

了解UCCX中的JTAPI架构和呼叫流程

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[JTAPI概述](#)

[JTAPI和UCCX](#)

[JTAPI架构](#)

[JTAPI观测器模型](#)

[JTAPI提供程序模型](#)

[JTAPI用户](#)

[P1和P2手柄](#)

[呼叫流](#)

[故障排除](#)

[启用和收集JTAPI日志](#)

[启用JTAPI调试](#)

[收集JTAPI调试](#)

[从RTMI](#)

[从CLI](#)

[JTAPI日志位置](#)

[参考链接](#)

简介

本文档介绍JTAPI的基本功能、其架构以及与UCCX相关的呼叫流程。

先决条件

要求

思科建议了解以下工具和功能：

- JTAPI - Java电话API
- API - 应用编程接口
- UCCX - 统一联系中心快捷版
- CUCM -

思科统一通信管理器

- CTI - 计算机电话集成

建议掌握下列主题的相关知识：

- Cisco Unified Communications Manager (CUCM)配置
- 统一联系中心快捷版(UCCX)配置

使用的组件

本文档中的信息基于以下软件版本：

- UCCX版本12.5.1.11002-481
- CUCM 版本 12.5.1.11900-146

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

本文档介绍JTAPI架构、呼叫流程、启用调试和收集日志。

JTAPI概述

Cisco Unified JTAPI用作由Sun Microsystems开发的编程接口标准，用于基于Java的计算机电话应用。Cisco JTAPI通过其他Cisco扩展实施Sun JTAPI 1.2规范。您需要使用Cisco JTAPI开发符合以下条件的应用：

- 控制和观察Cisco Unified Communications Manager电话。
- 使用计算机电话集成(CTI)端口和路由点（虚拟设备）路由呼叫。

JTAPI和UCCX

联系中心使用Cisco Unified JTAPI监控设备状态，并发出路由指令，以在适当的时间将呼叫发送到适当的位置。此外，JTAPI还用于在检索呼叫统计数据进行分析时开始和停止记录指令，并用于将屏幕弹出呼叫引入CRM应用、自动脚本编写和远程呼叫控制。

JTAPI架构

在企业环境中使用的Cisco Unified JTAPI结合用户可用性、位置和首选项，为基于在线状态的路由提供定制的独特环境。

以下是JTAPI的应用：

- JTAPI可以监控两个或多个电话和线路组合或收到通知。
- 联系中心应用使用JTAPI的完整呼叫模式。
- JTAPI提供以下功能：
 - 呼叫控制
 - 媒体控制
 - 媒体协商

