Cisco Finesse Failover Mechanisms

CTI Failover

Benchmark Parameters: For a Contact Center with the capacity of 2000 logged in agents and 6000 to 12000 configured agents, it takes up to 120 to 150 seconds (with up to 200 milliseconds WAN delay) for the Finesse server to recover its state during the CTI Failover in a voice only deployment. Failover involving Digital Channels can take longer based on the tasks and MRDs configured. The Finesse client desktop failover will be initiated after the Finesse server is back IN_SERVICE and can take a few more minutes.

CTI failover is when the Finesse server disconnects from one CTI server and reconnects to the same or another CTI server.

The prerequisites for successful CTI failover are as follows:

- Unified Contact Center Enterprise (Unified CCE) must be configured in duplex mode.
- The B Side CTI host and port must be configured through the Finesse administration console.

In the duplex mode, if Finesse loses connection to CTI server, it attempts to connect to the server which is running. Finesse alternates between the configured servers until it makes a successful connection.

While failover is in progress, Finesse transitions to OUT_OF_SERVICE state. During this period, Finesse does not entertain client requests or send out events. Any request made during this time receives a 503 Service Unavailable error message.

After reconnecting to a CTI server and transitioning to IN_SERVICE state, Finesse responds to client requests and publishes events.

Connection to the CTI server can be lost due to the following reasons:

- Finesse misses three consecutive heartbeats from the connected CTI server.
- Finesse socket that is opened to the CTI server fails.

After the failover is complete, the last state of call control, call data, or agent state are published as events to all clients. This allows Finesse clients to reflect an accurate view of the call control, call data, and agent state.

If an agent either makes or answers a call, and then ends that call during failover (that is, the entire call takes place during failover), the corresponding events are not published.

Note An agent or supervisor who signs in after being on an active conference with other devices (which are not associated with another agent or supervisor) may experience unpredictable behavior with the Finesse desktop due to incorrect call notifications from Unified CCE. These limitations also encompass failover scenarios where a failover occurs while the agent or supervisor is participating in a conference call. For example, an agent is in a conference call when the Finesse server fails. When the agent is redirected to the other Finesse server, that agent may see unpredictable behavior on the Finesse desktop. Examples of unpredictable behavior include, but are not limited to, the following:

- The Finesse desktop does not show all participants in a conference call.
- The Finesse desktop does not show that the signed-in agent or supervisor is in an active call.
- The Finesse receives inconsistent call notifications from Unified CCE.

Despite these limitations, the agent and supervisor can continue to perform general operations on the phone. Desktop behavior returns to usual after the agent or supervisor drops off the conference call.

Prevent Non-Voice Task RONAs during CTI Reconnect

When CTI disconnection happens, the agent state is changed to WORK, on the respective non-voice Media Routing Domain (MRD), to prevent tasks getting routed to the disconnected agents. Previous releases of Unified CCE used to change the agent states back to an available state when the CTI connection is re-established, even though the media handling gadgets and the media channels are not initialized by then.

The media handling gadgets, and the media channels are initialized only after the Finesse desktop failover completes.

Due to the significant delay in desktop failing over after the Finesse server reconnects to the CTI server, chances of occurrence of RONA (Redirection on No Answer) are high when dealing with non-voice tasks.

Unified CCE, Release12.5 (1) or later allows the agent state to remain in WORK mode after CTI reconnection. This allows the agents to change to an available state in non-voice MRD explicitly after the Finesse desktop and media channels are initialized. This avoids the task being routed to the user before the agent is ready to handle non-voice media tasks.

By default, Cisco Finesse Release 12.5(1) retains the earlier behavior, which can be modified using the **enableAutoWorkModeStateChange** property. By default, this property is set to *true*, and the administrator can set to *false* to change to the new behavior.

Note This behavior is supported from Unified CCE Release 12.5(1) onwards, and only after the relevant non-voice gadgets or custom desktop or clients support this behavior.

The agents remain in the WORK mode until they are explicitly set to active on the respective MRD using the REST API. This informs the CTI that the media channel is available (and connected) and the tasks can be routed to the respective user on that MRD.

The `Media—Change Agent from Work State to Active` API allows a user to change the agent state from WORK state to active (READY or NOT_READY), which is automatically computed by Unified CCE. Users can only use this API when an agent state is WORK.

AWDB Failover

The prerequisites for AWDB failover are as follows:

- The secondary Administrative Workstation Database (AWDB) is configured.
- The secondary AWDB host is configured through the Cisco Finesse administration console.
- Cisco Finesse can connect to the secondary AWDB host.
- The Distributor service is running on the secondary AWDB host.

Agents and supervisors are authenticated against the AWDB database. When an agent or supervisor makes a successful API request (such as a sign in or call control request), the credentials are cached in Cisco Finesse for 30 minutes from the time of the request. After a user is authenticated, that user continues to be authenticated until 30 minutes pass, even if both AWDBs are down. Cisco Finesse attempts to reauthenticate the user against the AWDB only after the cache expires.

