

VoIP Media Gateway Reference Platform

A Flexible and Cost-Effective Way to
Develop a VoIP System



Executive Summary

The VoIP Media Gateway Reference Platform is an integrated white box media gateway designed to introduce Voice over IP (VoIP) application developers and system integrators to the flexibility and media-processing-specific features of a VoIP solution built with powerful modular Dialogic® building blocks, such as Dialogic® Host Media Processing Software and Dialogic® HMP Interface Boards, and Paraxip Gateway software.



Table of Contents

| | |
|--|----|
| A Flexible, Cost-Effective VoIP Solution..... | 2 |
| Audience..... | 2 |
| Overview..... | 2 |
| Sample System Components..... | 2 |
| Paraxip Gateway Deployment Scenarios..... | 2 |
| Architecture..... | 3 |
| Component Installation..... | 4 |
| Setting Parameters..... | 4 |
| Basic Principles..... | 4 |
| Network and Media Device Allocation in Dialogic® Global Call Software..... | 4 |
| Determining How Many Resources Are Needed..... | 6 |
| Setting Paraxip Gateway Parameters..... | 7 |
| Setting Ethernet Connections..... | 7 |
| SipIVRApp Test Application..... | 8 |
| Overview..... | 8 |
| Application Configuration..... | 9 |
| Using a SIP Proxy..... | 9 |
| Appendix A. SipIVRApp Configuration File..... | 11 |
| Appendix B. System Installation Notes..... | 13 |
| Appendix C. Configuring a System Using Other HMP Interface Boards..... | 15 |
| Acronyms..... | 17 |
| For More Information..... | 17 |

A Flexible, Cost-Effective VoIP Solution

The VoIP Media Gateway Reference Platform (MGRP) is a base platform for running telephony applications using the Session Initiation Protocol (SIP). The platform also allows telephony applications to access the PSTN through SIP-PSTN media gateway software developed by Paraxip Technologies.

The MGRP is an easily configurable and cost-effective solution. Because it contains both a Dialogic® HMP Interface Board and Dialogic® Host Media Processing (HMP) Software, the MGRP can work with a wide variety of PSTN protocols and provide flexible media processing, tone generation/detection, and play/record functions.

The MGRP is a turnkey solution for development, interoperability testing, and pilot VoIP deployments. Because of its flexibility, the MGRP can be configured in a wide variety of ways; however, for demonstration purposes, a specific set of components has been chosen and a simple test application (SipIVRApp) will be used. Once the way in which the components work together is understood, developers and system integrators will be able to create more complex configurations to service a wide variety of applications.

Audience

This application note is useful for anyone wishing to understand how the VoIP Media Gateway Reference Platform works. However, developers and system integrators are the primary audience.

Overview

This section describes the components used in the sample system, deployment scenarios for the Paraxip Gateway, system architecture, and component installation.

Sample System Components

The VoIP Media Gateway Reference Platform has a very flexible design. However, in this application note, the focus is on just one configuration for the purposes of demonstration. The basic components used to build the reference platform discussed in this application note are:

- Intel Server Chassis SR2400 server equipped with dual Intel Xeon 3.6 GHz processors
- Microsoft Windows Server® 2003, Standard or Enterprise Edition
- Dialogic® Host Media Processing Software Release 2.0 for Windows®
- Dialogic® DNI/601TEPHMP Digital Network Interface Board
- Paraxip Gateway software version 2.0.1

You can find out more about these components by using the links in the For More Information section.

Paraxip Gateway Deployment Scenarios

A SIP application may be deployed in many ways with the Paraxip Gateway. Here are three options:

1. **External** — Paraxip Gateway runs on a separate server.
2. **Resident SIP telephony application** — SIP-based telephony applications run on the same server as the Paraxip Gateway.
3. **Resident SIP/HMP telephony application** — SIP-based telephony applications, one or more using host media processing, run on the same server as the Paraxip Gateway.

Using one server (options 2 and 3) has advantages over using two servers, one for applications and another for the Paraxip Gateway:

- A one-server solution is generally easier to integrate, deliver, configure, administer, support, and maintain than a two-server solution.
- Using an internal IP interface on a VoIP system results in greater efficiency when moving voice data between an application and the gateway. This is due to decreased use of external network bandwidth, and reduced licensing costs for the low-bit rate coders typically required for over-the-network transmissions.

This application note focuses on the third option — combining Paraxip Gateway software and a SIP-based telephony application using host media processing on one server.

