

Troubleshoot Legacy Voice Mail Integration with Cisco CallManager

Document ID: 18779

Contents

Introduction

Prerequisites

- Requirements
- Components Used
- Conventions

SMDI Integration with Cisco CallManager

- Direct Call Format
- Troubleshoot a Direct Call
- Direct Call Trace from CMI
- Forwarded Call Format
- Troubleshoot a Forwarded Call
- Forwarded Call Trace from Cisco Messaging Interface
- MWI Format
- Troubleshoot MWI
- MWI Traces from Cisco Messaging Interface
- Known Issues

Cisco DPA 7630/7610

- Troubleshoot Calls to Voice Mail through a Cisco DPA 7630/7610
- Troubleshoot MWI with a Cisco DPA 7630/7610

Related Information

Introduction

This document troubleshoots the integration of legacy voice messaging systems with Cisco CallManager that utilize Simplified Message Desk Interface (SMDI) and Digital PBX Adaptor (DPA) 7630/7610.

Prerequisites

Requirements

Readers of this document should have knowledge of these topics:

- Cisco CallManager 3.x or 4.x
- Telcordia GR 283–core Issue 2 (SMDI)
- Octel Voice Mail, if applicable (DPA –7630/7610)

Components Used

The information in this document is based on these software and hardware versions:

- Cisco CallManager 3.x and 4.x
- Cisco MCS–7835

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

Refer to the Cisco Technical Tips Conventions for more information on document conventions.

SMDI Integration with Cisco CallManager

The Cisco Messaging Interface is a service within Cisco CallManager that allows legacy voice mail platforms to connect to Cisco CallManager with the use of SMDI. The Cisco Messaging Interface provides information to a legacy voice mail system about the called and calling party, the reason why the call is presented, and which port the voice mail system should expect the call on. This allows the voice mail system to answer the call correctly.

There are two call types with SMDI:

- Direct call
- Forwarded call

Direct Call Format

The direct call SMDI format is:

- `<CR><LF>MDXXXLLLLT<0x20>YYYY<0x20><CR><LF><^Y>`

<CR>	Carriage Return
<LF>	Line Feed
MDXXX	Message Desk. This is a 3 digit field, usually 001
LLLL	Logical Terminal Number (0001 ? 4096)
T	Reason Code – D equals Direct Call
<0x20>	Space
YYYY	Calling Party Number
<0x20>	Space
<CR>	Carriage Return
<LF>	Line Feed
<^Y>	End of Medium

This is an example of a direct call from extension 2000. It was presented to the voice messaging system on LTN 0002, or port 2.

- `<CR><LF>MD0010002D<0x20>2000<0x20><CR><LF><^Y>`

Troubleshoot a Direct Call

When a caller places a direct call, the caller expects to be prompted to enter their password. In order for that to

occur, the gateway, route list and route group to the voice mail system must be configured as Configuring Cisco CallManager 3.0(x) for Integration to Voice Mail Systems via SMDI describes. The most common reason for a direct call to fail is a misconfiguration between either the Message Desk or the Logical Terminal Number (LTN). The Cisco Messaging Interface uses the Message Desk of 001 by default. If you have been given another message desk by either a Private Branch Exchange (PBX) vendor or a central office (CO), you must specify that in the Cisco Messaging Interface configuration.

MessageDeskNumber*	1	1
MwiSearchSpace		

If the MessageDeskNumber is correct, verify that the LTN is correct. On the Cisco CallManager side, the LTN equates to the ports you configured in the route group. The first port is LTN 1, the second port is LTN 2, and so on. This is why it is essential to configure the Port and Order drop-downs as Configuring Cisco CallManager 3.0(x) for Integration to Voice Mail Systems via SMDI describes. If all ports are chosen, the LTN generated by the Cisco Messaging Interface is LTN 1. This results in failed integration. On the voice mail side, it is essential to ensure that the cabling is correct between the gateway and voice mail ports. Since the SMDI message indicates which port (LTN) the call is being presented to the voice mail system on, all Layer 1 issues have to be resolved.

A second common reason why a direct call to voice mail fails is a misconfiguration of the VoiceMailPartition field in the Cisco Messaging Interface.

VoiceMailDn	3500
VoiceMailPartition	voicemail

The VoiceMailPartition field needs to contain the name of the partition that is assigned to the route pattern that points to the route list for voice mail. If this information is incorrect, the intercept for the VoiceMailDn is not triggered and no SMDI message is generated. Consequently, the voice mail system does not have the proper information to process the call as desired, and answers with a generic greeting.

