

瞭解UCCX Finesse架構深入探討

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[50,000英呎檢視](#)

[Finesse Tomcat](#)

[HTTPs](#)

[XMPP](#)

[PUBSUB](#)

[BOSH -使用同步HTTP的雙向流](#)

[CTI](#)

[JTAPI](#)

[30000英呎檢視](#)

[休眠](#)

[AXL](#)

[肥皂](#)

[20000英呎檢視](#)

[APACHE SHINDIG](#)

[WAR檔案](#)

[10000英呎檢視](#)

[AJAX - Finesse之美](#)

[使用AJAX的優點](#)

[使用AJAX](#)

[使用AJAX向伺服器傳送請求](#)

[案頭體系結構](#)

[小工具架構](#)

[參考連結](#)

簡介

本文檔全面介紹了Finesse體系結構，以便在對finesse問題進行故障排除時，使底層流程變得合理。

。

必要條件

需求

思科建議瞭解以下工具和功能：

JTAPI - Java電話API

API -應用程式設計介面

UCCX -統一聯絡中心快捷版

CUCM -思科統一通訊管理器

CTI -電腦電話整合

採用元件

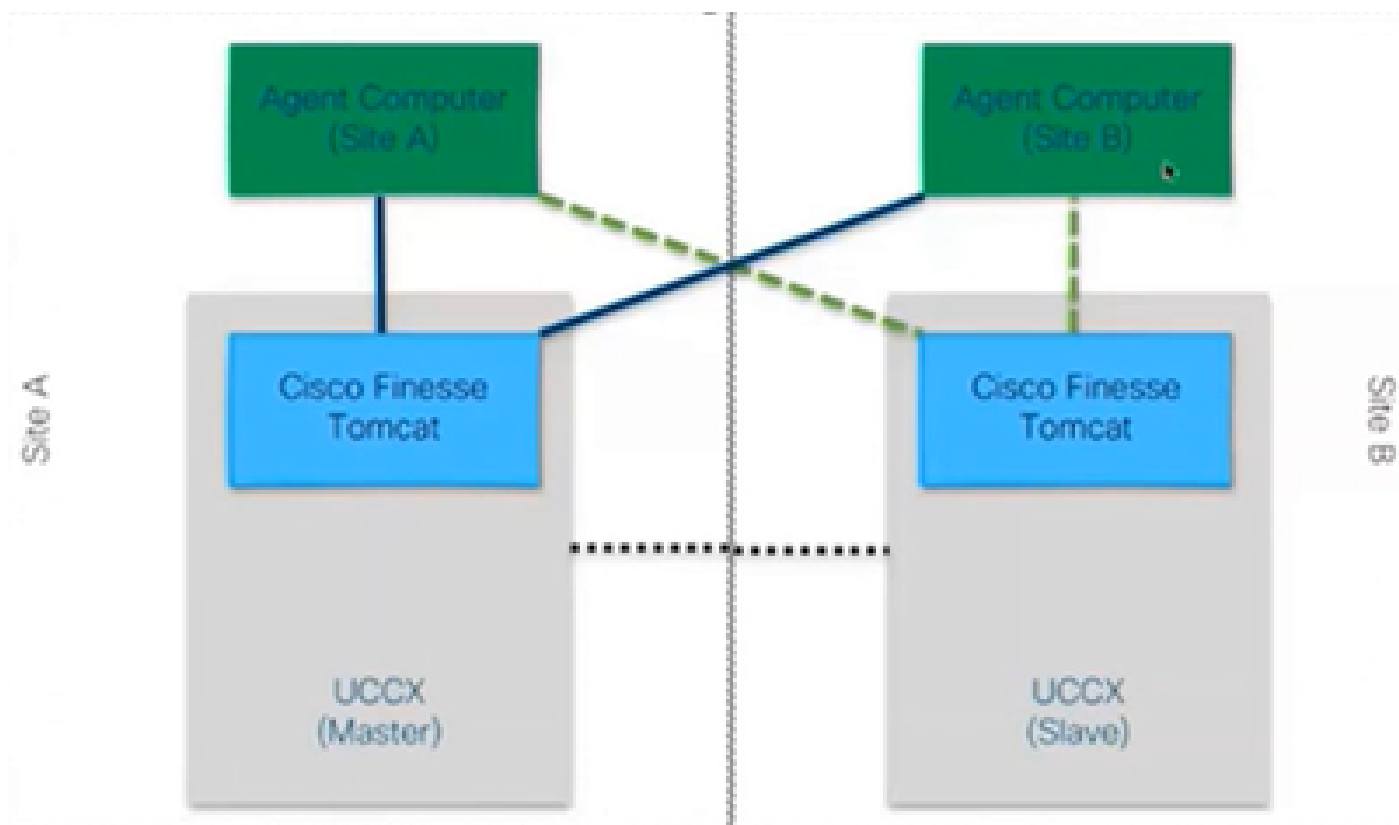
- Cisco Unified Contact Center Express (UCCX)

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

背景資訊

本文檔介紹的Finesse架構從高級概述開始，然後是深入的訊號流，以及示例和圖表。

50,000英呎檢視



50000檢視

Finesse Tomcat

Finesse Tomcat與CUCM中的Cisco Tomcat類似，因為載入網頁的功能相同，但適用於名為Finesse的其他服務。Tomcat是為Finesse而設計的，因為它是一個獨立的Web應用程式。根據11.5之前的版本，您只能登入使用CCX主節點。從11.6開始，只能登入到任何節點（但不推薦），而且只能在故障切換期間登入。因此，如果您考慮WAN集群，其中兩個代理（在站點A和站點B）都連線到主節點上的Finesse，則在任何時候，都存在到另一個節點的非活動連線(從Cisco Finesse到引擎)，這是故障切換所需的CTI openconf請求。

