



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

eServices Multitenancy and Load Balancing Guide

Web API Server

5/3/2025

Contents

- 1 Web API Server
 - 1.1 Load-Balancing Configuration for Web API Server
 - 1.2 Load-Balancing API

Web API Server

Web API Server uses Solution Control Server (SCS) to monitor the state of all servers. Load balancing that involves Web API Server includes:

- Load balancing between instances of Web API Server.
- Load balancing between instances of the following servers: Callback Server, Chat Server, E-mail Server, Interaction Server, Stat Server, UCS.

Load-balancing components react to:

- Changes in the configuration; specifically, the connection settings of Web API Server and the status (enabled or disabled) of relevant applications.
- Application states (running or stopped) as reported by SCS.

Load balancing performs the following actions:

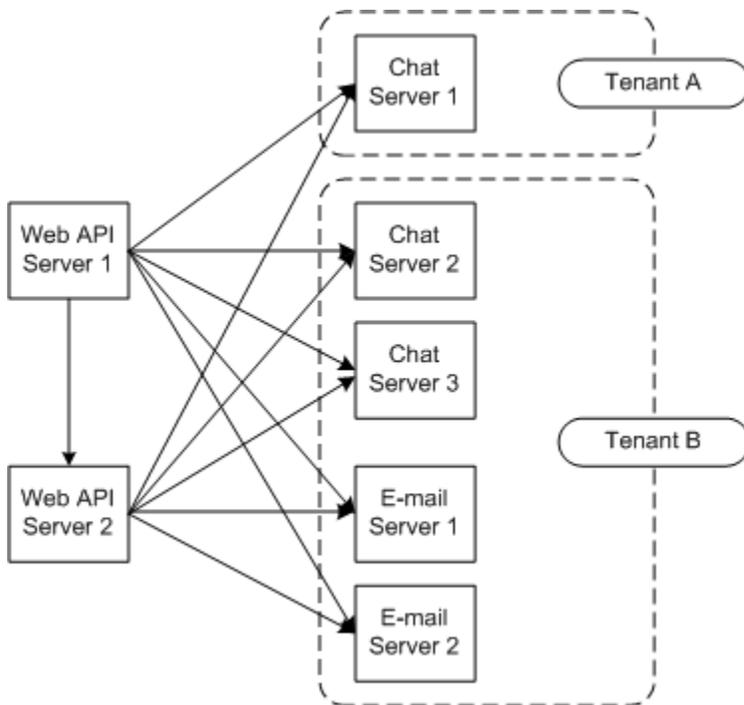
- For all instances of servers that are listed on Web API Server's Connections tab:
 1. The instance reports its status as RUNNING to SCS when it starts. The instance is then ready to process interactions.
 2. The load balancer then marks the server's status as RUNNING.
 3. A web application can now use the service provided by that instance.
- If the server's status changes to Service Unavailable:
 - It is excluded from the list of available servers and cannot be given in response to a web application's request.
 - If the server's status changes again to Started (Service Available), it returns to the list of available servers. In the case of Chat Server, it can continue to handle an online session that was established before the status changed to Service Unavailable.
- If the server shuts down:
 - It is excluded from the list of available servers and cannot be given in response to a web application's request.
 - All interactions that it was handling are either lost (in the case of Chat Server) or handled by other instances of the same type of server (E-mail Server).
- If the server is disabled in the Configuration Layer:
 - It is likewise excluded from the list of available servers and cannot be given in response to a web application's request.
 - None of the clients that are already working with this instance of the server are affected.
 - As soon as all interactions that the server is handling end, the server can be shut down.
- If the server is enabled, it returns to the list of available servers.

Load-Balancing Configuration for Web API Server

In the example eServices configuration shown in below, there are two instances of Web API Server, two of E-mail Server, and three of Chat Server, with the Chat Server instances divided between two tenants. All application servers are available to both Web API Servers.

Important

To enable cross-Tenant load balancing, you must add the Tenants to the Tenants tab of the Web API Server that you want to serve as the point of load balancing.



