

排除IX5000 Touch 10故障

目录

[简介](#)

[常见问题](#)

[即使整个系统成功启动，触摸面板也无法打开/启动电源](#)

[全新接触无法升级 — 停滞于\[维护模式\]](#)

[连接丢失 — \[连接丢失\]显示在用户界面中](#)

[Android崩溃 — 例如\[Phone App Has Stopped\]](#)

[UI状态不一致](#)

[通过串行连接从Touch10发送日志](#)

[恢复程序](#)

[出厂重置运行IX代码的Touch 10](#)

[运行TC/CE代码的出厂重置Touch 10](#)

简介

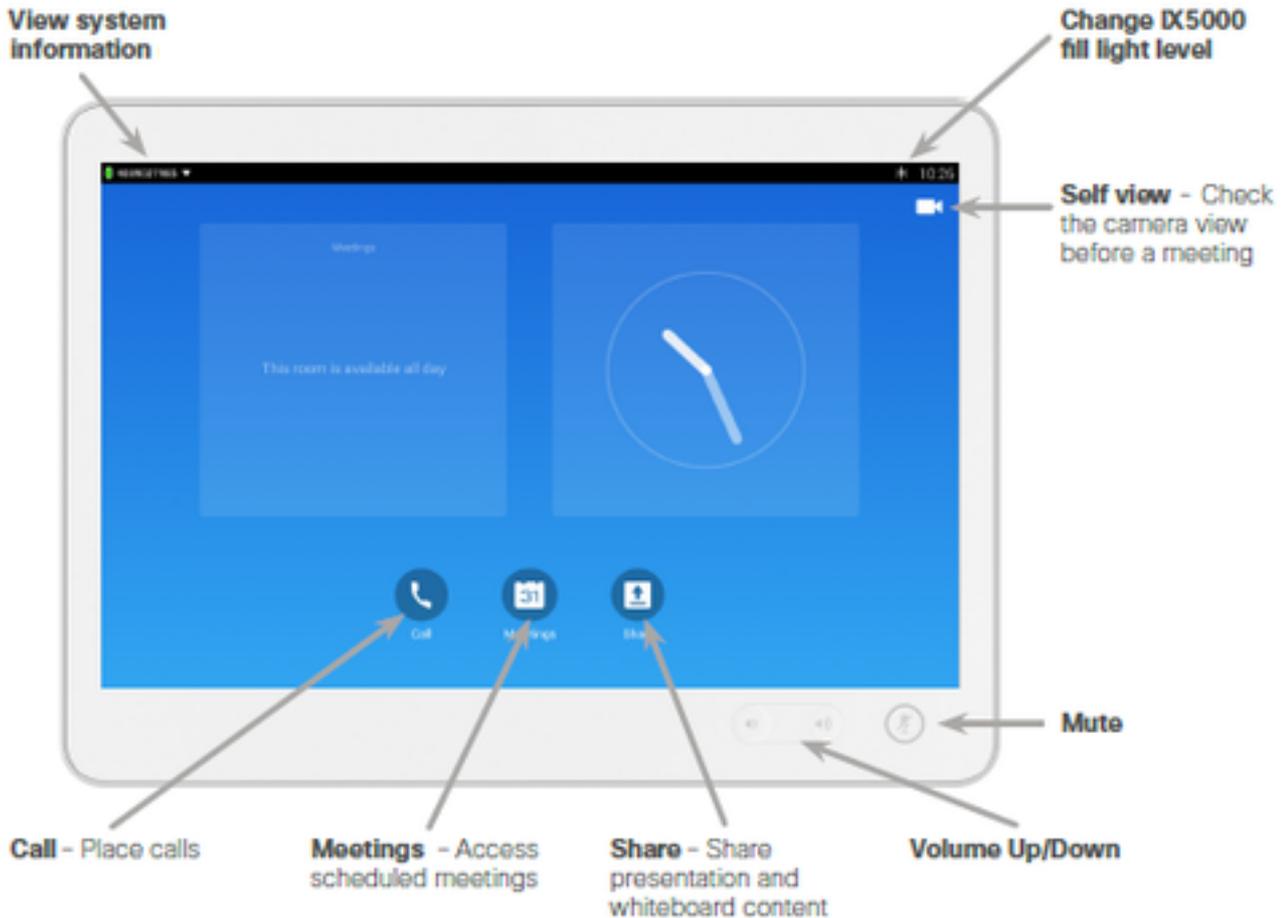
本文档介绍现场观察到的不同场景以及故障排除、隔离和收集信息以解决问题的步骤。

IX5000是新一代思科网真沉浸式终端，使用Touch 10而非CTS和TX沉浸式系统使用的Touch 12。它使用与TC终端不同的用户界面(UI)软件；但是，它使用相同的Android基础。

常见问题

/

Touch 10 Home screen overview



1. (PoE)Touch 10

2.