會定期傳送Openconf要求，以檢查節點是作用中還是待命。如果發生故障轉移，通知服務將使用名為systeminfo API的API，告知客戶端此節點已關閉並且需要故障轉移。接著會執行一個檔案，將瀏覽器重新導向到另一個節點，然後傳送openconf rejavascriptsponse。僅當接受證書時，此故障切換才起作用。（建立與伺服器的安全連線）。

提供3種證書，它們是：

- 該節點的通知服務證書。
- 遠端節點的通知服務憑證。
- 遠端節點的Finesse服務憑證。



注意：需要接受遠端節點證書，故障切換才能正常工作。

HTTPs

它是用於傳送超媒體文檔（例如HTML）的應用層協定。它是為Web瀏覽器和Web伺服器之間的通訊而設計的。它是一個無狀態協定，意味著伺服器不會在兩個請求之間保留任何資料。這是客戶端伺服器協定。對於UCCX，Finesse客戶端在代理瀏覽器(PC)上運行。它使用HTTP向伺服器發出請求。以下是一些常見的HTTP方法：

GET -從伺服器獲取資訊。

POST -傳送資訊至伺服器。

PUT -

用於替換伺服器上的任何內容。

刪除-從伺服器移除資訊。

Finesse使用http請求中的systeminfo api請求。例如，如果您想要變更代理程式的狀態，瀏覽器會傳送PUT而非POST，因為如果傳送POST，伺服器就會混淆，因為它有兩個狀態，哪一個要選取？因此，使用PUT將取代當前狀態。

XMPP

可延伸訊息和狀態通訊協定

它是用於即時消息傳送和線上狀態的一組協定。所有XMPP實體都使用其Jabber ID或JID進行標識。此XMPP協定的一個擴展協定（用於小工具）稱為PUBSUB。

PUBSUB

它不是您可以想到的發佈者訂閱者。它仍然發佈和訂閱，但與資料庫無關。XMPP使用一種稱為節點的機制。節點基本上是在監視您關心的實體。任何對您來說很重要且想要監視的專案，您都可以新增節點至其中。然後PUBSUB會在該節點上訂閱更新或通知。您可以針對每種型別的實體（例如代理程式、通話方塊等）擁有節點。如果為代理建立節點，則會訂閱該代理上的節點，然後無論代理執行什麼操作，都會收到有關它的通知。

此規範的目的在於允許XMPP伺服器（通知服務）獲取發佈到XMPP節點的資訊（主題），然後將XMPP事件傳送到訂閱該節點的實體。

如果是Finesse，電腦電話整合(CTI)伺服器會向Finesse Web服務傳送CTI消息，以告知Finesse有關配置更新，例如（但不限於）建立座席或聯絡服務隊列(CSQ)或呼叫的資訊。然後，此資訊會轉換為XMPP消息，Finesse Web服務會將其發佈到Finesse Notification服務。然後，Finesse Notification服務會透過BOSH將XMPP消息傳送到訂閱了某些XMPP節點的代理。

BOSH -使用同步HTTP的雙向流

BOSH是一種長時間執行的HTTP連線，伺服器將請求保持較長時間，直到收到響應，否則將傳送空響應。這適用於XMPP客戶端和XMPP伺服器，但也可以用於非XMPP應用程式。



注意：XMPP是有狀態的，而HTTP是無狀態的（它不儲存有關最後一個請求的資訊）

如果某個Web應用程式需要與XMPP搭配使用，就會產生多個問題。

問題1：瀏覽器本身不支援基於傳輸控制協定(TCP)的XMPP。

解決方案1：Web伺服器 and 瀏覽器透過HyperText Transfer Protocol (HTTP)消息進行通訊，因此Finesse和其他Web應用程式將XMPP消息包裝在HTTP消息中。

問題2：HTTP是無狀態協定。

解決方案2：您可以使用Cookie/張貼資料來進行此作業。

問題3：第三個問題是HTTP的單向行為，這意味著只有客戶端傳送請求，伺服器只能響應。伺服器無法推送資料使得透過HTTP實施XMPP變得不自然。

解決方案3：要解決此問題，您需要在HTTP和XMPP之間建立網橋。

建議的解決方案包括：

- 輪詢（傳統協定）：重複的HTTP請求要求定義新資料：HTTP輪詢
- 長輪詢也稱為BOSH：傳輸協定，它透過高效使用多個同步HTTP請求/響應對，無需使用頻繁輪詢，來模擬兩個實體之間長期、雙向TCP連線的語義。使用BOSH的原因是為了掩蓋伺服器不必在請求發出後立即作出響應這一事實。響應延遲到指定的時間，直到伺服器有客戶端的資料，然後作為響應傳送。客戶端收到響應後，立即發出新請求，以此類推。

Finesse案頭使用者端（Web應用程式）每30秒透過TCP連線埠7443建立一個過時的BOSH連線。30秒後，如果Finesse Notification Service沒有更新，則Notification Service會傳送一個HTTP回覆，其中包含一個200 OK和（幾乎）空的響應正文。例如，如果通知服務收到有關座席或通話（呼叫）事件存在情況的更新，則資料會立即傳送到Finesse Web客戶端。

