Process Manager - Low Level Design Document

1. Detailed Component Design

Class Diagrams

main
+main()
+RestartThroughPatch()
+cleanup(cli, command)
+deploySecuraa(command)
+StartMetaDB(cli, configObject, conf)
+StartShardCluster(isPassive, cli, tag, config, configObject)
+StartShardHandler(cli, conf, configObject)
+InitSecuraaAppsContext()
+SetSecuraaAppsContext(activeIntegrations, dbSession, configObject)
+InitializeMongoDBReplicaSet(isActive, isSecondaryActive, config, configStruct)
+checkMongoServiceReachability(serviceName, config, port)
+passiveServerUp(conf)
+completeSOARUp(conf, cli)
+isActive(conf)
+isPassive(conf)
+isSecondaryActive(conf)
+isServiceVM(conf)
+isDBVM(conf)
+checkAndUpdateActiveServerInfo(mongodbSession, mongoDbHost, conf)
+checkAndUpdatePassiveServerInfo(mongoMetaDBClient, mongoDbHost, conf)
+changeServerType(mongoDefaultClient, currentServerHost, serverType, state, conf)
+updateMetaDbMembers(mongoMetaDBClient, currentServerHost, serverType, priority)
+updateConfigServerMembers(mongoMetaDBClient, currentServerHost, serverType, priority, configObject)
+updateShardMembers(mongoMetaDBClient, currentServerHost, serverType, priority, configObject)
+changePriority(replicaSetDbClient, replicaSetName, currentServerHost, replicaSetPort, priority)

common
+ServiceData
+SystemServices
+CoreServices
+BatchServices
+SecuraaAppServices
+AccessTokenHashMap
+DBSession
+ConfigObject
+ActiveIntegrations
+BuildType
+IsServer()
+CheckIfCaseGroupingEnabled()

docker
+InitDockerServiceContexts()
+checkVolumeExists(volumeName)
+createVolume(volumeName)
+createNFSVolume(volumeName)
+removeVolume(volumeName)
+VerifyVolumeExistance(volumeName)
+CreateServices(cli, dockerTag, PatchTag, patchServiceLists)
+InitSystemServicesContext()
+InitCoreServicesContext()
+InitBatchesContext()
+GetContainerIDByServiceName(cli, serviceName)
+RunCommandInContainer(cli, containerId, commands)
+SetupUtilsPackages(cli)
+CreateUIServices(cli, dockerTag, PatchTag, patchServiceLists)
+CreateServiceWithContext(cli, Ctx)
+CreateZookeeperKafkaService(cli)

models
+IntegrationsResponse
+Response
+Tenant
+TenantDB
+ClientInfo
+SessionStruct
+SystemData
+TenantBackup
+UserIntegrationPortObject
+GetActiveIntegrations(dbSession, configObject, buildType)
+getActiveIntegrations(dbSession, configObject, buildType)
+GetPort(integrationID, executable, configObject, dbSession)
+GetAppParams(paramType, configObject, dbSession, buildName)

Data Structures & Relationships

- `ServiceData` : Represents a service with name, context, port, replicas.

- `IntegrationsResponse` : Integration metadata for tenants.

- `Tenant` , `TenantDB` , `TenantBackup` : Tenant and database details.

- `SessionStruct` : MongoDB session wrapper.

- Global maps: `SystemServices` , `CoreServices` , `BatchServices` , `SecuraaAppServices` (all map[string]ServiceData).

2. Database Design

Entity Relationship Diagram

Syntax error in text
mermaid version 10.9.4