Payara Server 5 Administration Cheat Sheet

The Payara® Platform - Production-Ready, Cloud Native and Aggressively Compatible.

Administration Cheat Sheet

Contents

Payara Server 5 Administration Cheat Sheet

This is a reverse dictionary for frequently used operations. See Payara Server Documentation for all commands and options.

Start and Stop Domain

Use Case	Operations
Start Domain	`asadmin start-domain [domain-name]` *1
Stop Domain	`asadmin stop-domain [domain-name]` *1 `Use Admin Console : Common Tasks > server (Admin Server) > Stop`
Restart Domain	`asadmin restart-domain [domain-name]` *1 `Use Admin Console : Common Tasks > server (Admin Server) > Restart`
View status of Domains	`asadmin list-domains`

*1 : Assumes **domain1** as domain name if *domain-name* is omitted.

Start and Stop Standalone Instance

There are two types of standalone instance:

- “Local” Instance is an instance that is running on the same machine as the DAS. (It’s using a ‘CONFIG’ node.)
- “Remote” Instance is an instance on ‘SSH’ node or ‘DCOM’ node that is on different machine running DAS.

Use Case	Operations
Start Local Instance	`asadmin start-local-instance [--sync={normal|full}] instance-name` *1 *2
Stop Local Instance	`asadmin stop-local-instance instance-name` *1
Restart Local Instance	`asadmin restart-local-instance instance-name` *1
Start Local or Remote Instance	`asadmin start-instance [--sync={normal|full}] instance-name` *2 Use Admin Console : Common Tasks > Instances
Stop Local or Remote Instance	`asadmin stop-instance instance-name` Use Admin Console : Common Tasks > Instances
Restart Local or Remote Instance	`asadmin restart-instance instance-name` Use Admin Console : Common Tasks > Instances
View status of Instances	`asadmin list-instances` Use Admin Console : Common Tasks > Instances

*1 : Only available when a domain is running.
*2 : --sync option only required when you want to force a synchronisation of an instance *e.g.* asadmin start-instance --sync=full instance1

Start, Stop, and Manage Deployment Group

Use Case	Operations
Create Deployment Group	`asadmin create-deployment-group deployment-group-name` Use Admin Console : Common Tasks > Deployment Groups
Delete Deployment Group	`asadmin delete-deployment-group deployment-group-name` *1 Use Admin Console : Common Tasks > Deployment Groups
View list of Deployment Groups	`asadmin list-deployment-groups` Use Admin Console : Common Tasks > Deployment Groups

Use Case	Operations
Add an Instance to Deployment Group	`asadmin add-instance-to-deployment-group --instance instance-name --deploymentgroup deployment-group-name` *2 Use Admin Console : Common Tasks > Deployment Groups
Remove an Instance from Deployment Group	`asadmin remove-instance-from-deployment-group --instance instance-name --deploymentgroup deployment-group-name` *2 Use Admin Console : Common Tasks > Deployment Groups
Start Deployment Group	`asadmin start-deployment-group deployment-group-name` Use Admin Console : Common Tasks > Deployment Groups
Restart Deployment Group	`asadmin restart-deployment-group deployment-group-name` Use Admin Console : Common Tasks > Deployment Groups
Stop Deployment Group	`asadmin stop-deployment-group deployment-group-name` Use Admin Console : Common Tasks > Deployment Groups

*1 : Does not affect the instances assigned to the Deployment Group.
*2 : Does not change the status of the instance.

Deploy and Undeploy Application

Use Case	Operations
Deploy Application (to DAS)	`asadmin deploy [--contextroot context-root] [--name=component-name] file-archive` Use Admin Console : Common Tasks > Applications
Deploy Application to Deployment Group	`asadmin deploy [--contextroot context-root] [--name=component-name] --target=deployment-group-name file-archive` Use Admin Console : Common Tasks > Applications
Deploy Application to an Instance	`asadmin deploy [--contextroot context-root] [--name=component-name] --target=instance-name file-archive` Use Admin Console : Common Tasks > Applications
Deploy Application to an Instance (already deployed on other instance)	`asadmin create-application-ref --target= instance-name component-name` Use Admin Console : Common Tasks > Applications
Deploy Application to multiple Instances	`asadmin deploy [--contextroot context-root] --name=component-name --target= instance-name-1 file-archive` `asadmin create-application-ref --target= instance-name-2 component-name` *1 Use Admin Console : Common Tasks > Applications
Undeploy Application (from DAS)	`asadmin undeploy component-name` Use Admin Console : Common Tasks > Applications
Undeploy Application from Deployment Group	`asadmin undeploy --target=deployment-group-name component-name` Use Admin Console : Common Tasks > Applications
Undeploy Application from an Instance (remove application completely)	`asadmin undeploy --target=instance-name component-name` Use Admin Console : Common Tasks > Applications