總結一下：

Finesse Web客戶端透過TCP埠7443設定到Finesse伺服器的過時HTTP連線(http-bind)。這稱為BOSH long調查。Finesse Notification Service是一種線上狀態服務，可發佈有關座席、呼叫等的狀態更新。如果通知服務有更新，它會以狀態更新作為HTTP回應主體中的XMPP訊息回覆http-bind要求。如果在收到http-bind請求後30秒內沒有狀態更新，則通知服務會在沒有任何狀態更新的情況下進行回覆，以允許Finesse Web客戶端傳送另一個http-bind請求。這樣，通知服務就可以知道Finesse Web客戶端仍然能夠連線到通知服務，並且代理未關閉其瀏覽器或讓其電腦進入睡眠狀態，以此類推。

CTI

您可以使用電腦電話整合(CTI)在撥打、接收和管理電話時，利用電腦處理功能。CTI應用程式允許您執行諸如使用呼叫方ID從資料庫檢索使用者資訊之類的任務，或使用互動式語音應答(IVR)系統收集的資訊將來自使用者的呼叫與其資訊一起路由到相應的服務代表。CUCM上的CTI Manager響應來自UCCX的JTAPI請求。CTI伺服器TCP埠為12018。這就是Finesse伺服器和引擎（CTI伺服器）相互通訊的方式。

以下是透過CTI交換的部分資訊：

- 當前系統配置和未來更新。
- 代理及其狀態。
- 呼叫及其狀態。
- 即時座席、呼叫和隊列統計資訊。

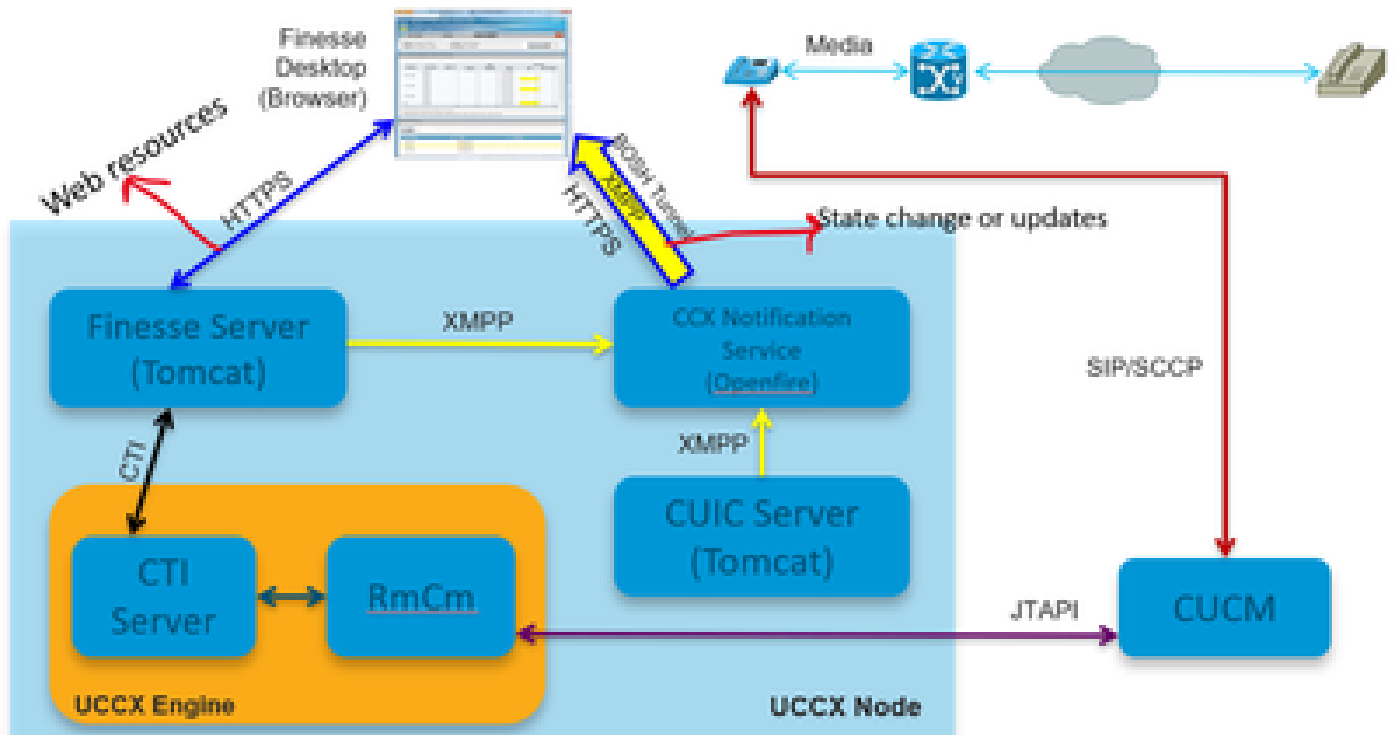
JTAPI

Cisco Unified JTAPI是Sun Microsystems開發的程式設計介面標準，用於基於Java的電腦電話應用。Cisco JTAPI透過其他Cisco擴展實施Sun JTAPI 1.2規範。UCCX和CUCM之間的任何通訊都駐留在JTAPI上。這就是CUCM和引擎（電話子系統）相互通訊的方式。