JTAPI观测器模型

下图说明了JTAPI运行的Provider-Observer模型。

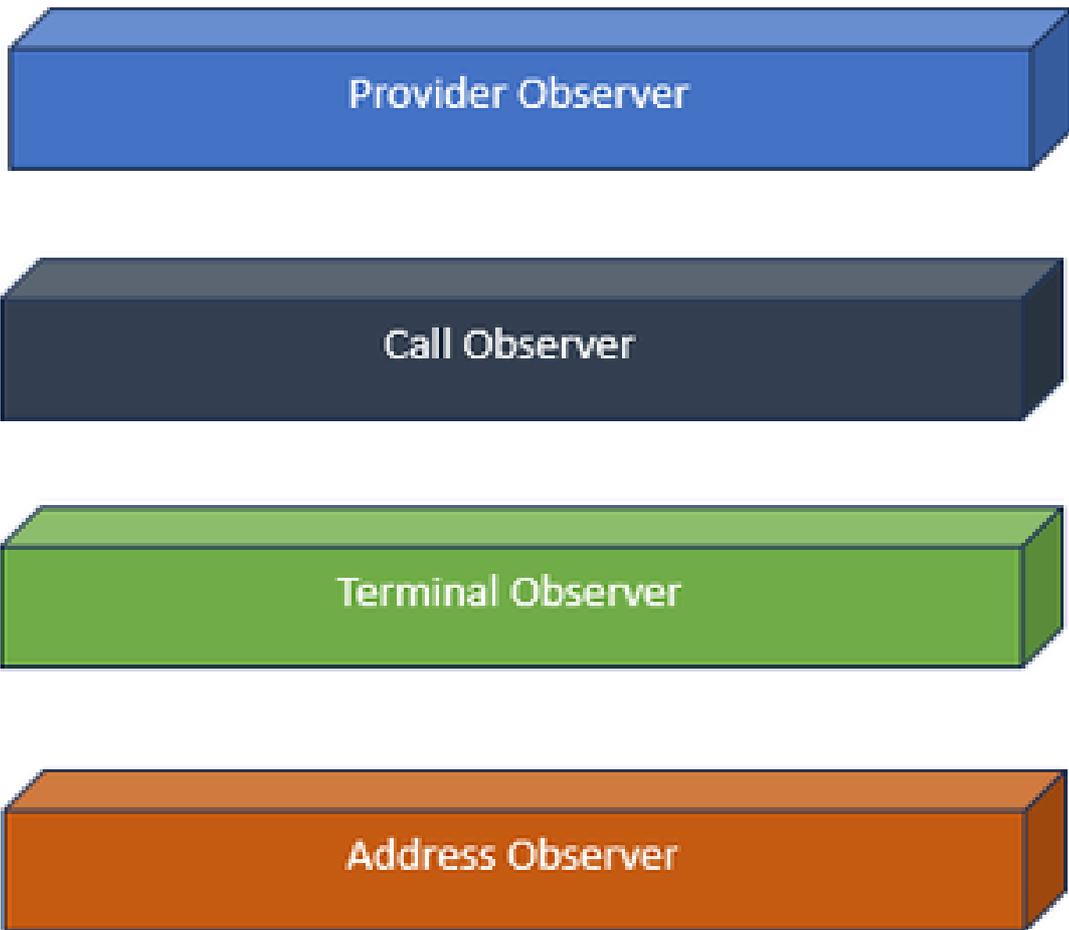
Provider

Call

Terminal

Address

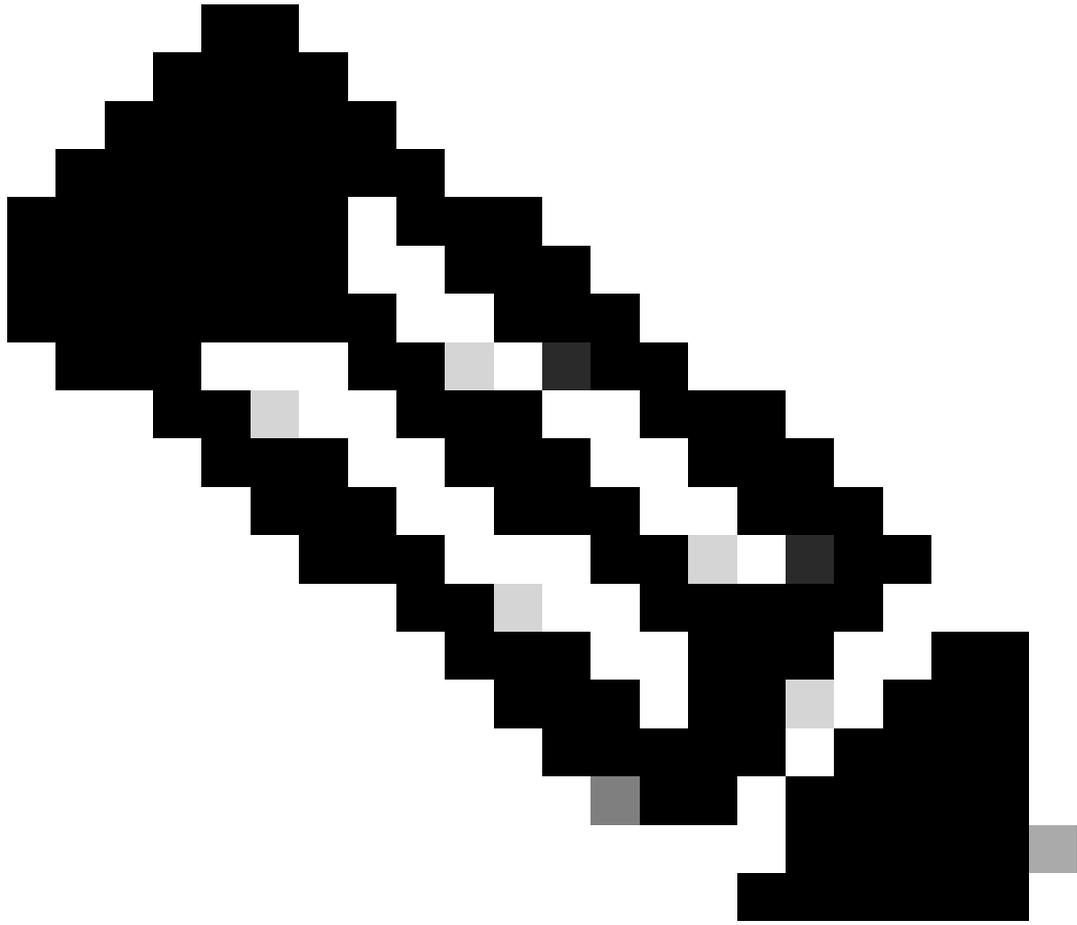
观察员



观察器接口

JTAPI是基于观察者的思想。观察者指的是观察事件的人的观点。您可以在系统中将多个观察者置于不同的对象上。JTAPI使用观察者来了解对象的状态变化。

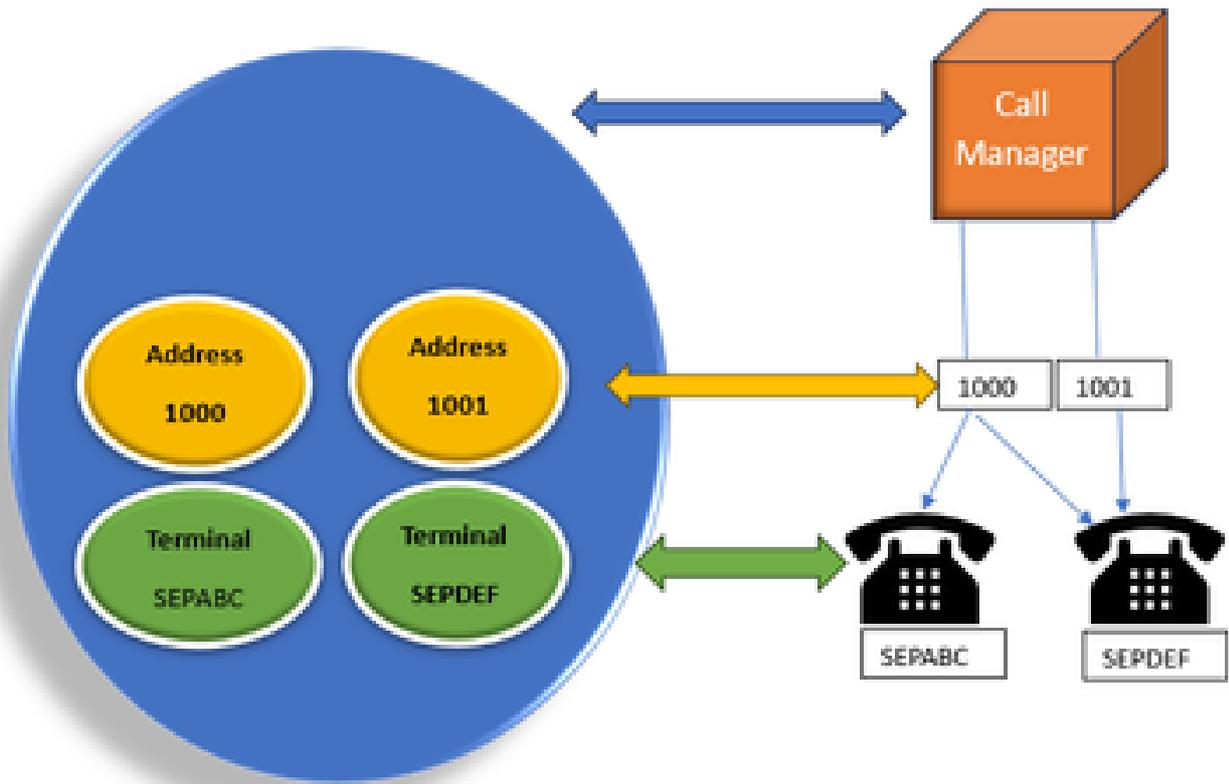
例如，您可以将观察者置于线路、电话、终端、地址等上，并了解其状态变化。



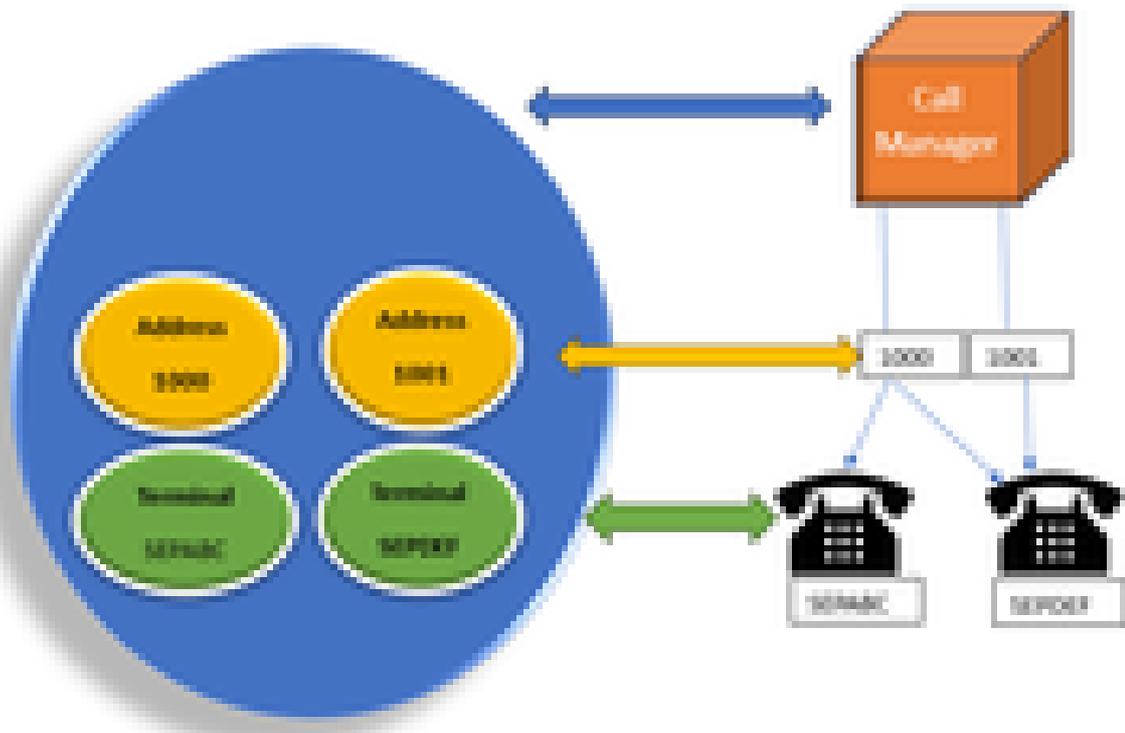
注意：如果没有将观察者置于对象上，则无法收到有关对其采取的操作的通知。

JTAPI提供程序模型

下图说明了JTAPI运行的提供商模型：



JTAPI提供程序模型



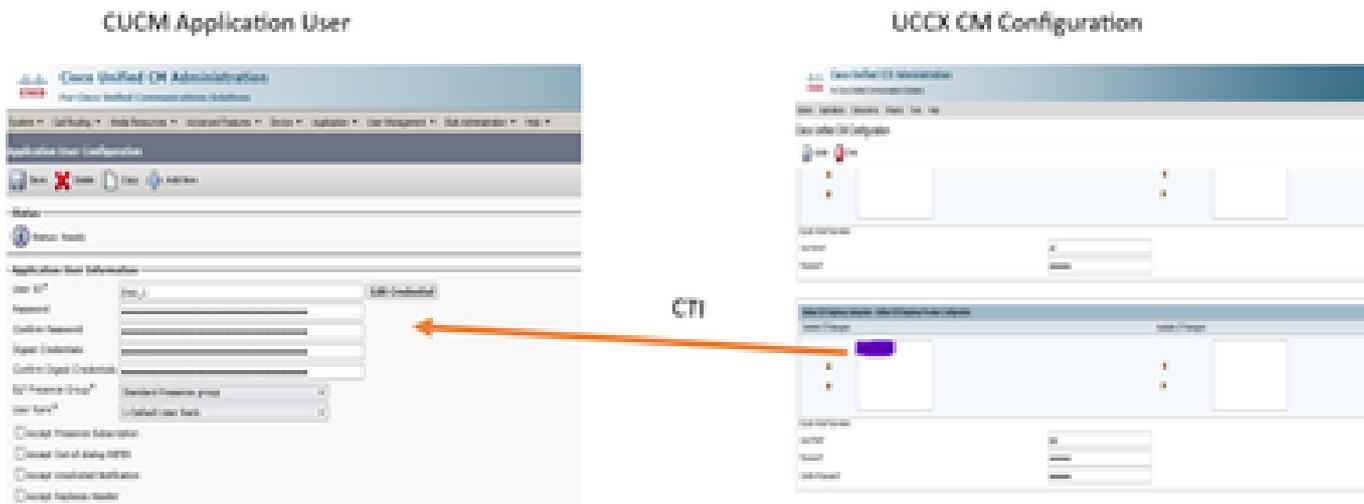
提供程序模型

JTAPI提供程序是从应用程序到Call Manager或CTI Manager的连接。提供程序用于将观察者附加到对象。还可以将观察者附加到提供

程序。提供程序用于获取有关对象所执行操作的通知。(您还可以控制连接了观察器的设备 (来自连接该观察器的提供商)。