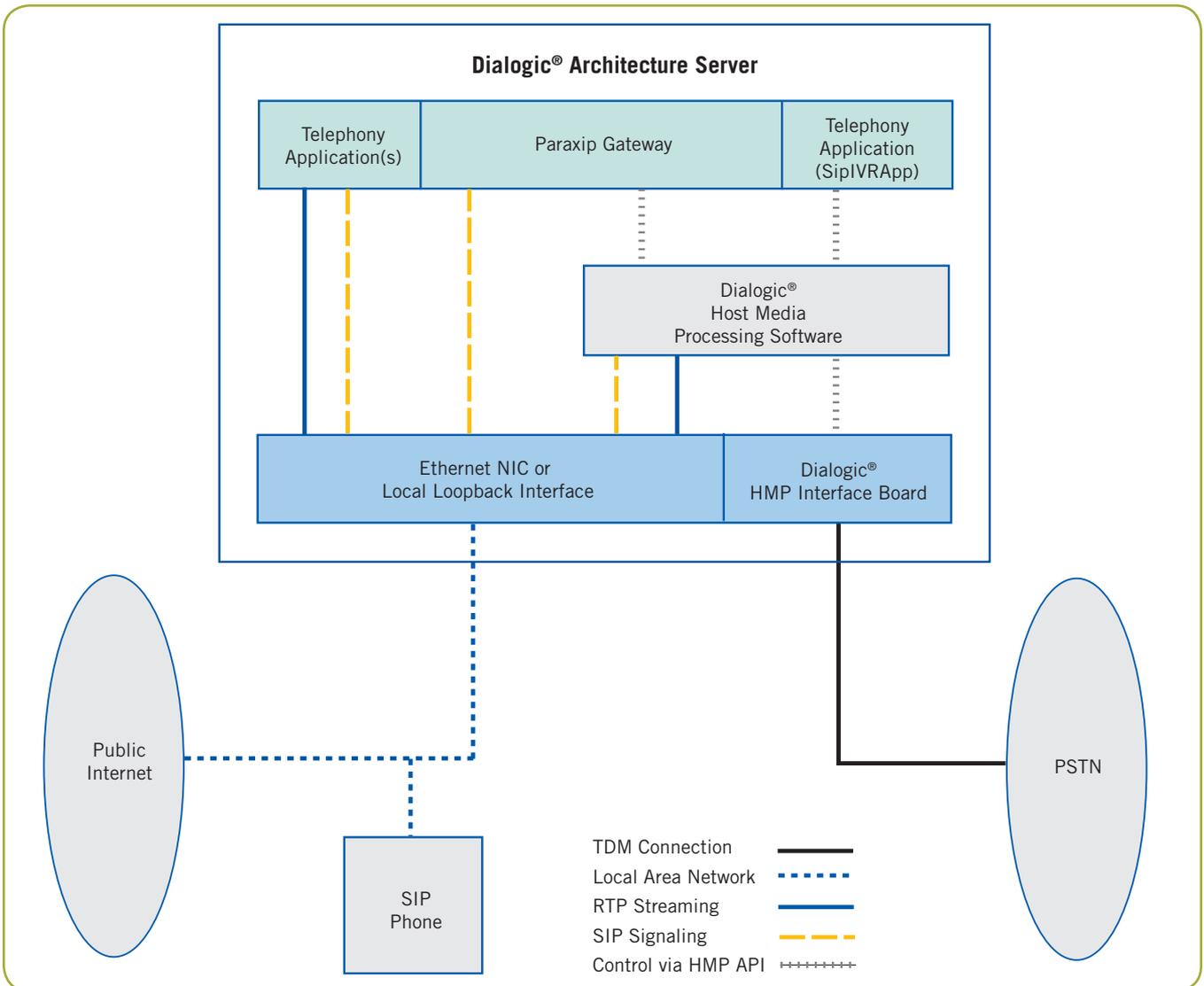


Figure 1. Reference Platform Architecture

Architecture

Figure 1 illustrates the components in the sample system discussed in this application note and how they relate to one another.

Top Layer

The top layer has three types of applications:

- Paraxip Gateway software for signaling and media conversion from ISDN to SIP.
- One or more SIP telephony applications that use Dialogic® HMP Software. SipIVRApp, the test application written for this application note, is of this type.

- SIP telephony applications that do not use Dialogic® HMP Software. None are included in the sample system.

Middle Layer

The middle layer consists of Dialogic® HMP Software and interacts in the following ways with the three applications in the top layer:

- **Telephony Application using Dialogic® HMP Software (SipIVRApp)** — Host media processing provides SIP signaling and RTP streaming. The application uses Dialogic® HMP Software to interface with a Dialogic® HMP Interface Board (in the sample system, a Dialogic® DNI/601TEPHMP board) for PSTN access or the Ethernet Network Interface Card (NIC) for IP calls.

- **Paraxip Gateway** — SIP signaling is enabled by the Paraxip Gateway, which uses an internal SIP stack. Host media processing provides RTP streaming for the gateway software and handles all interaction with the PSTN.
- **Other Telephony Applications** — Telephony applications that do not use Dialogic® HMP Software can also be included in the system for either SIP signaling or RTP streaming. These applications communicate via their own SIP and RTP stacks, using either the Paraxip Gateway for PSTN access or the Ethernet NIC for IP calls.

Bottom Layer

The bottom layer contains the communications interfaces. The local loopback interface allows IP communications within the server itself. The Ethernet NIC extends IP communications to the Internet. The Dialogic® HMP Interface Board (DNI/601TEPHMP) provides a time division multiplex (TDM) interface to the PSTN.

Component Installation

This application note assumes that a VoIP Media Gateway Reference Platform is in place with the following software pre-configured and pre-installed:

- Windows Server® 2003
- Dialogic® Host Media Processing (HMP) Software
- Paraxip Gateway software
- SipIVRApp test application

System integrators and developers who want to build the sample system or any other system based on the VoIP Media Gateway Reference Platform from scratch should consult the notes on component installation in Appendix B. The components must be installed in the specific order described in the appendix.

Setting Parameters

This section discusses some basic principles and then goes into detail about how parameters should be set for the sample system.

Basic Principles

Sharing host media processing resources between the Paraxip Gateway and a co-resident SIP application (in this case, SipIVRApp) is not difficult if certain principles are kept in mind. Two important principles are:

1. Any SIP stacks co-resident on the system must use different ports.
2. Network and media devices must be partitioned so that neither application attempts to use the other's resources.

Port selection for both SipIVRApp and the Paraxip Gateway is done by setting appropriate parameters. SipIVRApp should use the standard SIP port of 5060 so that IP calls trying to reach it can conveniently specify the standard default port setting.

