Engine Core

November 2011

Livnat Peer & Omer Frenkel
Agenda

- Overview
- Internals
- Infrastructure
- Road Map
- How To Contribute
Overview
Overview

Engine Clients:

- User Portal
- Web Admin
- REST API

Clients:

- JNDI lookup
- Python SDK

Database (DB):

- LDAP Directory
- PostgreSQL

IPA, AD

Ovirt Node

Fedora

RHEL

Host

VDSM

Host

VDSM

Host

VDSM

XML RPC
Internals

Engine Core

- LDAP Broker
- DB Broker
- VDS Broker

BackendBean

- Commands
- Queries
- Monitoring

LDAP Directory

DB

VDSM Host
Entry Point

- RunAction
- RunMultipleActions
- RunQuery
- RunPublicQuery
- RunAsyncQuery
- GetAsyncQueryResults
- Login
- Logout
Command Lifecycle

- CanDoAction
  - Authorization
  - Backwards compatibility validation
  - Input validation
  - Concrete canDoAction logic

- Execute
  - Handle transactivity
  - Concrete execute command logic
  - Error Handling
  - Logging

- EndAction
  - End successfully
  - End with failure
Commands Hierarchy

- The commands are divided into few main hierarchies
  - VMCommand
  - VMTemplateCommand
  - VDSCommand
  - VdsGroupCommand
  - StorageCommand
  - VMPoolCommand
  - In addition to: tags, bookmarks, permissions, roles, users
Query Lifecycle

- Authorization
- Concrete query logic

Async query
  - Get query id and use polling
Search Mechanism

- Lifecycle
  - Cache mechanism
  - Parsing module
  - Syntax checker
  - SQL statement generation
- Dynamically generated SQL Vs. precompiled PostgreSQL functions
- Code is hard to extend and maintain
  - POC of Lucene on top of Hibernate search
Monitoring

- Polling
  - Host
    - Host statistics
    - VM status and statistics
    - Storage & Network visibility
  - SPM
    - Synchronization of
      - SPM Host
      - Master Storage Domain
      - Master version
      - Storage Domains
VDS Broker

- Responsible for interaction with the hosts
  - Specific commands execution, polling..
  - Notify the backend bean of results and alerts
- XML/RPC – auto generated
- Entry Point – ResourceManagerProxy
  - stateless Bean
- Command pattern – BrokerCommandBase
VDS Broker

- VdsBrokerCommand
  - base for host level commands
- IrsBrokerCommand
  - base for SPM commands
  - SPM Selection
  - Error Handling
    - SPM Failover
Data Access Layer

- JDBC - JdbcTemplate
- Data Source
- Connection pool
- Entry Point – DBFacade
- Calls done through DAOs
- Use PostgreSQL functions
- 100% code coverage in testing
DAL
DAO diagram
Authentication module

- User management is done via LDAP servers
  - Real Time Discovering (using the LDAP SRV record) the LDAP servers of a specific domain
  - Kerberos authentication to LDAP servers
  - Querying for list of users and groups in the LDAP server
- Caching users/groups from LDAP directories
  - Timely syncs with LDAP servers
- Currently Supporting IPA and Active Directory (Auto-detecting the LDAP server type)
- Other LDAP vendors can be easily integrated
- The configuration of new domains is done via an external utility: ovirt-manage-domains
Multi Level Administration

- LDAP-Entity: user/group
- Role:
  - Action groups
  - Predefined Roles (SuperUser, PowerUser, User)
- System, Everyone
- Permission
  - LDAP-Entity + Role + Object
- Each command holds information on the required permission object
Infrastructure
Transaction Management

- JTA - JSR 907, one of the JEE API, provides demarcation of transaction boundaries
- Implemented transaction scopes: RequiresNew, Required, Suppress
- RollBackHandler
- Transaction scope is being set dynamically
- Commands default is Required, CanDoAction not within transaction
- @NonTransactiveCommandAttribute – for irreversible actions, InstallVdsCommand.
- Avoid long transactions – Compensation
JTA Players

RM = Resource Manager*

RM (JDBC Driver)
Resource DB

RM (JMS Driver)
Resource Messaging Services

RM
Resource transactional caches

Application
Application Server

Transaction Manager
Compensation

- Motivation: avoid long transactions
- Transaction is broken into small segments
  - Snapshotting entities initial state during segments
  - Persisting snapshot upon segment commit
- Optimizations
  - Entity snapshot can be limited to status (vs. full entity)
  - Entities snapshot is taken once per segment
- Compensation
  - Command failure
  - Non-graceful shutdown
Session Management

- Login
  - Authentication
  - Attaching a user to a token
  - Thread-Local
- Session invalidation done periodically
  - old and new generation concept

Generation switch:

Old

New

Old
ROAD MAP
Road Map - Today

Engine Clients:

- User Portal
- Web Admin
- REST API
- python SDK
- JNDI lookup
- XML RPC

Engine-core

LDAP Directory

DB

Host

VDSM

Host

VDSM

Host

VDSM
Road Map - Tomorrow

Engine Clients:
- User Portal
- Web Admin
- PythonSDK
- Nomad
- ???

REST API
- LDAP Directory
- DB
- engine-core
- BUS
- Qbg/h Manager
- Host
- Host
- Host
Road Map

- Task Management
- Commands prioritization
- Integrating policy engine (VM scheduling policy, PM policy etc.)
- Unit tests: JPA – multiple DB Vendors, VDSM Mock
- HA
- Scale out
How To Contribute

- Git repository
  
git://gerrit.ovirt.org/ovirt-engine

- Getting started wiki -
  
  • http://www.ovirt.org/wiki/Setting_Management_Development_Environment

- IRC Channel
  
  • #ovirt on oftc
THANK YOU!

http://www.ovirt.org