JTAPI用户

接下来的屏幕截图取自CUCM (左) 和UCCX (右), 用于说明JTAPI应用用户。



- 当您启动应用程序并要与CTI管理器建立连接时, 您会打开一个提供程序。
- 打开提供商的原因是为了您可以监控在设备、终端等上执行的操作。
- 在CUCM中实施的方式是通过应用用户。
- 您可以创建此用户并提供凭证, 使其可以首先通过CTI向CUCM进行身份验证。
- 因此, 在CUCM中创建的JTAPI应用用户是提供商。
- 除了简单监控, 还需要确保这些设备在关联设备下, 以确保可以控制设备。

注意：您不会在cucm上创建JTAPI用户，这是由应用(UCCX)通过CUCM上的AXL创建的。

P1和P2手柄

P1和P2是提供商句柄。当您要从同一应用打开多个提供商时，这些变得很重要。从UCCX中，您可以创建2个提供商，其中一个提供商控制CTI端口和路由点，另一个提供商控制称为RM-JTAPI提供商的座席电话。您在UCCX创建首先控制CTI端口和路由点的提供商时，会首先创建JTAPI P1提供商。在P1之后创建的另一个提供程序是处理代理设备的P2提供程序或RmCm提供程序。

如果看到P1新呼叫事件，则知道这是来自CTI端口或CTI路由点的呼叫事件。如果看到P2新呼叫事件，则表明这是来自实际电话的呼叫事件。您在CCX端创建了一个RM-JTAPI用户和一个JTAPI用户，但在CUCM端，您会看到创建了2个JTAPI用户。这是因为每个CCX节点都有自己的JTAPI用户，但它们共享一个RM-JTAPI用户。在UCCX上创建触发器时，触发器将添加到两个JTAPI用户。两个

节点分别打开一个提供程序。因此，在路由点上执行的任何操作都会在两个CCX节点上收到通知。

除此之外，辅助节点只是停留在原处，并不断通知它仍然是辅助节点。每个节点都有一组单独的CTI端口。节点1的CTI端口由jtapi_1控制。节点2的CTI端口由jtapi_2控制。

当呼叫到达CTI路由点时，两个CCX节点都会收到有关新呼叫事件的通知，但主用CCX节点会向呼叫管理器回复其控制的一个CTI端口的重定向请求。

如果您在CUCM中看到针对CTI路由点的IP，它是CCX的一个IP，但这并不意味着特定节点正在路由呼叫。

您经常会做的一件事是，我们将设备与JTAPI用户取消关联，然后重新添加。取消关联后，提供程序会收到有关它的通知并删除与它的所有连接；重新添加后，观察程序会再次添加，而提供程序会开始使用打开的提供程序请求再次监视它。您可以看到，除了添加到受控设备列表中的设备外，还有另一个称为“允许通过CTI控制设备”复选框的设备。这是为了带来更高级别的粒度。因此，如果您已在受控列表中添加路由点，但未选中CTI复选框，您仍然可以获取有关该路由点上的事件的通知，但无法对呼叫控制执行任何操作。