JTAPI用於控制和監控CUCM電話、使用CTI埠和路由點路由呼叫、在CUCM上開始和停止記錄以及任何呼叫路由功能

30000英尺檢視

下圖說明UCCX引擎、Finesse、CUCM和瀏覽器如何相互通訊。



30000英尺檢視

讓我們考慮與座席建立呼叫。現在，正在透過JTAPI監視代理擴展的RmCm向CTI伺服器告知代理正在通話的狀態更改。此資訊透過CTI從CTI伺服器 (CCX引擎內部) 傳送到Finesse伺服器(Tomcat)。Finesse伺服器使用XMPP將此資訊傳送到CCX通知服務有關狀態更改的資訊。通知服務(Openfire)會開啟通往代理程式瀏覽器的BOSH通道，並更新有關狀態變更的資訊，這也是您看見代理程式進入「保留」狀態的方式。使用WAR檔案、小工具等HTTPS (如果尚未在快取中)，向finesse伺服器請求任何型別的Web資源。

休眠

下圖說明了Hibernate服務。



休眠

HIBERNATE稱為高效能對象/關係永續性和查詢服務。簡而言之，它將JAVA類對映到資料庫表。例如，您有一個名為Team的JAVA對象，而您有一個名為Team的finesse資料庫中的資料庫表。JAVA類控制表內的資訊，而HIBERNATE則控制該資訊。它不使用SQL查詢，而是使用Java類來更新資訊。

AXL

管理XML。

XML代表「可延伸標籤語言」，是一種標籤語言，定義一些相對簡單的規則來編碼資料。它主要是用來以兩個系統都能理解的明確定義格式傳輸和接收結構化資料。在最基本的表單中，XML會定義以尖括弧(<>)括住的標籤，這些標籤會括住標籤所描述的資料。標籤可與其他標籤內含的標籤組成階層。例如，要定義基本電話裝置，您可以說電話裝置需要三個引數：名稱、說明和電話號碼。

它是基於SOAP的API，可在CUCM上啟用遠端調配。它用於增加、更新、刪除或檢索CUCM資料庫中的資訊。檢索功能包括檢查使用者身份驗證和運行SQL查詢。AXL API為您提供對整個CUCM資料庫的訪問。AXL API純粹用於調配，不提供對運行時或效能資料的訪問。

AXL API使用HTTPS基本身份驗證。如果任何CUCM使用者（應用程式或終端使用者）是標準CCM超級使用者訪問控制組或具有標準AXL API訪問角色的任何組的成員，則他們具有透過AXL進行讀/寫訪問。這表示所有超級使用者帳戶已經隱含地擁有AXL API的存取權。要建立專用於AXL API使用的帳戶，您必須首先建立一個訪問控制組並向其分配標準AXL API訪問角色，然後將應用程式使用者與新建立的組關聯。若要提供唯讀AXL API存取，您可以建立個別的存取控制群組，並只為其指派「標準AXL唯讀API存取」角色。

