>
			</noinput>

		</field>

		<transfer name="blindtransfer" destexpr="var1 + phone_number"
type="blind" aai="abcdefg1234567890">

			<prompt bargein="false">
				<audio src="prompts/blindPerforming.wav"/>
			</prompt>

			    <filled>
				      <if cond="blindtransfer == 'near_end_disconnect'">
				       <audio src="prompts/nearEndDisc.wav"/>
					       <log> near_end_disconnect </log>
				      <elseif cond="blindtransfer == 'unknown'"/>
				       <audio src="prompts/failedUnknown.wav"/>
					       <log> unknown </log>
				      </if>
				<goto next="intro.vxml"/>
			    </filled>

		</transfer>

		<catch event="connection.disconnect.transfer">
			<log> connection.disconnect.transfer </log>
			<exit />
		</catch>
		<catch event="error.connection.noauthorization">
			<log> error.connection.noauthorization </log>
			<goto next="intro.vxml"/>
		</catch>
		<catch event="error.connection.baddestination">
			<log> error.connection.baddestination </log>
			<goto next="intro.vxml"/>
```

```
              </catch>
                <catch event="error.unsupported.uri">
                        <log> error.unsupported.uri </log>
                        <goto next="intro.vxml"/>
                </catch>
              <catch event="error.unsupported.transfer.blind">
                    <log> error.unsupported.transfer.blind </log>
                    <goto next="intro.vxml"/>
              </catch>
       </form>
</vxml>
```

12.5 Web application server testbridgetransfer.vxml file

Here it is presented the **testbridgetransfer.vxml** file used in these Application Notes

```
<?xml version="1.0" ?>
<vxml version="2.1" xmlns="http://www.w3.org/2001/vxml" xml:lang="en-US" >

	<var name="var1" expr="'tel:'"/>

	<form id="get_number">

		<field name="phone_number">

			<prompt bargein="true">
					<audio src="prompts/TransferGetNumber.wav"/>
			</prompt>

			<grammar src="builtin:dtmf/digits?minlength=1;maxlength=10" />

			<noinput>
				<prompt bargein="false">
					<audio src="prompts/TransferNoNumberSorry.wav"/>
				</prompt>
				<reprompt/>
			</noinput>

		</field>

		<transfer name="bridgetransfer" destexpr="var1 + phone_number"
type="bridge" aai="abcdefg1234567890">

			<prompt bargein="true">
				<audio src="prompts/bridgePerforming.wav"/>
			</prompt>

			    <grammar src="builtin:dtmf/digits" />

			    <filled>
					<if cond="bridgetransfer == 'busy'">
					 <audio src="prompts/lineBusy.wav"/>
						<log> busy </log>
					<elseif cond="bridgetransfer == 'noanswer'"/>
					 <audio src="prompts/noAnswer.wav"/>
						<log> noanswer </log>
					<elseif cond="bridgetransfer == 'network_busy'"/>
					 <audio src="prompts/nwBusy.wav"/>
						<log> network_busy </log>
					<elseif cond="bridgetransfer ==
'near_end_disconnect'"/>
					 <audio src="prompts/nearEndDisc.wav"/>
						<log> near_end_disconnect </log>
					<elseif cond="bridgetransfer == 'unknown'"/>
					 <audio src="prompts/failedUnknown.wav"/>
						<log> unknown </log>
					<elseif cond="bridgetransfer ==
'maxtime_disconnect'"/>
					 <audio src="prompts/maxTimeDisc.wav"/>
						<log> maxtime_disconnect </log>
					<elseif cond="bridgetransfer ==
'network_disconnect'"/>
```

```
                                 <audio src="prompts/nwDisc.wav"/>
                                        <log> network_disconnect </log>
                                <elseif cond="bridgetransfer ==
'far_end_disconnect'"/>
                                 <audio src="prompts/farEndDisconnect.wav"/>
                                        <log> far_end_disconnect </log>
                                </if>

                          <prompt bargein="false">
                                 <audio src="prompts/bridgeThanks.wav"/>
                          </prompt>