3. Touch 10Touch 10PoE \11 \ 注意：**Mode**按钮位于交换机的正面和下面，与电源插头位于同一侧。交换机LED在三秒后开始闪烁，七秒后停止闪烁。然后，交换机重新启动并丢失其配置。要重新启动编解码器，请登录IX5000管理图形用户界面(GUI)并单击**重新启动/重置**。GUI的默认IP地址为169.254.1.1，默认用户名和密码为admin/cisco。如果已配置编解码器，则IP地址不同。交换机重新配置，Touch 10设备成功初始化。

一 0

TouchTCIXIXUI“maintenance mode.....”IXIX“”“//” TCIX8.1.1

另一种故障排除方法是，如果您拥有基于MX/SX的终端，且该终端基于SW代码TC7.1或更高版本，则Touch可以重新连接以获取TC软件，然后可以重新连接到IX。由于这有助于Touch恢复默认软件，因此，当您将IX连接到IX时，IX的软件可以再次被覆盖。

一 0

与IX5000的连接丢失通常在用户界面中显示为连接丢失。如果心跳丢失到IX5000,IX软件会显示此消息。心跳是每15秒发送到IX的命令/响应。如果两个心跳丢失(30秒后无响应),触摸单元将不再通过UI运行,因为命令不会通过IX,状态更新也不会从IX中检索,因此,消息将显示。此外,在启动Touch后,如果无法建立到IX的连接/配对,则显示消息。Touch将持续尝试建立与IX的连接,以达到正常的可操作状态。重新建立连接后,消息将消失。

出现“Lost of Connection (连接丢失)”消息时,触摸设备很少出现问题,必须执行IX和表开关的故障排除才能解决问题。

如果房间内的所有触摸设备突然收到此消息,但IX似乎运行良好,则表交换机出现问题的可能性很大。提取日志捆绑包时,位于/nv/log/touch/文件夹的日志。

日志中的心跳示例

```
LOG_NOTICE(169.254.1.102):06-08 12:16:28.683 WARN
com.cisco.telepresence.system.SystemServiceSocketThread #codecxcommandHeartBeat
ID:"88:43:E1:C6:54:51""30" | resultId="18093" LOG_NOTICE(169.254.1.101):06-08 12:16:34.785
WARN com.cisco.telepresence.system.SystemServiceSocketThread #codecxcommandHeartBeat
ID:"88:43:E1:C6:52:8E""30" | resultId="18476" LOG_NOTICE(169.254.1.102):06-08 12:16:43.718
WARN com.cisco.telepresence.system.SystemServiceSocketThread #codecxcommandHeartBeat
ID:"88:43:E1:C6:54:51""30" |ID="18094"
```

Android — [Phone App Has Stopped]

每当进程/应用中引发未处理的Java异常时,通常都会通过标准消息{the_app has stopped}(带有确认按钮)看到此异常。这不一定会妨碍系统的正常使用,并且可能不会产生任何后果。但是,它们不能发生。

为了调试发生此类崩溃的原因,只要在重现崩溃后尽快检索日志,logcat就会输出回溯。可能会引发许多不同类型的异常,因此最好在日志中搜索FATAL或Exception。请记住,IX上的每个文件都需要经过调查才能找到崩溃。当解压日志包时,这些文件夹位于/nv/log/touch/文件夹中。

