Operating Servoy Application Servers in a Cluster

Servoy Application Servers that operate within a Servoy Cluster can, for the most part be seen as normal Servoy Application Servers. This chapter describes the differences and the things to look out for.

Servoy Application Server instances that are part of a Servoy Cluster share the same Servoy Repository, databases and databroadcasting mechanism. The Servoy Application server is also aware of the clients connected to the other servoy Application servers in the cluster. This means the following:

Solution Imports

Import of solutions into a Servoy Application Server connected to the Servoy Cluster will broadcast to all connected Servoy Application Servers. When using the pre- and post import hooks ic.w. the Maintenance plugin, all operations performed by the Maintenance plugin will operate on the entire cluster.

Databases & Databroadcasting

All the Servoy Application Servers in the cluster need to connect to the same set of databases. Although the connection settings might differ, the physical database to which a Database Server connects needs to be the same, otherwise the databroadcasting mechanism will work incorrectly.

Servoy Admin page Clients overview
The Clients overview page on the Servoy Admin page will show all clients connected to the cluster, not just the clients connected to the specific Servoy Application Server instance. The Clients overview provides a link to toggle between grouping by Servoy Application Server first, then solution or vise versa. Flushing clients, sending messages and killing clients can be done on individual client level, per solution, per server or per solution per server.

The following things need to be taken into account when operating Servoy Application servers within a Servoy Cluster:

Server Plugins

Server plugins live on the server side. Each Servoy Application Server in the cluster will create and use an instance of the plugin. Unless specifically build to operate within a Servoy Cluster (cluster-aware), the plugin will only interact with it's Servoy Application server. Depending on what the plugin does this might be ok or not.

The RestfulWS plugin for example, which operates partly serverside, has a serverside setting, detailing the maximum number of licenses used for handling concurrent requests. As the plugin gets instantiated on each Server Application Server, this value is the maximum number per Servoy Application Server.

When using 3rd party plugins, check with the vendor if the plugins support operating in a Servoy Cluster.

Properties

A Servoy Application Server stores all of it's settings in the serrvoy.properties file in <servoy>/application_server. When running inside a Servoy Cluster, each Servoy Application Server instance still has it's own servoy.properties file.

Many of the properties will have to have the same value, to provide the enduser with a consistent behavior, regardless of which Servoy Application Server they connect to.

Except for the useAjax setting on solutions and solutions itself, changes to the configuration of a Servoy Application Server instance are not broadcasted to the other Servoy Application Servers in the cluster. This means that configuration changes need to be applied to all Servoy Application Servers in the cluster manually.

Client Licenses

- Client licenses: the same Client License key needs to be installed on every Servoy Application Server. It will work as if the cluster uses this license, allowing a number of clients to be connected per cluster, not per Servoy Application Server.
- CPU licenses: each Servoy Application Server requires it's own CPU License key

Folder Contents

The following folders of the Servoy Application Server installation should be synced between all Servoy Application Server instances that are part of the cluster:

- {servoyInstall}/application_server/beans
- {servoyInstall}/application_server/drivers
- {servoyInstall}/application_server/lafs
- {servoyInstall}/application_server/plugins
- {servoyInstall}/application_server/server/webapps/ROOT/servoy-webclient/templates

✅ To identify individual Servoy Application Servers when loggin onto the Servoy Admin page, an an INSTANCE system property can be set in the wrapper.conf or servoy_server.sh/.bat of each server, to provide a unique identifier to each server. The value set for the INSTANCE system property shows up on the main page of the Servoy Admin page.

For example in servoy_server.sh/.bat add '-DINSTANCE=serverX' to the java command. the text 'serverX' will appear on the Servoy Admin page.