If Cisco Finesse loses connection to the primary Administration & Data server, and the preceding prerequisites have been implemented, AWDB failover occurs. After Cisco Finesse loses connection to the primary Administration & Data server, it tries to reconnect to the secondary server.

Cisco Finesse repeats this process for every API request until it can connect to one of the Administration & Data servers. During failover, Cisco Finesse does not process any requests, but clients can still receive events.

If Cisco Finesse cannot connect to either of the Administration & Data servers and the cache has expired, the systems returns the following errors:

- Agents and supervisors who attempt to sign in to the Finesse desktop receive an “Invalid user ID or password” error message.
- Administrators cannot update or retrieve settings in the Cisco Finesse administration console.
- Users who are already signed in to Cisco Finesse receive an “Operation timed out” error message.
- Users who make API requests receive an 401 “Unauthorized” HTTP error message.

If Cisco Finesse loses connection to one AWDB and then receives requests, these requests may time out before Cisco Finesse can detect that the connection is down and connect to the alternate AWDB. In this scenario, the user (administrator, agent, or supervisor) may need to retry the operation for it to succeed.

Finesse Desktop Failover

With a two-node Finesse setup (primary and secondary Finesse servers), if the server that an agent is connected to goes out of service, the agent receives a notification that the connection with the server was lost. The Finesse desktop:

- Continues to check whether the current Finesse server recovers its state.
- Checks if the other Finesse server is available and in service.

If the other Finesse server is available, the desktop automatically signs the agent into the other server. If the current Finesse server recovers its state, the desktop notifies the agent that it has reconnected.

The Finesse smarter failover logic has three triggers to detect desktop failure:

- The Finesse desktop receives a SystemInfo event that the current server is OUT_OF_SERVICE.
- The user XMPP connection is disconnected.
- The “finesse” XMPP user presence changes to unavailable.

No matter which trigger is detected, the desktop reconnection logic is as follows:

1. Poll SystemInfo for current server every 20 seconds and the other Finesse server between 45-150 seconds.
2. If SystemInfo is IN_SERVICE, check the user XMPP connection.
3. If SystemInfo is IN_SERVICE, check if the lastCTIHeartbeatStatus is a success.

Note This is to ensure that the second side is healthy before failover, and does not immediately go out of service after the client has failed over . This may occur in CTI failure, since both Finesse servers connect to the same PG and CTI server, and a CTI failure can cause both Finesse servers to disconnect and connect to the alternate PG. Depending on the network topology the second server might be slower to sense a network disconnect.

4. If XMPP is disconnected, make an user XMPP connection request.
5. If user XMPP is connected and the server is IN_SERVICE, refresh the data.

While polling SystemInfo every 20 seconds, the desktop also checks the availability of the alternate server every 45-150 seconds. The smarter failover logic prefers to stay with the current server. If the failover logic detects that the alternate server is available, it checks the current server one more time. If the current server has recovered, the desktop reconnects to the current server. If the current server is still down, the desktop connects the agent to the alternate server. In this case, the agent does not automatically reconnect to the failed server after it recovers but instead remains connected to the alternate server.

If the user XMPP connection is the source of failure, the desktop makes three attempts to reconnect before changing the state of the desktop to disconnected. These attempts occur before the smarter failover logic begins.

Desktop failover can occur for the following reasons:

- The Cisco Finesse Tomcat Service goes down.
- The Finesse Webapp Service goes down.
- The Cisco Finesse Notification Service goes down.
- Finesse loses connection to both CTI servers.

Note After Finesse failover, the pending state of an agent will not be displayed once the agent fails over to the secondary Finesse node. The pending state change is reflected on the desktop only after the call ends.

Desktop Behavior

Cisco Finesse sends a code of 255 to the CTI server and you may see a different code on the CTI server side. The actual behavior of the desktop under these conditions depends on the setting for Logout on Agent Disconnect (LOAD) in Unified CCE. By default, the CTI server places the agent in Not Ready state.

Note Finesse takes up to 120 seconds to detect when an agent closes the browser. If the browser crashes, Finesse waits 60 seconds before sending a forced logout request to the CTI server. Under these conditions, Finesse can take up to 180 seconds to sign out the agent.

The following table lists the conditions under which Finesse sends this code to the CTI server.