Verify that the SMDI link is **UP** on the voice mail side, if your voice mail system relies upon this. Cisco CallManager runs on a standard PC, which means it has nine pins that are actually wired to the serial port. However, Cisco Messaging Interface only uses three of the RS-232 lines: TD, RD, and GND. There are two outbound control lines: RTS and DTR. These are both asserted when the port is opened, but Cisco Messaging Interface does not care if the lines are ignored or not. There are four incoming control lines: DCD, CTS, DSR, and RI. All of these lines are also ignored.

Direct Call Trace from CMI

For troubleshooting issues with SMDI, examine the Cisco Messaging Interface trace files located in the C:/Program Files/Cisco/Trace/CMI directory.

```
19:27:58.578 Cisco Messaging Interface| CMISsapiClient::RecvSsCallInfoResMsg()  
Received RecvSsCallInfoResMsg userdata = 5996360, Key = 7864, DSL2 = 2,  
calledparty =3500, callingparty =2000  
19:27:58.578 Cisco Messaging Interface| CMISsapiClient::RecvSsCallInfoResMsg()  
Direct Call port - 2, callingparty -2000  
19:27:58.578 Cisco Messaging Interface|-->CMISsapiClient::SendDirectCall()  
19:27:58.578 Cisco Messaging Interface| CMISsapiClient::SendDirectCall()
```

This is an example of a direct call from extension 2000. It was presented to the voice messaging system on LTN 0002, or port 2.

- Send direct call : [`<CR><LF>MD0010002D<0x20>2000<0x20><CR><LF><^Y>`]

Once you validate that the Cisco Messaging Interface intercepts the VoiceMailDn and generates SMDI messages, you can work with the voice mail vendor to ensure that the message is received. If the voice mail system does not receive any SMDI messages, even though the Cisco Messaging Interface generates them, a PC with hyper terminal can be used to demonstrate whether they leave the Cisco CallManager server.

Forwarded Call Format

Cisco CallManager 3.0 supports the forward reason code of A for All Calls forward. In Cisco CallManager 3.1 and later, the reason codes B and N are added for Busy and No Answer, respectively.

The forwarded call format is:

- `<CR><LF>MDXXXLLLLTDDDDDDD<0x20>CCCCCCC<0x20><CR><LF><^Y>`

<code><CR></code>	Carriage Return
<code><LF></code>	Line Feed
<code>MDXXX</code>	Message Desk. Number xxx is normally 001
<code>LLLL</code>	Logical Terminal Number (0001 – 4096)
<code>T</code>	Call Type. A is All Calls Forwarded (supported by CM 3.0), B is Busy Forward (CM 3.1), N is for Ring No Answer (CM 3.1)
<code>DDDDDDD</code>	Called Party
<code><0x20></code>	Space
<code>CCCCCCC</code>	Calling Party
<code><0x20></code>	Space
<code><CR></code>	Carriage Return
<code><LF></code>	Line Feed
<code></code>	End of Medium

This is an example of a forwarded call from extension 2000 to extension 2001. The call is sent to voice mail because the called party's phone is in a Call Forward All state. The call is presented to the voice messaging system on LTN 0002, or port 2.

- [`<CR><LF>MD0010002A2001<0x20>2000<0x20><CR><LF><^Y>`]

Troubleshoot a Forwarded Call

When a caller places a call to another party and is forwarded to voice mail, the expectation is that they receive the called party's greeting. In order for that to occur, gateway to the voice mail system must be configured as described in Configuring Cisco CallManager 3.0(x) for Integration to Voice Mail Systems via SMDI. Ensure that direct calls work before you try to troubleshoot forwarded calls.

When you troubleshoot forwarded calls, check to ensure that the legacy voice mail system supports the forwarded reason code. Some voice mail systems allow for the programming of the various reason codes. In some cases, where the Cisco CallManager was deployed before version 3.1, the voice mail system may not have been programmed to accept the Busy and No Answer forwarded reason code, since Cisco CallManager did not support them at that time.