以下是控制面板应用崩溃的示例,其中显示UI中的“控制面板已停止”消息:

```
2015-07-08 02:21:32.467 - FATAL EXCEPTION: main 2015-07-08 02:21:32.467 - Process: com.cisco.telepresence.dashboard,
PID: 6825 2015-07-08 02:21:32.467 - java.lang.NullPointerException 2015-07-08 02:21:32.467 - at
com.cisco.telepresence.dashboard.adapter.
MediaChannelListAdapter.
MediaChannelListAdapter.configureAdapter(MediaChannelListAdapter.java:76) 2015-07-08 02:21:32.467 - at
com.cisco.telepresence.dashboard.fragment.
MediaChannelListAdapter.onViewCreated(MediaChannelListAdapter.java:30) 2015-07-08 02:21:32.467 - at
android.app.FragmentManagerImpl.moveToState(FragmentManager.java:904) 2015-07-08 02:21:32.467 - at
android.app.FragmentManagerImpl.moveToState(FragmentManager.java:1062) 2015-07-08 02:21:32.467 - at
android.app.BackStackRecord.run(BackStackRecord.java:684) 2015-07-08 02:21:32.467 - at
android.app.FragmentManagerImpl.execPendingActions(FragmentManager.java:1447) 2015-07-08 02:21:32.467 - at
android.app.Fragment.performStart(Fragment.java:1721) 2015-07-08 02:21:32.467 - at
android.app.FragmentManagerImpl.moveToState(FragmentManager.java:918) 2015-07-08 02:21:32.467 - at
android.app.FragmentManagerImpl.moveToState(FragmentManager.java:1062) 2015-07-08 02:21:32.467 - at
android.app.BackStackRecord.run(BackStackRecord.java:684) 2015-07-08 02:21:32.467 - at
android.app.FragmentManagerImpl.execPendingActions(FragmentManager.java:1447) 2015-07-08 02:21:32.467 - at
android.app.FragmentManagerImpl$1.run(FragmentManager.java:443) 2015-07-08 02:21:32.467 - at
android.os.Handler.handleCallback(Handler.java:733) 2015-07-08 02:21:32.467 - at
android.os.Handler.dispatchMessage(Handler.java:95) 2015-07-08 02:21:32.467 - at android.os.Looper.loop(Looper.java:136)
2015-07-08 02:21:32.467 - at android.app.ActivityThread.main(ActivityThread.java:5076) 2015-07-08 02:21:32.467 - at
java.lang.reflect.Method.invokeNative(Native Method) 2015-07-08 02:21:32.467 - at
```

```
java.lang.reflect.Method.invoke(Method.java:515) 2015-07-08 02:21:32.467 - at
com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:779) 2015-07-08 02:21:32.467 - at
com.android.internal.os.ZygoteInit.main(ZygoteInit.java:595) 2015-07-08 02:21:32.467 - at dalvik.system.NativeStart.main(Native
Method) 2015-07-08 02:21:33.059 - LOG_NOTICE(169.254.1.8) :07-08 12:21:24.907 WARN
```

UI

如果UI未反映系统的正确状态，例如，（返回呼叫）栏在呼叫外部可见，或（返回呼叫）栏在呼叫中不可见，这可能是由来自IX的不完整状态更新引起的。例如，如果呼叫突然中断，IX会中断、介质问题等。

如果状态仍然存在，则重新启动或重置触控器可以解决问题。

Touch10

直接从有问题的Touch本身提取日志非常有益，特别是在Touch出现引导问题/连接丢失/软件升级问题时，因为没有日志可能会被传输到IX。可以从Touch提取日志，将微型USB电缆连接到Touch背面（用于为基于Android的普通手机充电的电缆）和计算机。使用以下设置打开串行终端：

波特率：115200

数据/部件/停止：8n1

提供外壳。在此外壳中输入logcat以输出完整日志。输入bugreport以输出日志和其他硬件/网络信息。必须从终端复制到文件或保存。日志在启动后将无法存在，因此在启动以进行恢复之前捕获它非常重要。

恢复程序

如果认定触摸面板是有问题的单元，请完成退货授权(RMA)。在完成RMA之前，尝试恢复面板。

- 直接从触摸面板收集日志，如串行连接所述。
- 通过重新通电重新启动Touch（重新连接Touch背面的网线）。
- 如本文档所述，执行触摸的出厂重置。
- 如果您有基于MX/SX的系统运行软件版本TC7.1或更高版本，则可以连接Touch以恢复出厂包。这通过重新同步而不是HTTP传输，这可能是使触摸处于可操作状态的最后手段。恢复后，将其连接回IX5000。

IXTouch 10

1. 从Touch10背面拔下电源/网络电缆。
2. 按住“Volume up（音量）”硬按钮并重新插上电源/网线。
3. 等待静音硬按键亮起（红色）—约10秒。
4. 释放Volume up按钮并单击Mute（静音）硬按钮。在步骤4后确认静音按钮上的绿色闪烁时，已成功执行出厂重置。

TC/CETouch 10

1. “”10
2. “Volume down”
3. “”