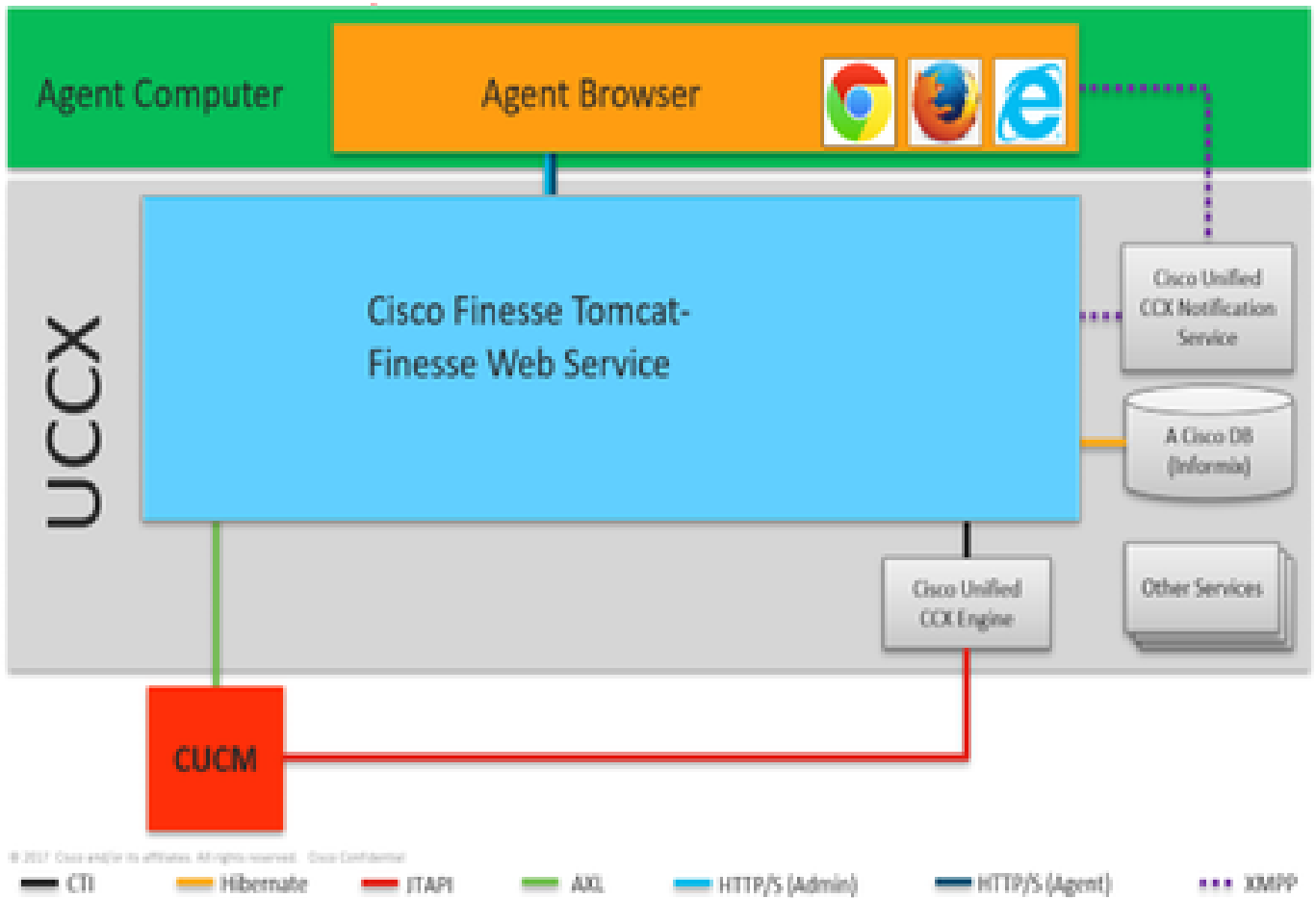
肥皂

簡單對象訪問協定(SOAP)是一種以XML格式在應用程式之間傳遞資訊的方法。SOAP消息通常透過HTTP會話從傳送應用程式傳輸到接收應用程式。實際的SOAP訊息是由Envelope元素所組成，其中包含Body元素和可選Header元素。

- Envelope - 此強制元素是SOAP消息的根，它標識傳輸的XML是SOAP資料包。信封包含主體部分和可選信頭部分。
- Header - 此可選元素提供指示消息處理資訊的擴展機制。例如，如果使用郵件的操作需要安全憑據，則這些憑據必須是信封報頭的一部分。
- Body - 此元素包含消息負載，即傳送和接收應用程式之間傳輸的原始資料。主體本身可由多個子元素組成，XML架構通常定義此資料的結構。

20000英尺檢視

下圖詳細解釋了Finesse架構中涉及的協定。

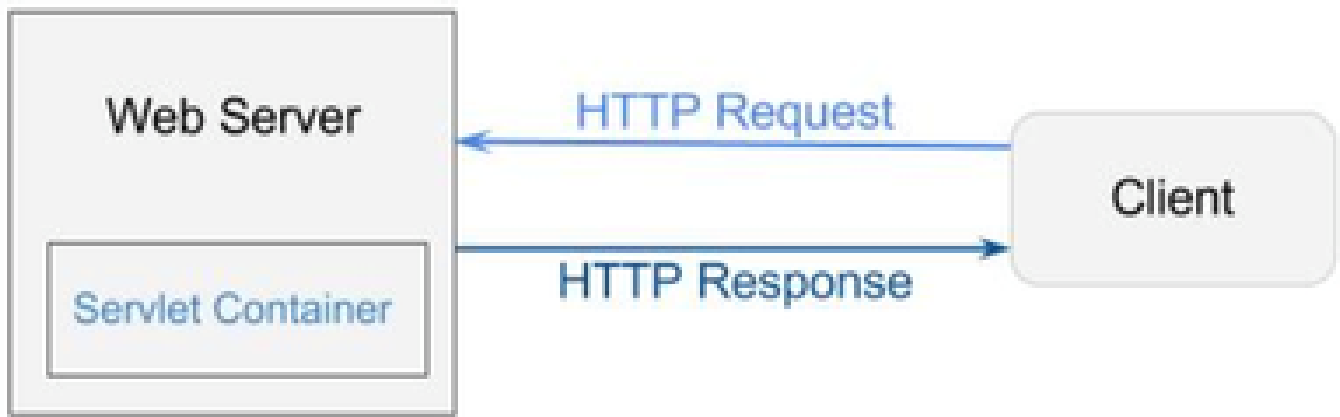


20000檢視

這些協定負責不同UCCX元件之間的通訊。

- UCCX引擎和Finesse Server透過CTI協定進行通訊。
- UCCX引擎和CUCM JTAPI通話協定。
- Finesse Tomcat和CUCM透過AXL進行通訊。
- Finesse通知服務和Agent瀏覽器在XMPP/BOSH上的通話。
- Finesse Tomcat和資料庫透過Hibernate進行通訊。
- Finesse Tomcat和Finesse Notification透過XMPP進行通訊。