Forwarded Call Trace from Cisco Messaging Interface

When you troubleshoot issues with SMDI, examine the Cisco Messaging Interface trace files located in the C:/Program Files/Cisco/Trace/CMI directory.

```
18:33:28.187 Cisco Messaging Interface| CMISsapiClient::RecvSsCallInfoResMsg()
Received RecvSsCallInfoResMsg fOriginalCdpn = 2001, fCgpn = 2000 fCallingPattern
= 2000
18:33:28.187 Cisco Messaging Interface| CMISsapiClient::RecvSsCallInfoResMsg()
Received RecvSsCallInfoResMsg userdata = 5996336, Key = 7864, DSL2 = 2,
calledparty = 2001, callingparty = 2000
18:33:28.187 Cisco Messaging Interface| CMISsapiClient::RecvSsCallInfoResMsg()
Forwarded Call port - 2, calledparty - 2001, callingparty - 2000
18:33:28.187 Cisco Messaging Interface|-->CMISsapiClient::SendCallForwardAll()
18:33:28.187 Cisco Messaging Interface| CMISsapiClient::SendCallForwardAll()
Send call forward all: [<CR><LF>MD0010002A2001<0x20>2000<0x20><CR><LF><^Y>]
18:33:28.187 Cisco Messaging Interface|-->CMISerialWorker::SendBuffer()
18:33:28.187 Cisco Messaging Interface| CMISerialWorker::SendBuffer()
Send Buffer - <CR><LF>MD0010002A2001<0x20>2000<0x20><CR><LF><^Y>
18:33:28.197 Cisco Messaging Interface|<--CMISerialWorker::SendBuffer()
18:33:28.197 Cisco Messaging Interface|<--CMISsapiClient::SendCallForwardAll()
```

This is an example of a forwarded call from extension 2000 to extension 2001. The call is sent to voice mail because the called party's phone is in a Call Forward All state. The call is presented to the voice messaging system on LTN 0002, or port 2.

Once you validate that the Cisco Messaging Interface intercepts the VoiceMailDn and generates SMDI messages, you can work with the voice mail vendor to ensure that the message is received. If the voice mail system does not receive any SMDI messages, even though the Cisco Messaging Interface generates them, a PC with hyper-terminal can be used to demonstrate whether they are leaving the Cisco CallManager server.

MWI Format

The format for message waiting on and off commands is:

MWI On: OP:MWI<0x20>XXXX!<EOT>

OP:MWI	Operate Message Waiting Indicator
<0x20>	Space
XXXX	Extension number
<EOT>	End of Transmission

OP:MWI<0x20>2001!<EOT>

A request to turn on message waiting for extension 2001.

MWI Off: RMV:MWI<0x20>XXXX!<EOT>

RMV:MWI	Remove Message Waiting Indicator
<0x20>	Space
XXXX	Extension number
<EOT>	End of Transmission

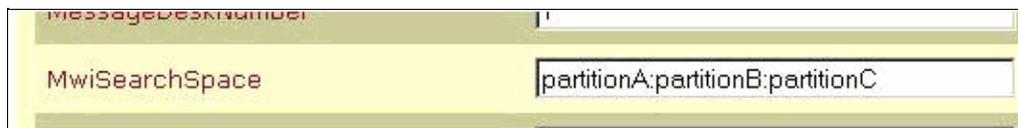
RMV:MWI<0x20>2001!<EOT>

A request to turn off message waiting for extension 2001.

Troubleshoot MWI

When you troubleshoot MWI issues with SMDI, the first step is determine whether or not the request is received by Cisco CallManager. The Cisco Messaging Interface trace files located at C:/Program Files/Cisco/Trace/CMI reveal whether or not the SMDI message is sent by the voice mail system and received by Cisco CallManager.

If the request is received by the Cisco CallManager, but MWI does not work correctly, verify that the MWI SearchSpace field in the Cisco Messaging Interface contains the partition that belongs to the phone whose lamp is being requested to be turned on/off. The MWISearchSpace field needs to contain the names of the partitions separated by a colon. This field is case sensitive.



MWI Traces from Cisco Messaging Interface

MWI On

```
18:33:47.846 Cisco Messaging Interface| CMISerialWorker::SerialThread()
Saw EOT, inbound message: [OP:MWI<0x20>2001!<EOT>]
18:33:47.846 Cisco Messaging Interface|-->CMISsapiClient::OnLampOn()
18:33:47.846 Cisco Messaging Interface| CMISsapiClient::OnLampOn()
Lamp On - 2001
18:33:47.846 Cisco Messaging Interface| CMISsapiClient::OnLampOn()
Processed number = 2001
18:33:47.906 Cisco Messaging Interface|<--CMISsapiClient::OnLampOn()
```