If the default port of 5060 is used, it is not necessary to set the parameter.

If the default port is not used, set the port for SipIVRApp by using the `sip_signaling_port` parameter in the `IP_VIRTBOARD` data structure, which stores configuration and capability information about an IPT board device. Parameters for the device are set before the SIP devices are initialized via Dialogic® Global Call software.

Network and Media Device Allocation in Dialogic® Global Call Software

Network and media device allocation in Dialogic Global Call software is more complicated. The device types most likely to be used in an integrated gateway and media system are:

- **dti** — TDM digital network interface device for T1 or E1
- **ipt** — VoIP signaling device (SIP)
- **ipm** — IP media RTP streaming device
- **dxxx** — Audio device for play and record

The Paraxip Gateway and SipIVRApp do not need all the device types. The Paraxip Gateway uses its own SIP stack and does not need the `ipt` device type available under Global Call software. Because SipIVRApp relies on the Paraxip Gateway for access to the PSTN, it uses no `dti` devices.

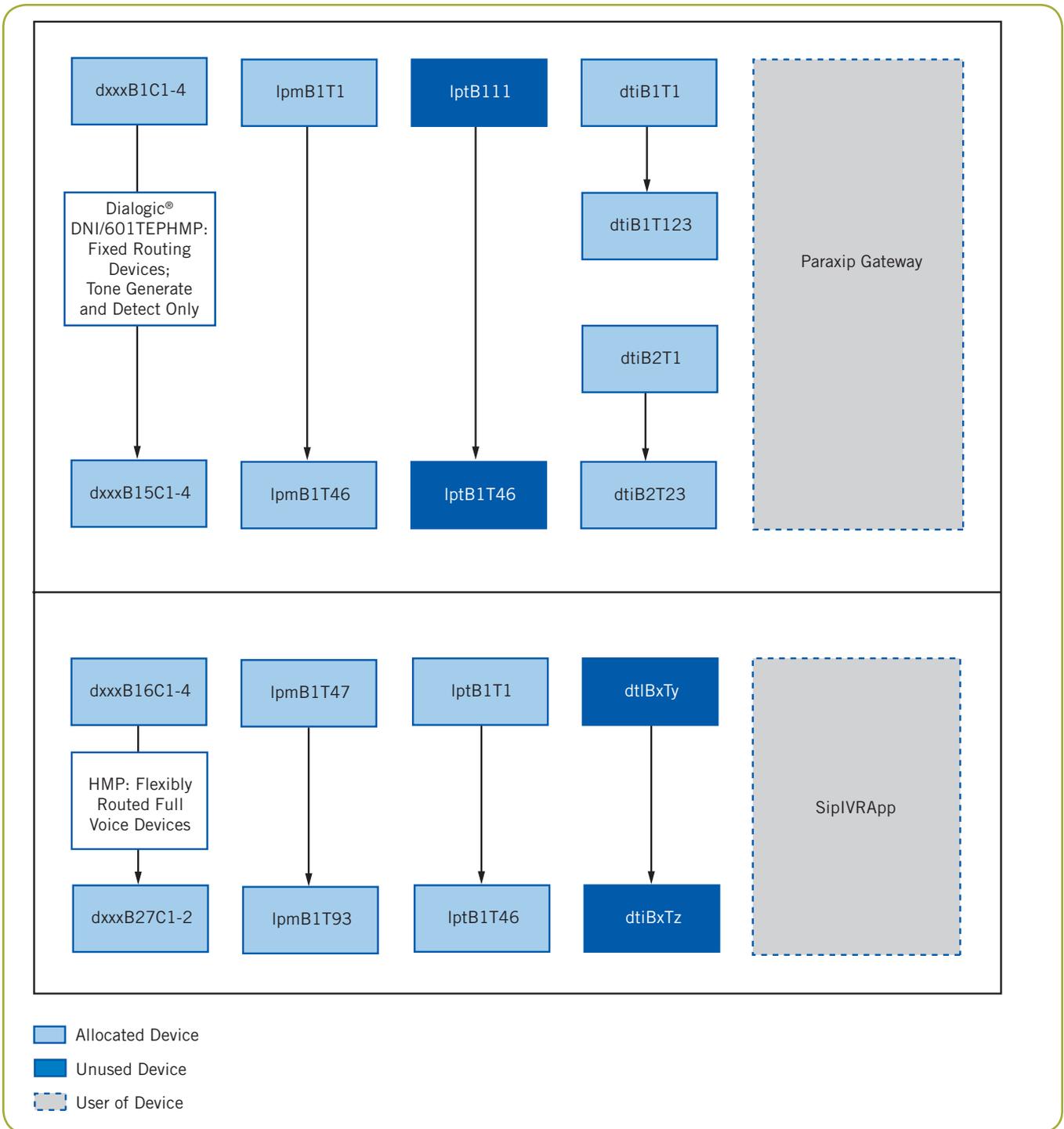


Figure 2. Device Use

Figure 2 shows device use for an integrated environment with:

- Paraxip Gateway software using DNI/601TEPHMP running 5ESS ISDN
- SipIVRApp using Dialogic® HMP Software

Because the DNI/601TEPHMP is a dual T1/E1 board, 46 to 60 TDM channels will be available for use, depending on the protocol chosen. In this example, the Dialogic® HMP Software is provisioned in a ratio of 1:1 (VoIP to TDM), and a configuration of 46 ISDN PRI/T1 channels is assumed. This setup allows all channels used by SipIVRApp to take and receive calls through

the PSTN. If additional VoIP channels are licensed and configured, SIP calls can interact with SipIVRApp independent of the Paraxip Gateway.