呼叫流

以下是UCCX呼叫流程中涉及的事件的顺序：

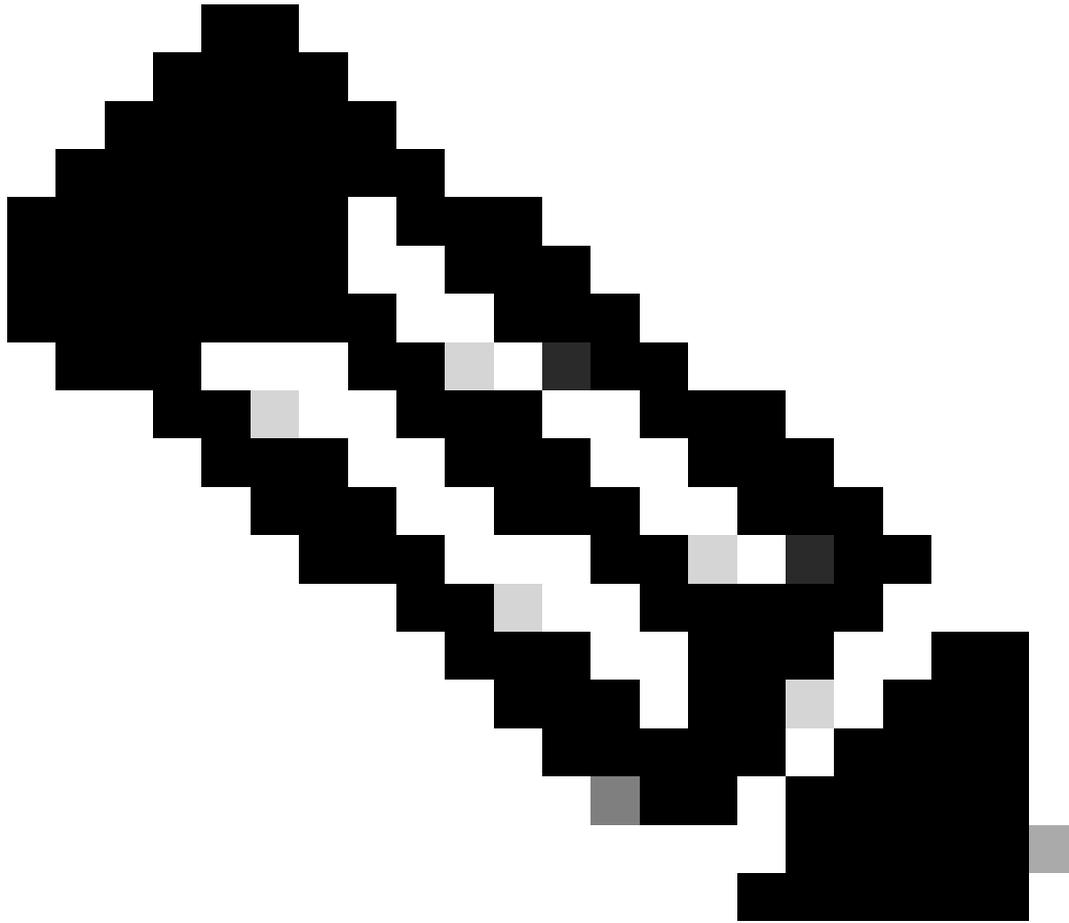
- 当呼叫到达CTI路由点时，它被重定向到CTI端口。
- CTI端口打开uccx盒上带有ipvms驱动程序的媒体信道。
- 一旦您确定座席需要接听呼叫，则从端口向座席进行咨询转接。
- 新的呼叫事件被发送到CTI路由点。
- 重定向请求转到CTI端口。
- 呼叫ID的状态变为空闲。
- 然后，另一个新的呼叫事件将发送到CTI端口。
- 如果重定向响应为0，则表明响应正常；如果响应非零，则表明响应失败。
- 然后，您发送呼叫接受请求（此请求未被应答，仅在端口上被接受）。
- 然后接受脚本中的步骤命中，即呼叫应答请求进入Jtapi时。



注意：这种情况发生得太快，而且有时会同时发生多个事件，例如来自Cisco Unity Connection的呼叫或来自CUCM的转移占用时间，这可能在接受步骤中导致RACE情况，这也是您要降低速度的原因。您可以通过添加延迟步骤before accept步骤来执行此操作。