APACHE SHINDIG



興迪格

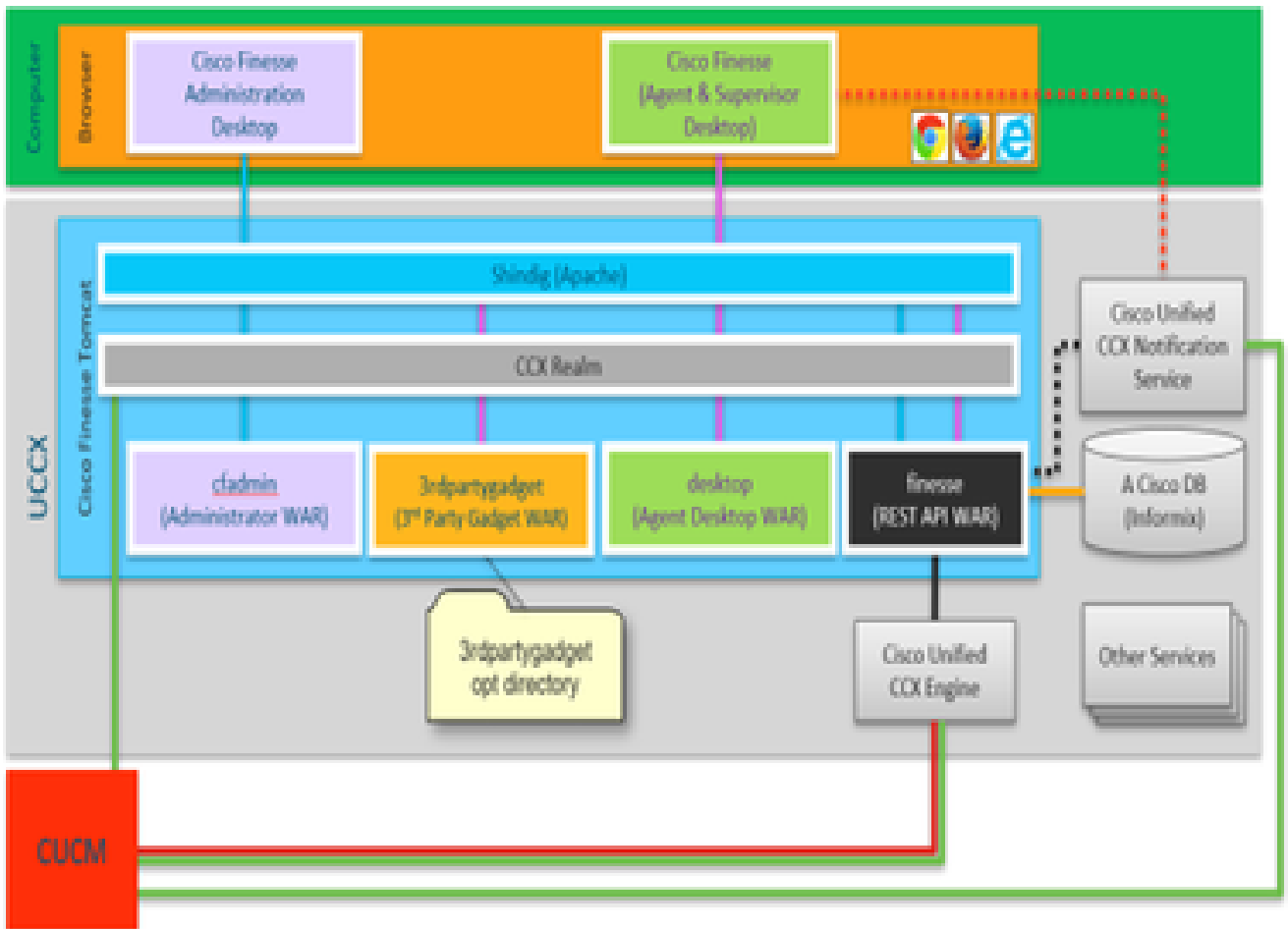
Apache Shindig是一個OpenSocial容器，透過提供用於呈現小工具、代理請求以及處理REST和RPC請求的代碼，幫助您快速開始託管OpenSocial應用。OpenSocial是一組API，用於構建在Web上運行的社群應用程式。(Web/Servlet) Web伺服器使用「容器」來動態產生網頁。

WAR檔案

WAR代表Web Archive。它包含Web專案的檔案。它可以有servlet、XML、JSP、image、HTML、CSS、JS等。Catalina日誌包含有關正在部署的WAR的資訊。

10000英尺檢視

下圖詳細說明了身份驗證流程在UCCX和Finesse元件中的工作方式。



© 2017 Cisco and/or its affiliates. All rights reserved. Cisco Confidential

■ CTI (SED-188) ■ Hibernate ■ JTAPI ■ AXL ■ HTTP/S (Admin) ■ HTTP/S (Agent) ■■■ XMPP (BOSH) ■■■ XMPP

10000檢視

視您登入的方式而定，需要使用WAR檔案來顯示和建立頁面。瀏覽器要求Shindig需要呈現小工具，Shindig然後與CUIC通話以呈現小工具。CCX領域用於使用AXL對CUCM進行身份驗證。通知服務還使用AXL透過CUCM進行身份驗證。

Finesse Rest API WAR是實際與通知服務、CCX引擎和資料庫通訊的主要儲存庫。Shindig僅與Finesse Rest API (WebServices)通話，因為cfadmin和案頭WAR僅用於顯示頁面。任何與Finesse Rest API WAR相關的內容，您都可以在Finesse WebServices日誌中看到，這些日誌是finesse最重要的日誌。您在Shindig和Finesse Web服務之間使用HTTP (Rest API WAR)。Finesse Web服務(Rest API WAR)和引擎透過CTI相互通訊。

AJAX - Finesse之美

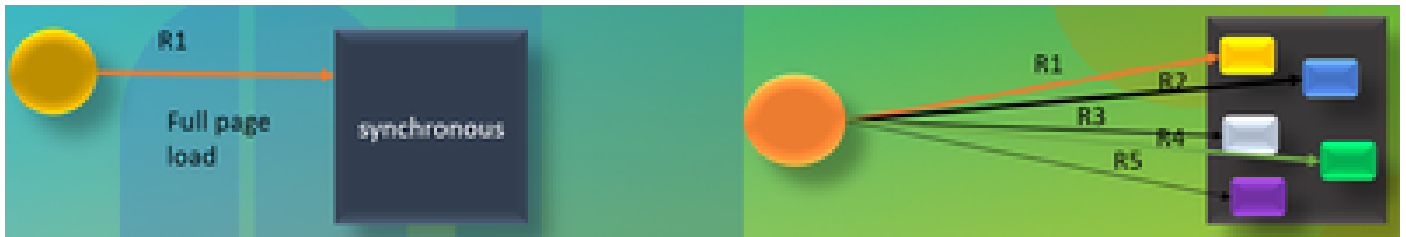
AJAX代表非同步Javascript和XML。它不是一種程式語言，而是一種從網頁訪問Web伺服器的方法。AJAX是一種進行部分頁面更新的機制。它可讓您使用來自伺服器的資料來更新頁面的區段，而不需要重新整理整個頁面。