In the sample configuration in Figure 2, devices are allocated as described in the following sections identified by device type.

dti

The TDM interface is set up for ISDN over a T1 line. Since ISDN uses one common channel per span for signaling, 23 of the 24 channels are available as bearer channels. Thus, the two spans dt1B1T1-dtiB1T23 and dtiB2T1-dtiB2T23 are used by the Paraxip Gateway for PSTN connectivity.

ipt

The Paraxip Gateway does not use SIP signaling devices, but SipIVRApp opens and uses 46 of them, one for each available ISDN channel. All channels are on a single board (iptB1). While it is possible to divide VoIP channels into “virtual” boards, there is no reason to do so here. Virtual boards are only useful when a group of channels need to share a common attribute, such as an IP address or signaling protocol.

ipm

The Paraxip Gateway and SipIVRApp each use 46 ipm streaming devices. The Paraxip Gateway uses the first 46 (ipmB1T1-ipmB1T46) and SipIVRApp the second 46 (ipmB1T47-ipmB1T92). As with the ipt devices, a single virtual board is used.

dxxx

For an ISDN configuration in the sample system illustrated in this application note, voice devices are not necessary since they are only needed for in-band signaling or if CPA is activated. In this case, they are used to

analyze the tones and voice heard when an outbound call is in progress after which they return a result such as busy, fax, answering machine, or live human voice. SipIVRApp uses neither.

However, the DNI/601TEPHMP has “extra” DSP capacity for on-board tone generation and detection devices that can be used by the Paraxip Gateway for T1 robbed-bit signaling and for CPA. The gateway will automatically open the devices when a DNI/601TEPHMP is detected, and use them if necessary.

Because Dialogic® voice devices are always grouped into 4-channel virtual boards, the implicit voice devices used by the Paraxip Gateway occupy the first 15 virtual boards, yielding a total of 60 tone-only voice resources specified as dxxxB1C1-dxxxB15C4. On the application side, the use of host-based voice devices ranges from dxxxB16C1-dxxxB27C2 to cover the 46 network channels available.

Determining How Many Resources Are Needed

The Dialogic® HMP Software license configuration that is bundled with the VoIP MGRP provides the following resources:

- 120 Voice
- 120 RTP G.711
- 60 Conferencing
- 8 Fax
- 60 enhanced RTP
- 60 Speech Integration
- 60 IP Call Control

Table 1 shows the resources actually needed by the Paraxip Gateway and SipIVRApp test application for the dual T1 ISDN configuration described in this application note. The acronym HMP signifies the Dialogic® HMP Software resources and the acronym DNI signifies DNI/601TEPHMP board resources.

| | Voice G.711 | RTP IP | Call Control TDM | Network Interface |
|-----------------|---------------|--------|------------------|-------------------|
| Paraxip Gateway | 46 DNI | 46 HMP | None | 46 DNI |
| SipIVRApp | 46 HMP | 46 HMP | 46 HMP | None |
| Total Devices | 46 HMP/46 DNI | 92 HMP | 46 HMP | 46 DNI |

Table 1. Resource Use with DNI/601TEPHMP

This resource allocation example can be used as a guide when creating other systems based on the VoIP MGRP.

Setting Paraxip Gateway Parameters

Configuration settings for the Paraxip Gateway are needed in the files `gw.properties` and `routing-rules.xml`. Enabling CPA is also discussed briefly.

gw.properties

Two settings are needed in `gw.properties`: inclusive range of ipm devices and SIP stack port.

Specify the inclusive range of the ipm devices used by the Paraxip Gateway on ipmB1 in this parameter:

```
paraxip.media.dialogic.ipmDeviceRange=1:46
```

Specify the port for the SIP stack used by the Paraxip Gateway in this parameter:

```
paraxip.sip.userAgentPort=5061
```

routing-rules.xml

Here is the section of the `routing-rules.xml` configuration file where settings are required:

```
<resource_group name="PSTN">
  <call_control provider="dialogic">
    <reserved_voice_devices firstBoard="16" lastBoard="45"/>
    <device_group name="default" direction="bidir"
      outbound_hunting_scheme="reverse linear"
      protocol="ISDN" type="digital">
      <range board="1" start="1" end="23"/>
      <range board="2" start="1" end="23"/>
    </device_group>
  </call_control>
</resource_group>
```

Set the “protocol” attribute to ISDN and the “range” tag to the range of the board and channels used. The “reserved_voice_devices” tag instructs the Paraxip Gateway to avoid using any host-based voice devices, making them available for application use.

Enabling CPA

To enable CPA, set the following tag in the PSTN OUT rule used by the Paraxip Gateway in `routing-rules.xml`:

```
<param name="pstn.out.cpa.enable" expr="true"/>
```

The *Paraxip Gateway User's Guide* provides additional information about using CPA with the gateway. CPA is not used in the test application `SipIVRApp`.

Setting Ethernet Connections

As previously mentioned, the Paraxip Gateway and a VoIP telephony application run most efficiently on the same server. Although it is easy to configure the Paraxip Gateway and applications to communicate over an IP network (dedicated or not), the entire system will work more efficiently if it is not distributed and the local loopback interface is used to support RTP voice streaming.

The functionality of a local loopback interface is identical to that of an external Ethernet interface, but it is available only to processes that are running on the same system. The electrical signaling used to represent the data does not need to be physically moved through the extra circuitry of an Ethernet card, and the card's driver software does not need to be run. Voice data transfer takes place in data buffers close to the host processor where it can occur more quickly and efficiently.