Use Case	Operations
Undeploy Application from an Instance (still deployed on other Instances)	`asadmin delete-application-ref --target=`*`instance-name component-name`* `Use Admin Console : Common Tasks > Applications`
Undeploy Application from Multiple Instances	`asadmin delete-application-ref --target=`*`instance-name2 component-name`* [*2] `asadmin undeploy --target=`*`instance-name1 component-name`* `Use Admin Console : Common Tasks > Applications`
View Deployed Applications	`asadmin list-applications [`*`target`*`]` [*2] `Use Admin Console : Common Tasks > Applications`
View Deployed Applications (only on Instances)	`asadmin list-application-refs [`*`target`*`]` [*2] `Use Admin Console : Common Tasks > Applications`

*1 : Repeat for all instances that are no longer needed to run the application.
*2 : Assume setting **server** (=DAS) as default value for *target* is omitted.

Create and Delete JDBC Connection Pool

Use Case	Operations
Install JDBC Driver	`asadmin add-library `*`jar-file`*
Create JDBC Connection Pool (as Non-XA Data Source)	`asadmin create-jdbc-connection-pool --data-sourceclassname=class-name --restype=javax.sql.DataSource [-property `*`name=value`*`[:`*`name=value`*`]*] `*`pool-name`* [*1] `Use Admin Console : Common Tasks > JDBC > JDBC Connection Pools`

Use Case	Operations
Create JDBC Connection Pool as XA Data Source, *e.g.* for EJB Timer	`asadmin create-jdbc-connection-pool [--data-sourceclassname=class-name] --restype= javax.sql.XADataSource [-property name=value) [:name=value]*] pool-name` *1 Use Admin Console : Common Tasks > JDBC > JDBC Connection Pools
Delete JDBC Connection Pool	`asadmin delete-jdbc-connection-pool [--cascade= {false|true}] pool-name` Use Admin Console : Common Tasks > JDBC > JDBC Connection Pools
Ping JDBC Connection Pool	`asadmin ping-connection-pool pool-name` Use Admin Console : Common Tasks > JDBC > JDBC Connection Pools
View list of JDBC Connection Pools	`asadmin list-jdbc-connection-pools` Use Admin Console : Common Tasks > JDBC > JDBC Connection Pools
Create JDBC Resource for a Connection Pool	`asadmin create-jdbc-resource --connectionpoolid pool-name [--enabled={true|false}] [--target={server|domain|deployment-group-name|instance-name}] jndi-name` Use Admin Console : Common Tasks > JDBC > JDBC Resources
Delete JDBC Resource	`asadmin delete-jdbc-resource [--target={-server|domain|deployment-group-name|in-stance-name}] jndi-name` Use Admin Console : Common Tasks > JDBC > JDBC Resources
View list of JDBC Resources	`asadmin list-jdbc-resources {server| domain| deployment-group-name |instance-name}` Use Admin Console : Common Tasks > JDBC > JDBC Resources

*1 : There are many options and most of them are important for optimal performance.

Enable or Disable Monitoring

Use Case	Operations

Enable Monitoring Service *1	`asadmin set configs.config.${config-name}.monitoring-service.monitoring-enabled=true`
	
	`asadmin set configs.config.${config-name}.monitoring-service.mbean-enabled=true`
	
	`Use Admin Console : Configuration > ${config-name} > Monitoring > General (Tab) > Monitoring Service → Check Enabled`
	
	`Use Admin Console : Configuration > ${config-name} > Monitoring > General (Tab) > Monitoring MBeans → Check Enabled`

Disable Monitoring Service *1	`asadmin set configs.config.${config-name}.monitoring-service.mbean-enabled=false`
	
	`asadmin set configs.config.${config-name}.monitoring-service.monitoring-enabled=false`
	
	`Use Admin Console : Configuration > ${config-name} > Monitoring > General (Tab) > Monitoring Service → Uncheck Enabled`
	
	`Use Admin Console : Configuration > ${config-name} > Monitoring > General (Tab) > Monitoring MBeans → Uncheck Enabled`

Start Monitoring Console	`asadmin set-monitoring-console-configuration --enabled=true`
	
	`Access console at http://<das-host>:8080/monitoring-console`

Change Monitoring Level (for All Components) *1 *2	`Use Admin Console : Configuration > ${config-name} > Monitoring > General (Tab) > Component Level Settings;` 1. `Select All Modules` 2. `Level → "HIGH" (Enabled) or "OFF" (Disabled), and then click "Change Level"` 3. `Click "Save"`

Use Case	Operations

Change Monitoring Level (for Each Components) *1 *2	`asadmin set configs.config.${config-name}. monitoring-service.module-monitoring-levels. ${module-name}=HIGH (Enabled)`
	
	or
	
	`asadmin set configs.config.${config-name}. monitoring-service.module-monitoring-levels. ${module-name}=OFF (Disabled)`
	
	Use Admin Console : Configuration > ${config-name} > Monitoring > General (Tab) > Component Level Settings;
	
	1. Select ${module-name}
	2. ${module-name} > Monitoring Level → "HIGH" (Enabled) or "OFF" (Disabled)
	3. Click "Save"

View Monitoring Settings *1	`asadmin get configs.config.${config-name}. monitoring-service.*` Use Admin Console : Configuration > ${config-name} > Monitoring > General (Tab) > Component Level Settings

*1 : ${config-name} is e.g. "server-config".
*2 : See "Table 1. Components and Services" about ${module-name}.