MWI Off

```
18:33:47.846 Cisco Messaging Interface| CMISerialWorker::SerialThread()
Saw EOT, inbound message: [RMV:MWI<0x20>2001!<EOT>]
18:33:47.846 Cisco Messaging Interface|-->CMISsapiClient::OnLampOff()
18:33:47.846 Cisco Messaging Interface| CMISsapiClient::OnLampOff()
Lamp Off - 2001
18:33:47.846 Cisco Messaging Interface| CMISsapiClient::OnLampOff()
Processed number = 2001
18:33:47.906 Cisco Messaging Interface|<--CMISsapiClient::OnLampOff()
```

Known Issues

- FXS ports on the 6624 blade and AT-x gateway do not go on-hook in a timely manner.

The default behavior of these devices is to go on-hook when they receive reorder tone. However, many voice mail systems do not disconnect on reorder, but rather dialtone. You can modify the Call Restart Timer from its default value of 5000 to **1234** in order to change the behavior of the gateway.

Timers	
Call Restart Timer(1000-5000ms)*	<input type="text" value="1234"/>

- **Error messages about the Cisco Messaging Interface in event viewer.**

```
Error: CMyProcessConfigList::GetSafeIntegerParam() catch  
handler Parameter MessageDeskNumber not found in database, using  
default value of 1.
```

In early releases of Cisco CallManager, the Message Desk parameter was hard coded to 1. In Cisco CallManager 3.0(7), support for configurable Message Desk numbers was introduced. If you do not define a Message Desk Number, Cisco CallManager uses Message Desk 1.

- **Cisco Messaging Interface does not stay running.**

With the release of Cisco CallManager 3.0(7), the VoiceMailDn field is required.

- **No MWI with Octel 250/350 when Octel uses multiple SMDI links.**

When Octel voice mail systems use multiple SMDI links, it is necessary to specify over which link the MWI command needs to be sent. In the mailbox profile, you can edit the **Int. Link Number** field to correspond to the SMDI link that connects the Octel to the Cisco CallManager.

Cisco DPA 7630/7610

The Cisco DPA 76xx Voice Mail Gateway is a VoIP gateway that enables legacy voice mail equipment to connect to a Cisco IP Telephony solution network. The Cisco DPA 7610 allows Octel voice mail systems that currently use Digital Station Emulation to connect to a Nortel Meridian 1 PBX, to instead connect to a Cisco IP CallManager system. The Cisco DPA 7630 allows Octel voice mail systems that currently use Digital Station Emulation to connect to an Avaya Definity PBX, to instead connect to a Cisco IP CallManager system without any changes to the voice mail system.

The Cisco CallManager views the Cisco DPA 76xx as a collection of 30 VIP phones and the PBX's Avaya G3 views the DPA as a collection of 7504D digital phones. The Nortel Meridian M1 views the DPA as a collection of 2616 phone sets.

Troubleshoot Calls to Voice Mail through a Cisco DPA 7630/7610

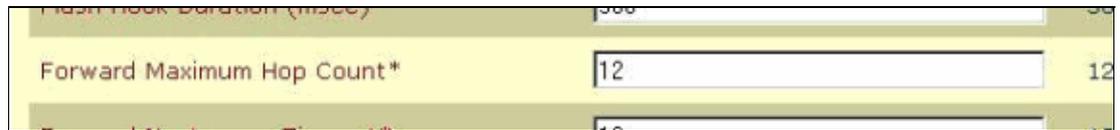
The Cisco DPA 76xx gateways have very precise wiring requirements. It is essential to verify cabling when you troubleshoot issues with the Cisco DPA76xx.

- You can find the Cisco DPA 7630 requirements in Connecting to the Octel and Definity Systems on a Cisco DPA 7630.
- You can find the the Cisco DPA 7610 requirements in Connecting to the Octel and Meridian 1 Systems on a Cisco DPA 7610.

Once you are certain that the Layer 1 issues are resolved, you can investigate these suggestions.

- **Symptom:** Reorder tone is played out to the caller when a call is placed to voice mail.
 - ◆ **Suggestion:** If the DPA is configured beyond twelve ports, the thirteenth caller can receive reorder if the first twelve ports are in use. Cisco CallManager has a service parameter titled

ForwardMaximumHopCount. By default, this field is set to a value of 12.



A screenshot of a configuration interface showing a field labeled "Forward Maximum Hop Count*" with the value "12" entered. The field is highlighted in yellow.