Since data transfer on a local loopback interface is very efficient, low bit rate vocoders such as G.723 or G.729 are not necessary. Such vocoders may sometimes be used to reduce network bandwidth, but they also require additional processing power to compress the voice stream. When a vocoder is used, G.711 is chosen for its efficiency and superior voice quality because it takes a direct 64 Kbps sampling of analog voice data.

The numerical address for the local loopback interface on a system is usually 127.0.0.1, although referencing that address is not required and the system's external IP address may be used instead. If the destination address is the same as the system's external address, the local loopback interface will be used to deliver the packets. This simplifies setting up IP addresses for Dialogic® HMP Software, the Paraxip Gateway, and SipIVRApp.

Ethernet addresses should be set to the system's external address in four places:

1. Dialogic® Configuration Manager (DCM) under the Default IP Address tab for the HMP_Software device.
2. gw.properties file of the Paraxip Gateway software. The relevant parameters are:

```
paraxip.net.primaryIPAddress = <address>
paraxip.sip.userAgentIPAddress = <address>
```

3. routing-rules.xml file of the Paraxip Gateway software under the default_sip_out rule as follows:

```
<param name = "sip.out.requestUri"
  expr="sip:<address>:5060"/>
```

4. The configuration file for the test application (SipIVRApp.cfg).

SipIVRApp Test Application

This section begins with an overview of the basic SipIVRApp test application and then provides specific configuration information, which will be the same for each of the three sample configuration files. You can download the files at <http://www.dialogic.com/goto/?10567>.

Overview

SipIVRApp demonstrates how to configure and use a typical SIP telephony application on the same platform as the Paraxip Gateway. SipIVRApp is written to the Global Call software API, using SIP for call control and RTP for media streaming. The application can be configured to accept inbound SIP calls, generate outbound SIP calls, or direct the Paraxip Gateway to initiate a blind transfer to another SIP device.

The SIP channels of SipIVRApp are set to interact with those of the Paraxip Gateway. Inbound calls from the PSTN arrive at the TDM network interface controlled by the gateway and are relayed to the application. Outbound SIP calls from the application are relayed by the gateway to the TDM network interface and PSTN. A call transferred by the application disconnects the application and directs the gateway to call to an alternate SIP endpoint.

The application flow is simple. When a call is answered (inbound) or connected (outbound), a greeting/prompt is played, a short recording made, and the recording replayed.

SipIVRApp was developed under the Microsoft® Visual C++® 7.0 development system with the Visual C++ 7.0 solution files. Visual C++ must be installed on the system if further development or internal changes are to be made on the application. SipIVRApp may be configured and will run on the system without Visual C++.

Application Configuration

Three sample configuration files are supplied based on a generic SipIVRApp.cfg file:

- **SipIVRApp-in.cfg** — Opens half of the IP channels available (the other half are used by the Paraxip Gateway) and puts them in a wait-for-call state. When a call comes in, it is answered and goes through the application flow. When the flow is complete, the application waits for a far-end hangup before resetting the channel.
- **SipIVRApp-out.cfg** — Opens a single IP channel for use. When the channel is opened, the application makes a call to the specified extension (default 0000) through the Paraxip Gateway. When the call is connected, the application goes through the application flow. When the flow is complete, the application waits for a far-end hangup before resetting the channel.
- **SipIVRApp-xfer.cfg** — Opens a single IP channel for use, and puts it in a wait-for-call state. When a call comes in, it is answered and goes through the application flow. When the flow is complete, the Paraxip Gateway is directed to transfer the PSTN call coming in through it to the specified SIP device. The application then resets for another incoming call.

The MGRP provides batch files on its desktop that will automatically copy any of the three configurations described above and run the application. The same naming conventions are followed throughout.

The scenarios in the sample configuration files provide examples of common application behavior. Other scenarios are possible, and IP addressing and device specification may need to be altered if other scenarios are used. The alterations can be done with a text editor. The parameters in the files are given as name/value pairs, separated by an “=” sign. A line beginning with the “#” character is a comment. Each line must end with a carriage return.

The configuration files are divided into three sections:

- **IP Addressing** — Specifies IP addresses, SIP users, and ports
- **Device Allocation** — Specifies number of channels used by the application and how they are numbered. This section determines which network and voice

devices the application uses. In conjunction with the Paraxip Gateway configuration files, it ensures that there are no conflicts in using Dialogic® devices.

- **Application Behavior** — Handles inbound calls and outbound calls, SIP Proxy registration, single call, multiple calls, and application call flow.

Details on the parameters in each section are contained in a basic version of the configuration file, which is provided in Appendix A.

Using a SIP Proxy

An external SIP proxy or application server may be used to monitor calls within an application’s domain and make intelligent decisions as to the processing of the calls. For example, a SIP call whose destination is an application that has no need for PSTN access may be directed to an IP-only server, while another call may use a service that will eventually transfer it to the PSTN.

Normally, SIP proxies are used when applications and services are distributed over multiple servers rather than with the kind of integrated platform on which this application note focuses. However, because both the Paraxip Gateway and the SipIVRApp test applications support SIP proxies, how to configure them to do so will be discussed here.

For both a test and production proxy, the Open SIP Express Router (OpenSER) is a possibility. OpenSER is an open source, full-featured, multi-purpose SIP server that functions as a SIP proxy in its default configuration. It can be found at <http://openser.org/> and currently runs only on Linux.