11. 然后从端口获得应答响应。

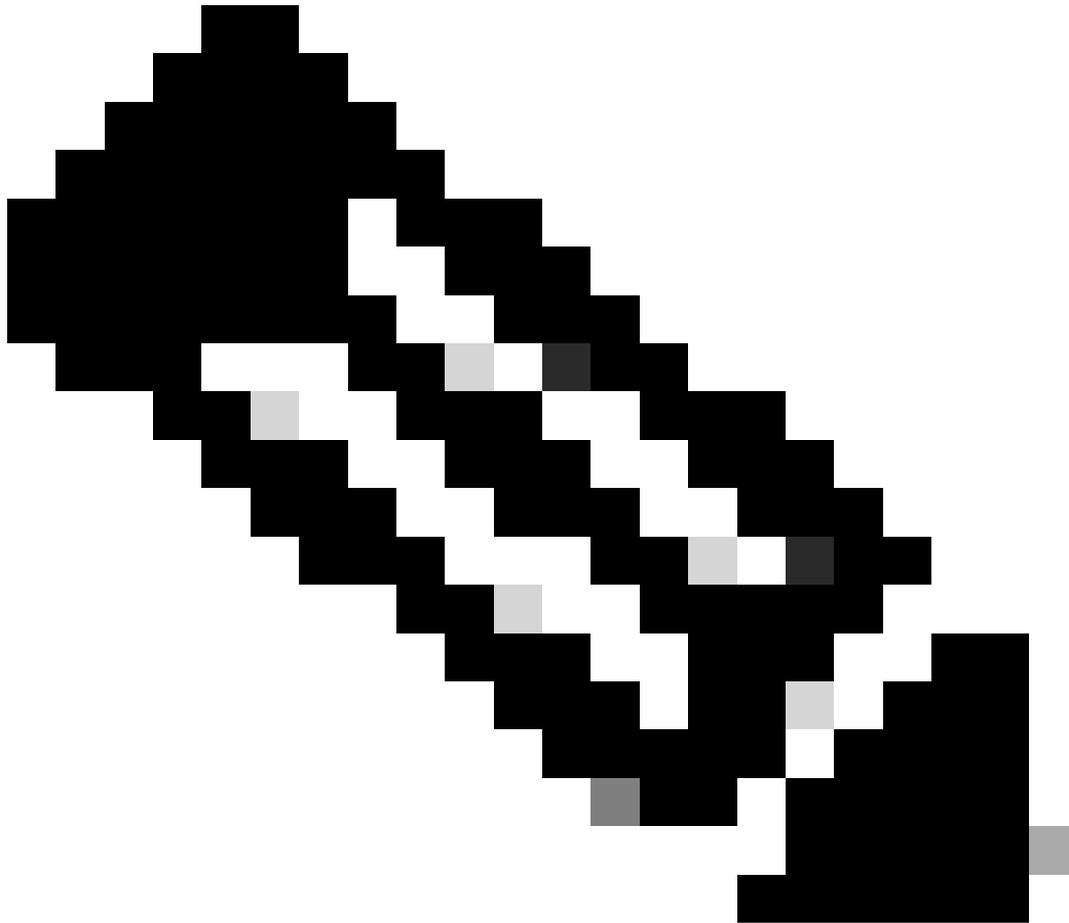
12. 呼叫状态更改为“已连接”。



注意：CTI不同步，而sip或skinny电话在发送新请求之前等待响应，这就是为什么JTAPI CTI消息中的消息顺序可以乱码乱码的原因。

-
13. 从端口获得应答响应后，将进行媒体协商。
 14. 端口发送open_logical_channel请求，在该请求中它共享其端口以及它希望远程方向其发送RTP的ip。
 15. 在打开逻辑通道之前，它会与UCCX盒上的IPVMS驱动程序建立连接并打开RTP流。
 16. 下一个事件是start_reception事件，它告诉我们远端信息（即呼叫设备的ip和端口）。
 17. 然后您会像收到任何其他媒体信号一样收到start_transmission事件。

18. 现在您正在收听IVR和提示。



注意：这是呼叫设置完成的位置。

19. 现在，当呼叫到达脚本中使呼叫转接至座席的一个步骤时，您会看到CallSetupTransferRequest。

20. 与第一条消息不同的是，这是咨询转接，而不是重新定向。

21. 将呼叫转为咨询转接的原因在于，如果座席已就绪，但不在座席处，并且我们重定向呼叫，则呼叫仍未应答，但如果进行咨询转接，则如果座席不在场，则呼叫将再次置于队列中。

22. 像任何其他咨询转接一样，这是CTI端口第一次按电话上的转接按钮。

注意： `ConsultWithoutMedia` 是用于咨询转接的API。

23. 在定期咨询转接中，A和C之间有一个媒体路径，但在本例中，您指示CUCM不要这样做，因为在UCCX和代理之间没有建立媒体的感觉。

24. 因此，您需要在代理和CTI端口之间不进行媒体切换的情况下执行咨询转接。

25. 此时，CTI端口将呼叫方置于保持状态，我们看到呼叫状态已更改，事件为HOLD。

26. 现在，您会看到从CTI端口到座席设备的新呼叫事件。（使用原始呼叫ID，但使用其中的一个子集和P2事件。）

27. 如果您搜索P2事件呼叫ID，则会看到下一条消息为CallAnswer请求，表示座席已接听呼叫。

28. 一旦您知道坐席已接听呼叫（使用P2），并且CTI端口侧呼叫也处于连接状态（使用P1），则路由点将看到 CallDirectTransferRequest(这意味着已第二次按转接按钮)。

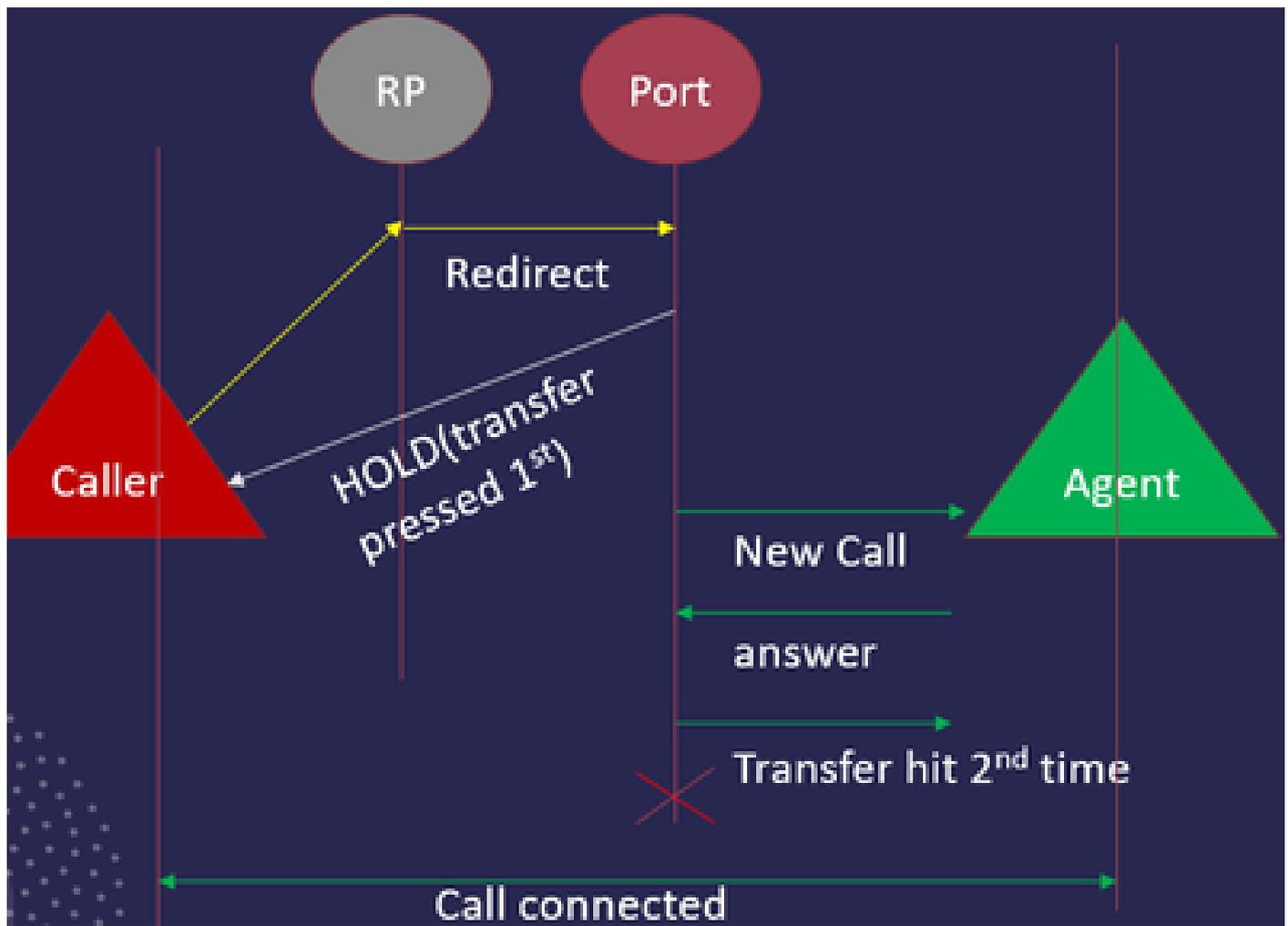
29. 现在，由于CTI端口知道座席已接听呼叫，因此无需等待，它会立即发送CallDirectTransferRequest 消息，即CTI端口第二次按下转接按钮，如上所述。