Server Monitoring

Use Case	Operations
Monitor Server *1	`asadmin get server.${module-name}.*` *2 `Use Admin Console : Common Tasks > server (Admin Server) > Monitor (Tab) > Server (Tab)` `Use JMX or MicroProfile Metrics` *3

*1 : Require to enable monitoring service with one or more monitored components/services.
*2 : See “Table 1. Component and Services” about ${module-name}.
*3 : Require additional settings (environment dependent). See Payara Server Documentation and Payara Blog in detail.

Table 1. Components and Services

asadmin	Admin Console	asadmin	Admin Console
jvm	Jvm	jpa	Java Persistence
transaction-service	Transaction Service	jdbc-connection-pool	Jdbc Connection Pool
connector-service	Connector Service	thread-pool	Thread Pool
jms-service	Jms Service	ejb-container	Ejb Container
security	Security Service	orb	ORB (Object Request Broker)
web-container	Web Container	connector-connection-pool	Connector Connection Pool
jersey	Jersey (Restful Web Services)	deployment	Deployment
web-services-container	Web Services Container	http-service	Http Service

Configure and View Log

Use Case	Operations
View and Configure Server Logger Settings	`asadmin set-log-attributes prop-key=prop-value` Use Admin Console : Configuration > ${config-name} > Logger Settings > General (Tab)
View Server Log	Use Admin Console : Common Tasks > server (Admin Server) > General (Tab) > Click "View Log Files" (Log Viewer) Use Admin Console : Common Tasks > server (Admin Server) > General (Tab) > Click "View Raw Log" (Raw Log Viewer) Open "server.log" file directly. Default Location is as follows: • DAS: ${install-dir}/glassfish/domains/${domain-name}/logs/server.log • Instance: ${install-dir}/glassfish/nodes/${node-name}/${instance-name}/logs/server.log
Rotate Log (Force)	`asadmin rotate-log` Use Admin Console : Common Tasks > server (Admin Server) > General (Tab) > Click "Rotate Log"
Enable Access Logging [*1]	`asadmin set configs.config.${config-name}.http-service.access-logging-enabled=true` Use Admin Console : Configuration > ${config-name} > HTTP Service > Access Logging → Check "Enabled"
Disable Access Logging [*1]	`asadmin set configs.config.${config-name}.http-service.access-logging-enabled=true` Use Admin Console : Configuration > ${config-name} > HTTP Service > Access Logging → Uncheck "Enabled"

Use Case	Operations
View Access Log	Open "access.log" file directly. Default Location is as follows: • DAS: ${install-dir}/glassfish/domains/${domain-name}/logs/access/server_access_log.YYYY-MM-DD.txt • Instance: ${install-dir}/glassfish/nodes/${node-name}/${instance-name}/logs/access/server_access_log.YYYY-MM-DD.txt

*1 : By default, access logging is disabled.

Connect to DAS with Secure Admin

Use Case	Machine	Operations
Setup/Enable Secure Admin	Local	1. asadmin start-domain [--domain-name] 2. asadmin change-admin-password *2 3. asadmin enable-secure-admin 4. asadmin restart-domain Use Admin Console 1. Common Tasks > Domain > Administrator Password (Tab) → Set "New Password" and Click "Save" 2. Common Tasks > server (Admin Server) > Click "Secure Administration..." > Click "Enable Secure Admin" 3. Restart Domain
Access to Admin Console	Local	Open https://localhost:4848/ *1 And enter admin user and password
	Remote	Open https://*das-hostname*:4848/ *1 And enter admin user and password

Use Case	Machine	Operations
Use asadmin command (before Login)	Local	asadmin [--user *admin-user-name*] *command options* *2 And enter admin user and password
	Remote	asadmin --host *das-hostname* [--user *admin-user-name*] *command options* *2 And enter admin user and password
Login	Local	asadmin [--user *admin-user-name*] login *2 And enter admin user and password
	Remote	asadmin --host *das-hostname* [--user *admin-user-name*] login *2 And enter admin user and password
Use asadmin command after Login	Local	asadmin [--user *admin-user-name*] *command options* *2 **Without** enter admin user and password
	Remote	asadmin --host *das-hostname* [--user *admin-user-name*] *command options* *2 **Without** enter admin user and password

*1 : It should be “HTTPS” access if secure admin is enabled.
*2 : By default, admin-user-name is “admin”.

sales@payara.fish **+44 207 754 0481** **www.payara.fish**