A common way a SIP proxy is used with a gateway is to route all calls originating from the PSTN to the proxy, where a decision on their ultimate destination is made. Although it is possible to set up sophisticated routing rules within the Paraxip Gateway itself, it may be preferable to keep enterprise-wide routing information in a central repository with a SIP proxy front end. In such a case, it is only necessary to configure the Paraxip Gateway to direct all its outbound SIP calls to the proxy. This may be done by setting the following parameter in the general Paraxip configuration file gw.properties:

```
paraxip.sip.localOutboundProxyURI=sip:<IP address>
```

To enable full communication with a SIP proxy for calls between an application and the Paraxip Gateway in both directions, additional modifications must be made to two configuration files.

First, an additional parameter in gw.properties must be set to enable registration with the proxy. The parameter contains the full path and name of the XML configuration file, which holds the proxy's address and the SIP users to be registered:

```
paraxip.sip.client.Registration.=C:\Program Files\Paraxip Gateway\  
config\sip-client-registration.xml
```

Here is an example of how the IP address of the SIP proxy (registrar) should be set in the XML file:

```
<registrar>  
  <uri>sip:192.168.1.100</uri>  
  <retry_interval_sec>300</retry_interval_sec>  
</registrar>
```

Each SIP user for which the Paraxip Gateway may receive an incoming SIP call must also be registered in the XML file. The SIP user's Address of Record and URI usually contains an extension number. The IP Address in the URI is the address of the Paraxip Gateway, along with the port on which it is listening. An entry will be similar to the following:

```
<registration_entry>  
  <address_of_record_uri>  
    sip:201@mailserver.intel.com  
  </address_of_record_uri>  
  <contact>  
    <uri>sip:201@192.168.1.107:5061</uri>  
  </contact>  
</registration_entry>
```

The SIP application must also be configured to register with the proxy. This is done by setting the following three parameters in SipIVRApp.cfg as follows:

```
sipRegister=true  
sipProxyAddress = 192.168.1.100  
sipProxyPort = 5060
```

On application startup, SIP registration is initiated, and all outbound SIP messages will be directed to the proxy.

Appendix A. SipIVRApp Configuration File

```
#####
# SIP Addressing Section
#####

# SIP user name for application.
localUser = SipIVRApp

# Local IP address to use.
localAddress = 192.168.1.104

# UPD port this app will use for SIP signaling (Usually 5060).
localAppSipPort = 5060

# Extension/SIP user name used at the final destination (in this
# case CSF Gateway app).
# Note: Name must be numerical, as it is used to generate sequential
# extensions if multiple outbound channels are configured. Extensions
# are 4 digits long, left padded with zeroes. The value specified here
# will be the first extension used on the first port. Extensions
# will increment by one for each additional port.
gatewayUser = 0000

# Calls destined for the Paraxip Gateway use the system's external IP
# address, but are routed through its local loopback interface.
gatewayAddress = 192.168.1.104

# Port for Paraxip Gateway (usually 5061).
gatewayPort = 5061

# SIP proxy address. Used only if sipRegister below is true.
sipProxyAddress = 192.168.1.100

# SIP proxy port. Used only if sipRegister below is true.
sipProxyPort = 5060

# IP address of the transferee.
# Used only if transferCall below is true.
transferAddress = 192.168.1.23

# SIP user name for SIP transfer.
# Used only if transferCall below is true.
transferUser = 23

# Port for SIP transfer.
# Used only if transferCall below is true.
transferPort = 5060
```

```
#####
# Device Allocation Section
#####

# Total number of IP channels actively used when the application runs.
# Determines total number of channels opened and used.
# All channels will be on-board 1 - iptB1/ipmB1
channelsUsed = 46

# Start channel on the IP network board. Channels up to this
# one are claimed by the Paraxip Gateway.
firstChannel = 47

# First virtual 4 channel voice (Vox) board on the system that the
# application will use. Number of boards used depends on network
# channels in use. One Vox device per channel is allocated.
firstVoxBoard = 28

#####
# Application Behavior Section
#####

# Register/do not register with a SIP proxy.
sipRegister = false

# If true, outbound call(s) done on app startup. If false, app
# waits for inbound call(s) on startup.
makecallOnStart = false

# If true, the application simply plays a file before moving on to the
# next call. If false, it goes through a play/record/playback cycle.
appPlayOnly = true

# If true, the app remains running, and either makes sequential
# outbound calls on all available channels, or resets and waits for an
# inbound call after hang-up on the current call.
# If false, the app exits after a single inbound or outbound call.
continuousCalling = true

# If true, application will do a blind transfer of the call to the
# specified SIP user and address after playing the welcome prompt.
# May be used with either inbound or outbound calls.
# Note that control may be an issue if system is doing a transfer
# with an outbound call and continuousCalling is set to true.
# If false, the app will either simply exit or reset after the inbound
# session is complete.
transferCall = false
```

Appendix B. System Installation Notes

The information in Appendix B is provided for those building a complete system from scratch based on the VoIP Media Gateway Reference Platform.

Procedure for Installing Components

This appendix lists the hardware and software components to be installed and discusses potential problems and any non-standard installation steps. This appendix is not intended as a detailed installation procedure, and the installation guide for each component should be reviewed carefully before installation.

The software and hardware should be installed in the order given below.

Installation 1: Operating System

Install Microsoft Windows Server® 2003, Standard or Enterprise Edition, with Service Pack 1.