例如，如果您談到Facebook messenger，當有新消息傳入時，您無需刷新整個頁面即可獲取該消息，相反，頁面本身的消息部分會刷新並即時獲取新消息，而無需刷新整個頁面。

每個瀏覽器都有一個稱為XMLHTTPREQUEST(也稱為XHR)的內建對象。對伺服器中AJAX的每個請求都會透過此XML請求。這包含您需要更新的特定資訊。

使用AJAX的優點

下圖說明了非同步請求和同步請求之間的差異。



AJAX

如果是同步請求，您必須等待第一個請求得到處理，然後可以傳送第二個請求。例如，需要重新整理頁面，您必須先重新整理頁面，才能執行任何動作。另一方面，如果是非同步請求，則不必等待第一個請求完成即可傳送第二個請求。您可以同時傳送多個請求。例如，網站上的天氣應用小工具。您可以刷新頁面的天氣部分，同時也可以同時處理網站的其他部分，而無需刷新整個頁面。這是非同步要求的主要優點。

使用AJAX

AJAX是XMLHTTPREQUEST (XHR)的組合，用於從Web伺服器傳送和接收更新，以及用於顯示或使用資料的Javascript和HTML。

使用AJAX向伺服器傳送請求

這是一個三步過程，下面將介紹這一過程：

1. 建立變數，並將XHR物件儲存在變數中。

```
Var 請求=新的XMLHttpRequest();
```

2. 訪問在XHR對象內部具有負載的請求變數。

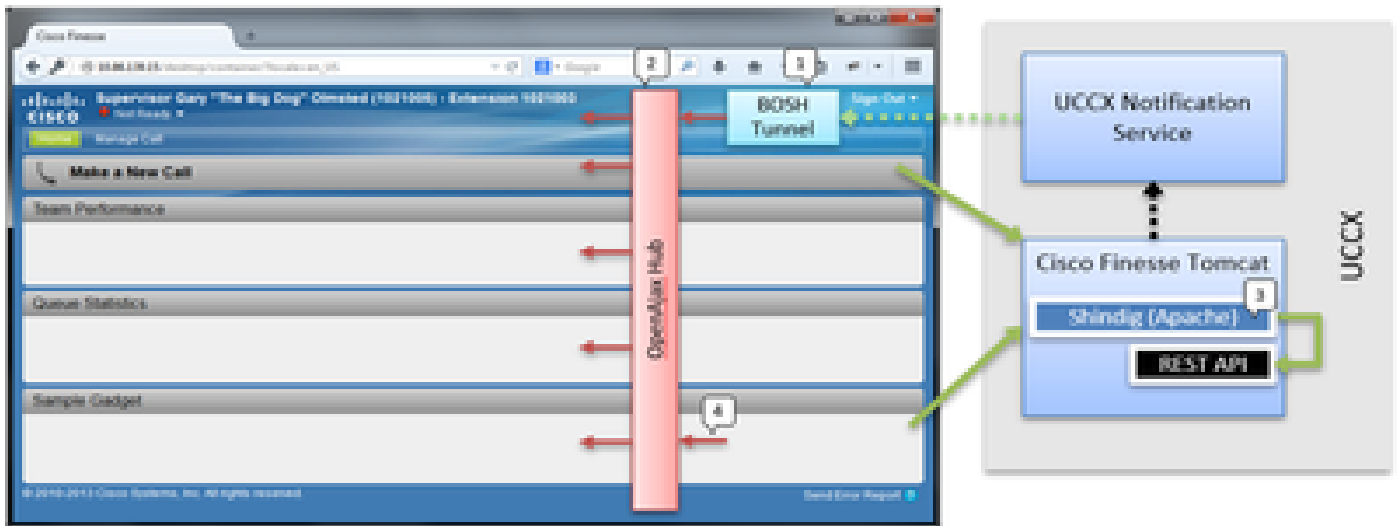
```
請求.open(GET, URL);
```

3. 傳送請求

```
Request.send();
```

案頭體系結構

下圖說明網頁呈現小工具時AJAX訊號的流程。



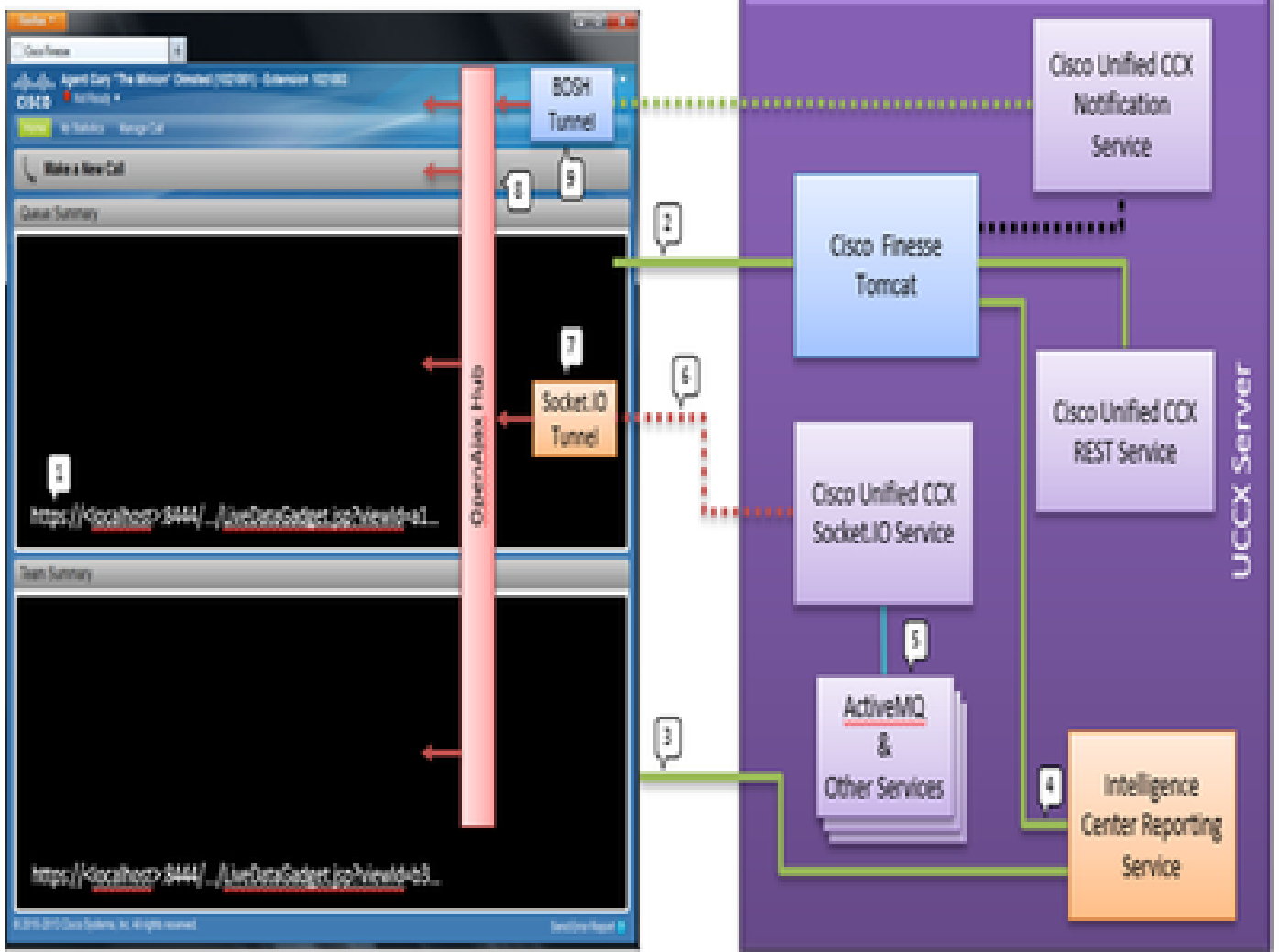
— CTI (GED-188)
 — HTTP/S
 ... XMPP (BOSH)
 ... XMPP
 ← OpenAjax

案頭體系結構

IFrame駐留在容器中，用於託管BOSH隧道。提供OPENAJAX集線器以跨小工具發佈消息（使用pubsub方法）REST請求也透過Shindig代理到其他伺服器。小工具可以在AJAX中心上發佈自己的消息。

小工具架構

下圖詳細解釋了Finesse小工具架構。