Set this value to the number of ports configured on the DPA.

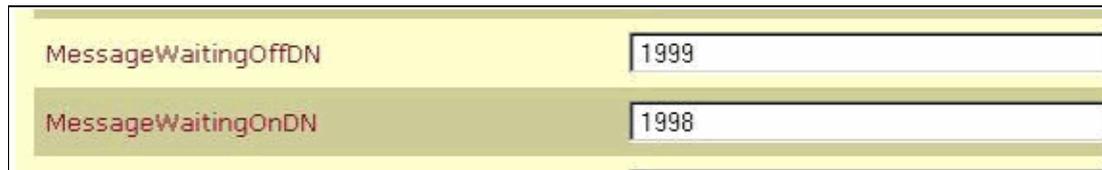
- **Symptom:** Callers get the main greeting (open trees) rather than the greeting for the subscriber's mailbox.

- ◆ **Suggestion:** The DPA relies upon the value entered into the Cisco CallManager 'Pilot' directory number field for deployments of more than one DPA. The CallManager 'Pilot' directory number should be the first port of the first DPA. While the first DPA knows the directory numbers (DNs) associated with its own ports, subsequent DPAs need this information to ensure correct integration.

Note: The most common misconfiguration is when a customer configures a CTI route point or a hunt group that forwards to the DPA. This is not necessary, as you can simply assign the appropriate DN to the first DPA port.

Troubleshoot MWI with a Cisco DPA 7630/7610

In order for message waiting notification to work correctly, Cisco CallManager and the DPA must be configured to use the same DNs for MWI.



A screenshot of a configuration interface showing two fields: "MessageWaitingOffDN" with the value "1999" and "MessageWaitingOnDN" with the value "1998". Both fields are highlighted in yellow.

The values that are entered under **Service > Service parameters > Cisco CallManager**, for MessageWaitingOn DN and MessageWaitingOffDN must match the values entered in the DPA under **Configure > CallManager**.

```
CallManager MWI ON directory number (1998)
CallManager MWI OFF directory number (1999)
```

- **Symptom:** MWIs do not turn on or off.
 - ◆ **Suggestion:** Ensure that the MWI on/off DNs are not part of a more generic pattern. For example, check that there is not a route pattern of *lxxx* that points to a gateway.
- **Symptom:** MWIs do not turn off on the Octel 250/350 platform.
 - ◆ **Suggestion:** For MWIs to function properly, the Message Waiting Timeout feature in menu 6.2, In-Band Integration of the Octel system must be set to the **1 – Timeout = Positive Acknowledgment** setting. The **2 – Timeout = Negative Acknowledgment** setting directs the Octel system to interpret the lack of a confirmation tone as an error indication. The DPA 7630/7610 requires that the **1 – Timeout = Positive Acknowledgment** option be used.
- **Symptom:** With the Cisco DPA 7630, MWIs do not turn on or off.
 - ◆ **Suggestion:** The Avaya G3 PBX makes use of feature codes to set message waiting on the PBX. The Octel voice mail system consequently uses them also. These are traditionally *4

and #4 for Leave Word Calling Activate and Cancel, respectively. On the Octel 250/350 platform, in menu 6.2, In-Band Integration, there are two parameters: Dialing Sequence to Activate MWI and Dialing Sequence to Deactivate MWI. These values should be the same as those configured in the DPA under **Configure > Octel/Definity integration**.

```
| Lucent MWI ON pre-extension dial string (*4)
| Lucent MWI OFF pre-extension dial string (#4)
```

For one reason or another, many Octel systems have the Dialing Sequence to Activate MWI and Dialing Sequence to Deactivate MWI configured as *4PN and #4PN, rather than simply *4N and #4N. The P represents a pause, usually of 500 ms, and this delay can cause issues with MWI before DPA code 1.2(1).

Related Information

- [Cisco DPA 7630 and 7610 v.1.2](#)
- [Voice Technology Support](#)
- [Voice and IP Communications Product Support](#)
- [Troubleshooting Cisco IP Telephony](#) 
- [Technical Support & Documentation – Cisco Systems](#)

[Contacts & Feedback](#) | [Help](#) | [Site Map](#)

© 2014 – 2015 Cisco Systems, Inc. All rights reserved. [Terms & Conditions](#) | [Privacy Statement](#) | [Cookie Policy](#) | [Trademarks of Cisco Systems, Inc.](#)

Updated: Feb 03, 2006

Document ID: 18779