Ethernet drivers are not installed as part of the operating system installation on the Intel Server Chassis SR2400. The drivers can be found on the Intel Server Deployment Toolkit CD and should be installed as follows:

1. Insert CD in drive.
2. Go to Control Panel and select Add Hardware Wizard.
3. Answer yes to the question “Is the hardware connected?”
4. Choose Ethernet Controller from the Installed Hardware list.
5. Answer yes to the question “Can Windows connect to Windows Update and search for the software?”
6. Select “Install the software automatically.”

The Intel PRO/1000MT DualPort Network Connection will be installed. Then configure the Ethernet controller’s IP parameters normally. The path to the parameter settings is:

Control Panel -> Network Connection -> Local Area Connection

Note that the Paraxip Gateway only supports the use of a single Ethernet port.

Installation 2: HMP Interface Board

Install the Dialogic® DNI/601TEPHMP in an unused PCI-Universal slot.

Installation 3: Host Media Processing Software

Install Dialogic® HMP Software Release 2.0 for Windows®. The following packages must be selected:

- Core Runtime Package
- License Package
- Circuit Connectivity Runtime Package — Dialogic® DNI boards

Be sure that you do not select the DSI boards.

If your application involves automatic speech recognition (ASR) or in-band, non-T.38 fax, comfort noise generation (non-linear processing [NLP]) should be disabled in echo cancellation (EC) for optimum ASR performance.

To disable NLP on the DNI/601TEPHMP, a configuration file change is required because EC is done with on-board DSPs. See the *Dialogic® Host Media Processing Software Release 2.0 for Windows Release Update* for more information.

The following parameters must be added to the configuration file generated for the network protocol being used:

```
SetParm=0x2c00, 0x1 ! Echo Cancellation (0x1 (enable-default), 0x0 (disable))
SetParm=0x2c01, 0x0 ! Enable Echo Canceller NLP (0x1 (enable-default), 0x0
(disable))
```

After the file is edited, run `fcdfgen` to produce `fed` and `pcd` files before starting the system.

If you are using the Dialogic® DNI/300TEPHMP Digital Network Interface Board or Dialogic® DNI/1200TEPHMP Digital Network Interface Board, note that EC is done in host software. To set NLP, see the section on `ec_setpartm` in the *Continuous Speech Processing API for Linux and Windows Operating Systems Library Reference*.

Installation 4: Paraxip Gateway Software

Refer to the `ReadMeFirst.txt` file distributed with the Paraxip Gateway. For further information, see the *Paraxip Gateway – HMP Edition Installation Guide*.

When installing the Paraxip Gateway, make these selections:

- **Telephony Configuration:** ISDN 46 ports
- **Default SIP URL to contact on incoming PSTN calls:** `sip:localhost:5060`
- **Media Provider:** IP Media Library

Installation 5: Applications

Unzip `SipIVRApp.zip` in a convenient location and rename the configuration file you would like to use to `SipIVRApp.cfg`.

Verify Installed Components

Each component should be verified after it is installed to ensure that it is functioning properly before the next component is installed. This section contains a list of tools that can help.

Host Media Processing Software

After Dialogic® HMP Software is installed, the proper functioning of at least one VoIP channel in HMP should be checked. To do this, you can use the `IPMediaServer` demo, which is located in

`C:\Program Files\Dialogic\HMP\demos\IPMediaServer\Release`

Once the demo is installed, do the following:

1. Edit the configuration file (`IPMediaServer.cfg`) to change the protocol to SIP and `DTMFType` to `rfc2833`.
2. Start the server.
3. Use a SIP phone to dial into the application. You can use either a SIP softphone with a sound card on another system or a standalone hardware-based SIP phone.

Paraxip Gateway Software

The `ReadMeFirst.txt` file contains instructions for verifying and testing the Paraxip Gateway. Particularly useful is a web-based utility that displays the operating status of the gateway. When the Paraxip service is running, it can be reached with a web browser at <http://localhost:7777>.

Ethernet Monitoring Utility

A valuable tool for debugging IP protocols is `Ethereal`, an open source, GUI-based Ethernet monitoring utility available as a free download at <http://www.ethereal.com>. The VoIP protocols SIP and RTP are included.

An enhanced version of `Ethereal` for Windows can be downloaded from `Openxtra` at <http://www.openxtra.co.uk/downloads/ethereal-download.php>. Instructions for installing and using the utility can also be found on the download website, and an introductory paper on its use in debugging VoIP applications is available at <http://www.dialogic.com/products/applnote.htm>.

System Operation

On Windows® operating systems, IP traffic on the local loopback interface is not visible through Ethereal. A convenient way to view SIP messages is through the Paraxip Gateway log file when verifying or debugging system operation. See the *Paraxip Gateway User's Guide* for instructions on how to enable the logging of SIP messages sent and received by the Paraxip Gateway

Appendix C. Configuring a System Using Other HMP Interface Boards

This appendix discusses the media resources needed for configuring a system using the Dialogic® DNI/300TEPHMP Digital Network Interface Board and Dialogic® DNI/1200TEPHMP and the settings needed in Paraxip Gateway software configuration files for them.

Required Resources

Configuring a system with the DNI/300TEPHMP and DNI/1200TEPHMP is a straightforward process because neither board contains the on-board tone generation and detection resources found on the DNI/601TEPHMP.

All media resources used will originate from Dialogic® HMP Software, making voice device naming simpler. Tables 2 and 3 below are similar to Table 1 provided for DNI/601TEPHMP. Explanations for the device and resource names are also found in this section.

