

# Odysseus.conf

This file is placed in folder `ODYSSEUS_HOME` and allows to configure a large set of different configurations in Odysseus

- `Tenant.DefaultName ()`

## Storage of Users, views and queries:

- `StoretypeUserMgmt`
  - `Memorystore`
  - `Filestore`
- `StoretypeDataDict`
  - `Memorystore`
  - `Filestore`
- `Filestore.StoreQueries`
  - `false`
  - `true`
- `tenantStoreFilename (tenants.store)`
- `userStoreFilename (users.store)`
- `privilegStoreFilename (privileges.store)`
- `roleStoreFilename (roles.store)`
- `slasFilename (slas.store)`
- `streamDefinitionsFilename (viewDefinitions.store)`
- `streamOrViewFromUserFilename (viewFromUser.store)`
- `viewDefinitionsFilename (logicalViewDefinitions.store)`
- `sinkDefinitionsFilename (sinkDefinitions.store)`
- `sinkDefinitionsUserFilename (sinkDefinitionsUser.store)`
- `entitiesFilename (entities.store)`
- `entityUsedByFileName (entityUsedBy.store)`
- `sourceTypeMapFilename (sourceTypeMap.store)`
- `entityFromUserFilename (entityFromUser.store)`
- `sourceFromUserFilename (sourceFromUser.store)`
- `queriesFilename (queries.store)`
- `queriesUserFilename (queriesUser.store)`
- `queriesBuildParamFilename (queriesBuildParam.store)`
- `storedProceduresFilename (procedures.store)`
- `storedProceduresFromUserFilename (proceduresUser.store)`
- `datatypesFromDatatypesFilename (datatypes.store)`

## ReloadLog

- `storeReloadLog (true)`
- `reloadLogStoreFilename (odysseusHome+ "reloadlog.store")`
- `schedulingConfigFile (odysseusHome+ "scheduling.conf")`

## Session

- `sessionTimeout, (24 * 60 * 60000) in Milliseconds`

## Scheduling

- `debug_Scheduler (false)`
- `debug_Scheduler_maxLines, 1048476 + ""`;
- `scheduler_TimeSlicePerStrategy, 10 + ""`;
- `scheduler_simpleThreadScheduler_executorThreadsCount, (-1)`
- `scheduler_DebugFileName, "SchedulerLog"`;
- `scheduler_trainSize, 1 + ""`;
- `Scheduler.sleepAllowed, "false"`;
- `Scheduler.Simplethreaded.SourcesPerThread, "10"`;
- `Scheduler.Simplethreaded.NumberOfThreads, "10"`;
- `Scheduler.Simplethreaded.SourceThreadCount, "10"`;

## SLA

- `sla_history_size, 10000 + ""`; // Milliseconds
- `sla_updatePenaltyTime, 60000 + ""`; // Milliseconds

## new SLA

- `sla_starvationFreedomFuncName, "QueueSizeSF"`;
- `sla_prioFuncName, "max"`;

- sla\_starvationFreedomDecay", 0.1 + """);
- sla\_querySharingCostModel", "none");
- sla\_querySharing", Boolean.FALSE.toString());
- sla\_costFunctionName", "quadratic");
- sla\_pathTimeUpdateInterval", 10000 + """); // Milliseconds

## Odysseus Storing

- storing\_database", odysseusHome + "database.conf"

## AC

- ac\_memHeadroom", "0.4"
- ac\_cpuHeadroom", "0.4"
- ac\_standardMemCost", "4"
- ac\_standardCpuCost", "0.00002"
- ac\_operator\_useHistograms", "true"

## Event Dispatcher

- EventHandlerDispatcherPoolSize(10);

## Web-Service

- WebService.Endpoint1(<http://0.0.0.0>);
- WebService.Port (9669);
- WebService.Endpoint2 (/odysseus);
- WebService.MaxPort (10000);
- WebService.SSL (false);
- Webservice.SSL\_Client\_Authentication (false);
- webservice.queryconnect.sink.minport (10000);
- webservice.queryconnect.sink.maxport (20000);