— HTTP/S
 - - - XMPP (BOSH)
 - - - XMPP
 ← OpenAjax
 - - - Socket.IO
 — IMS

小工具架構

與典型的小工具不同，CUIC小工具還從OpenFire接收即時XMPP源。在UCCX中，CUIC和Finesse與UCCX共存，則有一個共用的OpenFire例項。大多數Gadget內容和所有REST API都是透過Finesse伺服器中的Shindig代理的。這適用於Finesse小工具和REST API，以及CUIC小工具例項和REST API。CUIC小工具使用D-Grid來呈現報表。必須執行引導過程，此操作將直接與CUIC一起完成。因此，CUIC小工具在載入過程中最初直接與CUIC伺服器通訊。因此，使用者瀏覽器必須接受CUIC證書（除了Socket.IO隧道）。小工具內容和REST API在Finesse和CUIC之間代理到客戶端。Rest API呼叫同時對Intelligence Center Reporting Service和CCX Web Service執行。CCX Live Data Socket.IO服務透過JMS從ActiveMQ獲取來自Live Data的消息。CCX Live Data Socket.IO服務透過來自客戶端的Socket.IO連線發佈即時報告JSON。與Finesse Desktop的BOSH Tunnel iFrame保持與Cisco Finesse Notification Service的BOSH連線類似，主即時資料小工具也有一個Socket.IO Tunnel iFrame，它保持與CCX Live Data Socket.IO服務的Socket.IO (Websocket)連線。

OpenAjax中心將所有事件分發到訂閱的偵聽程式。這既可以是小工具，也可以是Finesse容器本身的一部分。Finesse案頭有一個BOSH隧道iFrame，用於維護與Cisco Unified CCX通知服務的BOSH連線。這會將事件發佈到OpenAjax中心。

參考連結

- [Finesse Web服務開發人員指南](#)

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。