The configuration assumes ISDN with 23 bearer channels per span. We also assume the Paraxip Gateway software has CPA enabled and requires voice resources.

| | Voice | G.711 RTP | IP Call Control | TDM Network Interface |
|------------------------|--------------------|-----------------------|----------------------|-----------------------|
| Paraxip Gateway | dxxxB1- dxxxB6 | ipmB1C1- ipmB1C23 | None | dtiB1T1- dtiB1T23 |
| SipIVRApp | dxxxB7- dxxxB12 | ipmB1C24- ipmB1C46 | iptB1T1- iptB1T23 | None |
| Total Devices | 48 HMP | 46 HMP | 23 HMP | 23 DNI |

Table 2. Resource Use with DNI/300TEPHMP

| | Voice | G.711 RTP | IP Call Control | TDM Network Interface |
|------------------------|---------------------|------------------------|---------------------|-----------------------|
| Paraxip Gateway | dxxxB1- dxxxB23 | ipmB1C1- ipmB1C92 | None | dtiB1T1- dtiB1T92 |
| SipIVRApp | dxxxB24- dxxxB46 | ipmB1C93- ipmB1C184 | ptB1T1- iptB1T92 | None |
| Total Devices | 184 HMP | 184 HMP | 92 HMP | 92 DNI |

Table 3. Resource Use with DNI/1200TEPHMP

Paraxip Gateway Configuration Settings

The PSTN resource group would be set up as follows in the Paraxip Gateway software configuration file routing-rules.xml for a DNI/1200TEPHMP:

```
<resource_group name="PSTN">
  <call_control provider="dialogic">
    <reserved_voice_devices firstBoard="24" lastBoard="47"/>
    <device_group name="default" direction="bidir"
      outbound_hunting_scheme="reverse linear" protocol="ISDN"
      type="digital">
      <range board="1" start="1" end="23"/>
      <range board="2" start="1" end="23"/>
      <range board="3" start="1" end="23"/>
      <range board="4" start="1" end="23"/>
    </device_group>
    <voice_devices firstBoard="1" lastBoard="23"/>
  </call_control>
</resource_group>
```

This configuration is similar to that given for the DNI/601TEPHMP earlier, but in this example the first 92 voice devices (23 four-channel boards) are reserved for the Paraxip Gateway and the second 92 for the application.

In the Paraxip file gw.properties, specify the inclusive range of the ipm devices on ipmB1 used by the gateway:

```
paraxip.media.dialogic.ipmDeviceRange=1:92
```

Acronyms

| | |
|---------|-----------------------------------|
| API | Application Programming Interface |
| ASR | Automatic Speech Recognition |
| CPA | Call Progress Analysis |
| DCM | Dialogic® Configuration Manager |
| DSP | Digital Signal Processor |
| EC | Echo Cancellation |
| HMP | Host Media Processing |
| IP | Internet Protocol |
| MGRP | Media Gateway Reference Platform |
| NLP | Non-Linear Processing |
| OpenSER | Open SIP Express Router |
| PSTN | Public Switched Telephone Network |
| RTP | Real Time Protocol |
| SIP | Session Initiation Protocol |
| TDM | Time Division Multiplexing |
| URI | Uniform Resource Identifier |
| VoIP | Voice over Internet Protocol |
| XML | eXtensible Markup Language |

For More Information

Intel Server Chassis SR2400

<http://www.intel.com/design/servers/chassis/sr2400/>

Dialogic® Host Media Processing Software Release 2.0 for Windows®

<http://www.dialogic.com/support/helpweb/dxall/HMPWin/hmp20/default.htm>

Dialogic® HMP Interface Boards

http://www.dialogic.com/products/ip_enabled/hmp_enabled_boards.htm

Paraxip Gateway

http://www.paraxip.com/open_gateway.htm

Ethereal

<http://www.ethereal.com/>

Openxtra Ethereal

<http://www.openxtra.co.uk/products/ethereal-xtra.php>

Open SIP Express Router (OpenSER)

<http://openser.org>

To learn more about Dialogic® products, go to www.dialogic.com.

Dialogic Corporation

9800 Cavendish Blvd., 5th floor
Montreal, Quebec
CANADA H4M 2V9

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH PRODUCTS OF DIALOGIC CORPORATION OR ITS SUBSIDIARIES ("DIALOGIC"). NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN A SIGNED AGREEMENT BETWEEN YOU AND DIALOGIC, DIALOGIC ASSUMES NO LIABILITY WHATSOEVER, AND DIALOGIC DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF DIALOGIC® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT OF A THIRD PARTY.

Dialogic products are not intended for use in medical, life saving, life sustaining, critical control or safety systems, or in nuclear facility applications.

Dialogic may make changes to specifications, product descriptions, and plans at any time, without notice.

Dialogic is a registered trademark of Dialogic Corporation. Dialogic's trademarks may be used publicly only with permission from Dialogic. Such permission may only be granted by Dialogic's legal department at 9800 Cavendish Blvd., 5th Floor, Montreal, Quebec, Canada H4M 2V9. Any authorized use of Dialogic's trademarks will be subject to full respect of the trademark guidelines published by Dialogic from time to time and any use of Dialogic's trademarks requires proper acknowledgment.

Windows, Windows Server, and Visual C++ are registered trademarks of Microsoft Corporation in the United States and/or other countries. Other names of actual companies and products mentioned herein are the trademarks of their respective owners. Dialogic encourages all users of its products to procure all necessary intellectual property licenses required to implement their concepts or applications, which licenses may vary from country to country.

This document discusses one or more open source products, systems and releases. Dialogic is neither responsible for your decision to use open source in connection with Dialogic products including without limitation those referred to herein, nor is Dialogic responsible for any present or future effects such usage might have, including without limitation effects on your products, business, or intellectual property rights