30. 现在，原来的呼叫段（处于保持状态的呼叫段）是幸存的呼叫段。

31. 另一个呼叫段（端口和座席之间）被破坏。

32. 此时，CUCM和UCCX在图外，RTP通过网关在呼叫方和座席之间建立。

下图以总结的方式说明了前面提到的呼叫流程。



JTAPI呼叫流摘要

故障排除

启用和收集JTAPI日志

启用JTAPI调试

请检查以下步骤以启用JTAPI调试级别。

- 登录UCCX。
- 转到CCX Serviceability。
- 转至跟踪。
- 转到配置。
- 从Select Service下拉列表中选择Cisco Unified CM Telephony Client。
- 选中Debugging复选框。
- 选中除MISC_DEBUGGING之外的所有复选框。

收集JTAPI调试

请检查以下步骤，以从RTMT或CLI启用JTAPI调试级别。

从RTMT

- 登录CCX实时监控工具。
- 单击Trace & Log Central。
- 单击Collect Files。
- 为所有服务器选择JTAPI Client。
- 单击 Next。
- 选择要保存日志文件的时间戳和位置。

从CLI

JTAPI日志位置

```
activelog /uccx/log/JTAPI
```

用于收集JTAPI日志的命令：

```
file get activelog /uccx/log/JTAPI/* recurs compress
```

语法:

```