Load Balancing Configuration

To simplify the overall configuration, you can use application objects of type `Application Cluster` to group available servers in the configuration. An `Application Cluster` is a configuration object that stores connection specifications. The eServices configuration wizard offers the opportunity to create `Application Clusters`. If you did not create an `Application Cluster` while running the wizard, you can add one manually, using the following procedure.

Configuring an Application Cluster

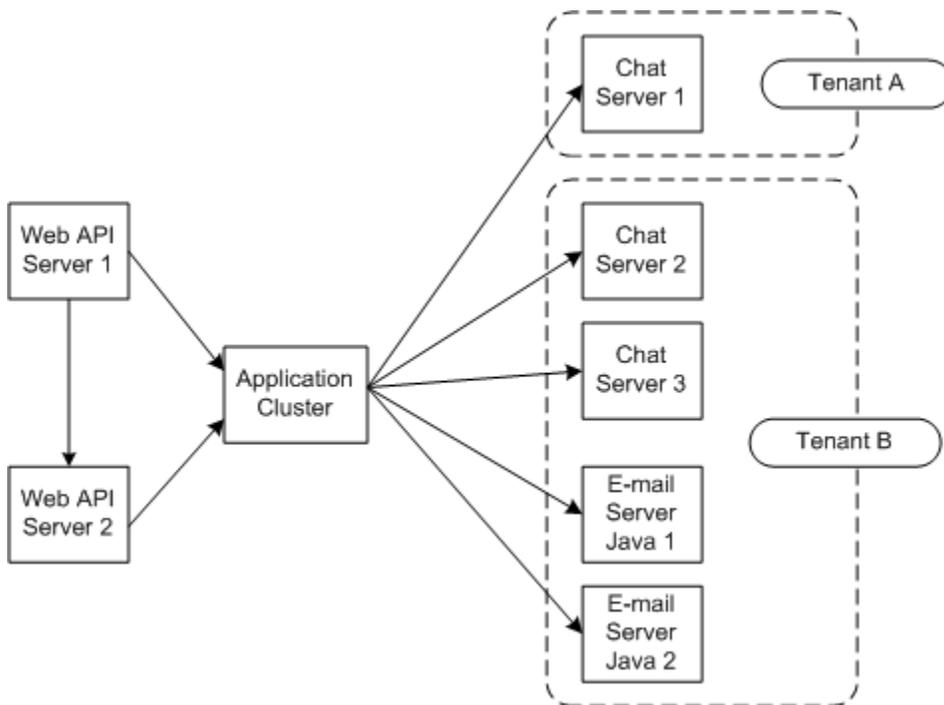
Start

1. In Configuration Manager, locate the template `ApplicationCluster_<version-number>.apd` in the `templates` directory of your product CD and import it.
2. Use the template to create a new Application Cluster.
3. On the `Connections` tab, add connections to the servers that Web API Server will balance.
4. On the `Server` and `Start Info` tabs, enter arbitrary characters in the empty fields. Web API Server ignores this information, but there must be something in these fields for you to be able to save the Application.
5. In your Web API Server Application object or objects, add a connection to the Application Cluster.

End

Application Clusters are transparent to load-balancing components, and the configuration that uses connections through Application Clusters is equivalent to a configuration that uses direct connections between servers.

The configuration shown in the previous section is reconfigured using an Application Cluster in the figure below.



Load Balancing with Application Cluster

Load-Balancing API

To use eServices 8.1 load-balancing capabilities, use the Load-Balancing API in the web application. The Load-Balancing API provides the following functionality:

- It can select a particular server instance from a set of instances of the specified server type.

- Upon the first request to a server instance, it can create an alias for the selected server instance and store it for future use.
- It can use the alias to obtain connection parameters (host name and port) of the server instance.
- It has access to configuration information.

The eServices 8.1 simple samples demonstrate how to use the eServices Load-Balancing API to develop web applications that use load balancing. For details, see the “About Web API Clients” chapter in the eServices Web API Client Developer’s Guide.