Scenario	Desktop Behavior	Server Action	Results
The agent closes the browser, the browser crashes, or the agent clicks the Back button on the browser.	Finesse desktop makes a best-effort attempt to notify the server.	Finesse receives a presence notification of *Unavailable* from the client. Finesse waits 60 seconds, and then sends a forced logout request to the CTI server.	**Race Conditions** **1.** The agent closes the browser window. Finesse receives a presence notification of *Unavailable* for the user. Finesse tries to sign the agent out; however, that agent is already signed out. **2.** If the browser crashes, it can take the Finesse server up to 120 seconds to detect that the client is gone and send a presence notification to Finesse. A situation can occur where the client signs into the subscriber before the publisher receives the presence notification caused by the browser crash. In this case, the agent may be signed out or put

			into Not Ready state on the subscriber. **3.** If the Finesse desktop is running over a slower network connection, Finesse may not always receive an *Unavailable* presence notification from the client browser. In this situation, the behavior mimics a browser crash, as described in the preceding condition. **4.** If agent is in Not Ready State before Failover, agent moves to Not Ready — Connection Failure after CTI Disconnect or Reconnect. If agent is in Ready State before Failover, agent moves to Not Ready — Connection Failure upon his next state change to Not Ready.
The client refreshes the browser	—	Finesse receives a presence notification of *Unavailable* from the client. Finesse waits 60 seconds before sending a forced logout request to the CTI server to allow the	—

		browser to reconnect after the refresh.	
The client encounters a network glitch (Finesse is IN_SERVICE)	Connection to the Finesse server temporarily goes down, consequently the client fails over to the subscriber.	The publisher receives a presence notification of *Unavailable* from the client. Finesse is IN_SERVICE, so it sends a forced logout request to the CTI server for the agent.	**Race Conditions** A situation can occur where the forced logout does not happen before the client signs in to the subscriber. If the agent is on a call, the publisher sends the forced logout request after the call ends. The agent will be signed out or put into Not Ready state when the call ends, even though the client is already signed in to the subscriber. If agent is in Not Ready State before Failover, agent moves to Not Ready — Connection Failure after CTI Disconnect or Reconnect. If agent is in Ready State before Failover, agent moves to Not Ready — Connection Failure upon the next state change to Not Ready.
The Refresh Token has expired. For more information on tokens, see https://developer.cisco.com/docs/finesse/#single-sign-on-apis.	Finesse desktop sends a forced logout request to the CTI server.	The Finesse server forwards the forced logout request to the CTI server.	The session expiry warning appears 10 minutes and 5 minutes before the Refresh Token expires. In the last minute, a countdown timer appears till the Refresh Token expires. The agent is forcefully logged out when the timer reaches zero and must log in again. For Unified CCE, the state of the agent changes to Log Out or

			Not Ready based on the Load parameter set as below. **Load parameter = 0** • When the agent's current state is Not Ready, Ready, or Wrap-Up, the agent's state after force logout is changed to Not Ready – Connection Failure. • When the agent's current state is Talking, the Agent goes into Not Ready – Connection Failure state after the call ends. **Load parameter = 1** • When the agent's current state is Not Ready, Ready, or Wrap-Up, the agent goes to Logged Out – System Failure. • When the agent's current state is Talking, the Agent goes to Logged Out – System Failure immediately even though the call is still active.

Desktop Chat Failover

The following table lists the desktop chat failover scenarios:

Failover Type	Desktop Chat Behavior
Cisco IM&P server failover	The desktop chat status is retained, and all active chat sessions are lost.
Finesse server failover	The desktop chat status is retained, and all active chat sessions are lost.
CTI server failover	The desktop chat status and all chat sessions are retained.

Finesse IP Phone Agent Failover

Finesse IPPA failover can occur for the following reasons:

- The Finesse REST API Service transitions to OUT_OF_SERVICE.
- The Finesse Notification Service transitions to OUT_OF_SERVICE.
- If Finesse IPPA detects a server failure before Finesse fails over to the alternate CTI server, then Finesse IPPA declares the Finesse server OUT_OF_SERVICE.

The server that an agent is connected transitions to OUT_OF_SERVICE, the Finesse IP Phone Agent (IPPA) displays a notification that the server is unavailable. Finesse IPPA continues to check whether the current Finesse server recovers its state and notifies the agent if it reconnects.

Finesse IPPA attempts to reconnect to the server every 5 seconds and declares it OUT_OF_SERVICE after three failed attempts. The total time required for the transition to OUT_OF_SERVICE is approximately 15 seconds.

Unlike the Finesse desktop, Finesse IPPA does not check whether the subscriber is available. To connect to subscriber, the agent must exit the publisher, and manually sign into the subscriber.

Finesse IPPA failover logic has the following two triggers to detect failure:

- Finesse IPPA receives a SystemInfo event that the publisher is OUT_OF_SERVICE.

  Finesse IPPA polls SystemInfo every 5 seconds to check whether the Finesse server is IN_SERVICE. After three attempts, if the Finesse server is not IN_SERVICE, Finesse IPPA displays a server unavailable message to the agent.

- Finesse IPPA receives notification that the Finesse Notification Service is disconnected.

  Finesse IPPA tries every 5 seconds to reconnect with the XMPP server. After three attempts, if the Finesse Notification Service cannot be reestablished, Finesse IPPA displays a server unavailable message to the agent.

While the agent is still signed into the current service, Finesse IPPA continues attempting to reestablish the connections with the Finesse and XMPP servers. If they both resume service, Finesse IPPA displays the **Sign In** screen and the agent can sign in again and continue as usual.

Alternately, the agent must exit the current Finesse service and try to connect using an alternate Finesse service.