file get { activelog|inactivelog|install } file-spec [options]
```

要传输的file-spec必需文件

选项可选剩余时间月份 | 周 | 日 | 小时 | 分钟时间值

`abstime hh : mm : MM/DD/YY hh : mm : MM/DD/YY`

`match regex`

循环

压缩

根据时间戳下载日志的5种方法

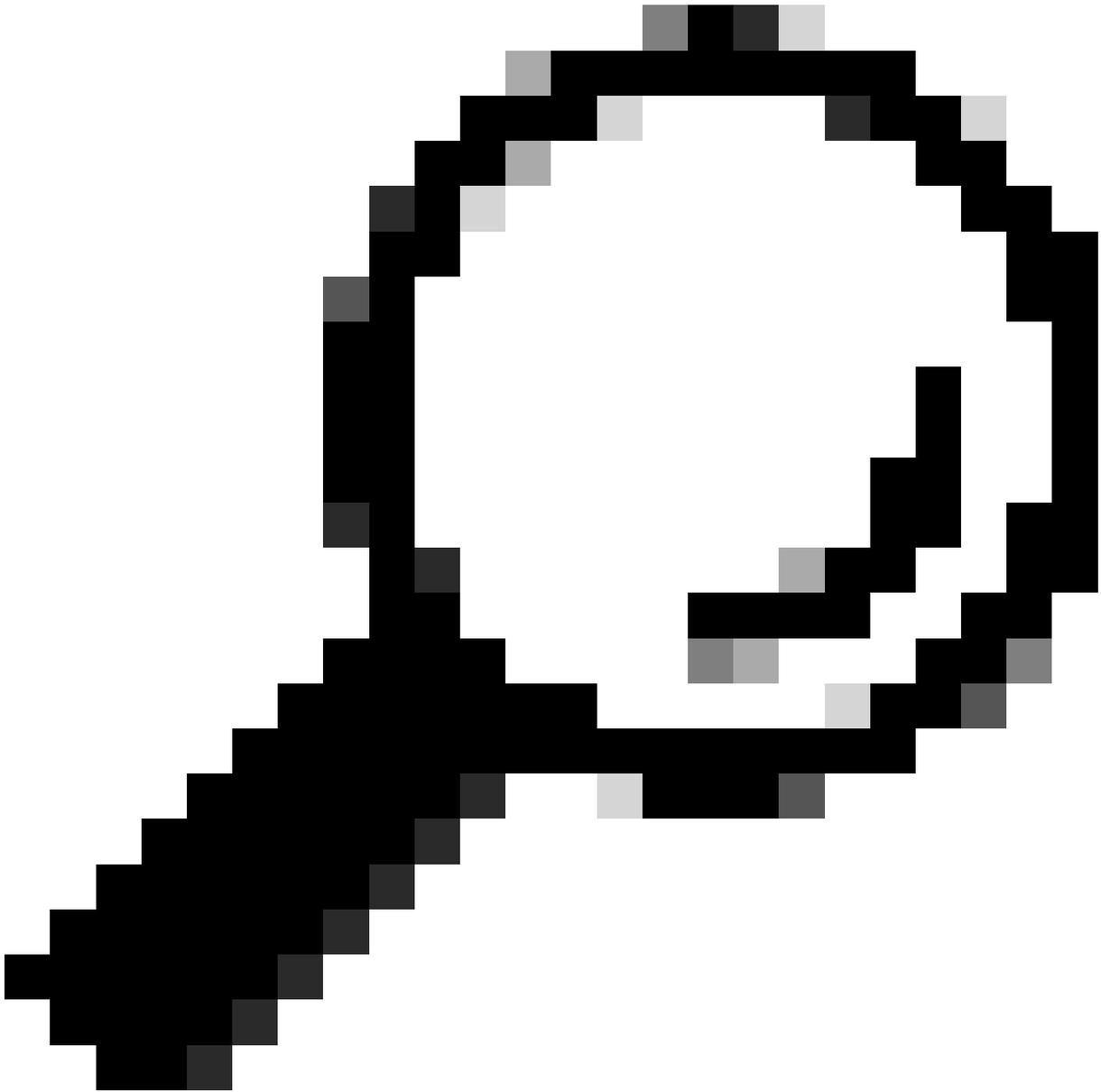
reltime -相对时间段，指定为分钟 | 小时 | 天 | 周 | 月数值

abstime -绝对时间段，指定为hh : mm : MM/DD/YY hh : mm : MM/DD/YY

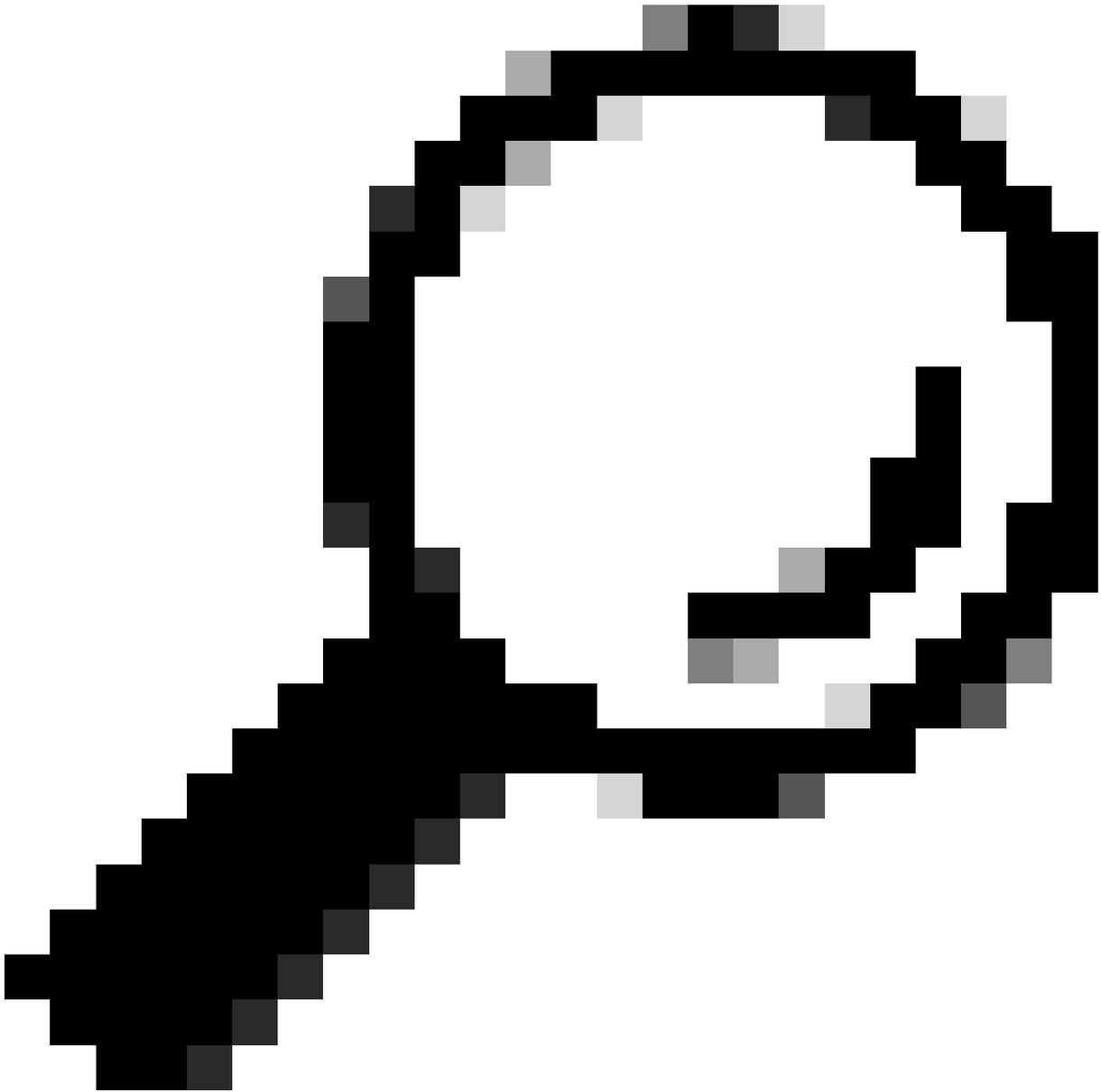
match— 匹配文件名中指定为字符串值的特定字符串

recurs -获取所有文件，包括子目录

compress选项允许您以压缩格式下载文件。



提示：要下载文件，请确保已配置外部安全文件传输协议(SFTP)服务器并且可访问。



提示： Recurs选项允许您遍历所有子目录和文件的目录。如果要从目录提取所有日志，则使用此命令。

参考链接

- [《JTAPI开发人员指南》](#)
- [UCCX跟踪级别](#)

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。