                                <goto next="intro.vxml"/>
                        </filled>

                </transfer>

                <catch event="connection.disconnect.hangup">
                        <log> connection.disconnect.hangup </log>
                        <exit />
                </catch>
                <catch event="error.connection.noauthorization">
                        <log> error.connection.noauthorization </log>
                        <goto next="intro.vxml"/>
                </catch>
                <catch event="error.connection.baddestination">
                        <log> error.connection.baddestination </log>
                        <goto next="intro.vxml"/>
                </catch>
                <catch event="error.unsupported.transfer.bridge">
                        <log> error.unsupported.transfer.blind </log>
                        <goto next="intro.vxml"/>
                </catch>
             <catch event="error.unsupported.uri">
                    <log> error.unsupported.uri </log>
                    <goto next="intro.vxml"/>
             </catch>
                <catch event="error.connection.noroute">
                        <log> error.connection.noroute </log>
                        <goto next="intro.vxml"/>
                </catch>
                <catch event="error.connection.noresource">
                        <log> error.connection.noresource </log>
                        <goto next="intro.vxml"/>
                </catch>
        </form>
</vxml>
```

12.6 Web application server testconsulttransfer.vxml file

Here it is presented the **testconsulttransfer.vxml** file used in these Application Notes.

```
<?xml version="1.0" ?>
<vxml version="2.1" xmlns="http://www.w3.org/2001/vxml" xml:lang="en-US" >

	<var name="var1" expr="'tel:'"/>

	<form id="get_number">
		<field name="phone_number">
		<prompt bargein="true">
			<audio src="prompts/TransferGetNumber.wav"/>
		</prompt>

	<grammar src="builtin:dtmf/digits?minlength=1;maxlength=12" />

	  <noinput>
		<prompt bargein="false">
			<audio src="prompts/TransferNoNumberSorry.wav"/>
		</prompt>
		   <reprompt/>
	  </noinput>

	</field>
		<transfer name="consultationtransfer" destexpr="var1 + phone_number"
type="consultation"   aai="tellmeitworks">
			<prompt bargein="false">
				<audio src="prompts/consultPerforming.wav"/>
			</prompt>

			  <filled>
				<if cond="consultationtransfer == 'busy'">
						<audio src="prompts/lineBusy.wav"/>
						<log> busy </log>
					<goto next="intro.vxml"/>
				<elseif cond="consultationtransfer == 'noanswer'"/>
						<audio src="prompts/noAnswer.wav"/>
						<log> noanswer </log>
					<goto next="intro.vxml"/>
				      <elseif cond="consultationtransfer ==
'near_end_disconnect'"/>
						<audio src="prompts/nearEndDisc.wav"/>
					      <log> near_end_disconnect </log>
					<goto next="intro.vxml"/>
				      <elseif cond="consultationtransfer ==
'network_busy'"/>
						<audio src="prompts/nwBusy.wav"/>
						<log> network_busy </log>
					<goto next="intro.vxml"/>
				      <elseif cond="consultationtransfer == 'unknown'"/>
						<audio src="prompts/failedUnknown.wav"/>
						<log> unknown </log>
					<goto next="intro.vxml"/>
				</if>
			</filled>

		</transfer>

		   <catch event="connection.disconnect.hangup">
```

```
                        <log> connection.disconnect.hangup </log>
                        <goto next="intro.vxml"/>
                </catch>

                <catch event="error.connection.noauthorization">
                        <log> connection.disconnect.transfer </log>
                </catch>

                <catch event="error.connection.noauthorization">
                        <log> error.connection.noauthorization </log>
                        <goto next="intro.vxml"/>
                </catch>

                <catch event="error.connection.baddestination">
                        <log> error.connection.baddestination </log>
                        <goto next="intro.vxml"/>
                </catch>

                <catch event="error.connection.noroute">
                        <log> error.connection.noroute </log>
                        <goto next="intro.vxml"/>
                </catch>

                <catch event="error.connection.noresource">
                        <log> error.connection.noresource </log>
                        <goto next="intro.vxml"/>
                </catch>

              <catch event="error.unsupported.uri">
                    <log> error.unsupported.uri </log>
                    <goto next="intro.vxml"/>
              </catch>

                <catch event="error.unsupported.transfer.consultation">
                        <log> error.unsupported.transfer.consultation </log>
                        <goto next="intro.vxml"/>
                </catch>

       </form>
</vxml>
```

Please e-mail any questions or comments pertaining to these Application Notes along with the full title name and filename, located in the lower right corner, directly to the Avaya DevConnect Program at devconnect@avaya.com.