Engine Core

February 2012

Omer Frenkel -
oVirt engine core maintainer and
Team Lead, Senior Software Engineer, Red Hat

Agenda

- What is the engine core?
- Technologies
- Overview
- Internals
- Road Map
- How To Contribute

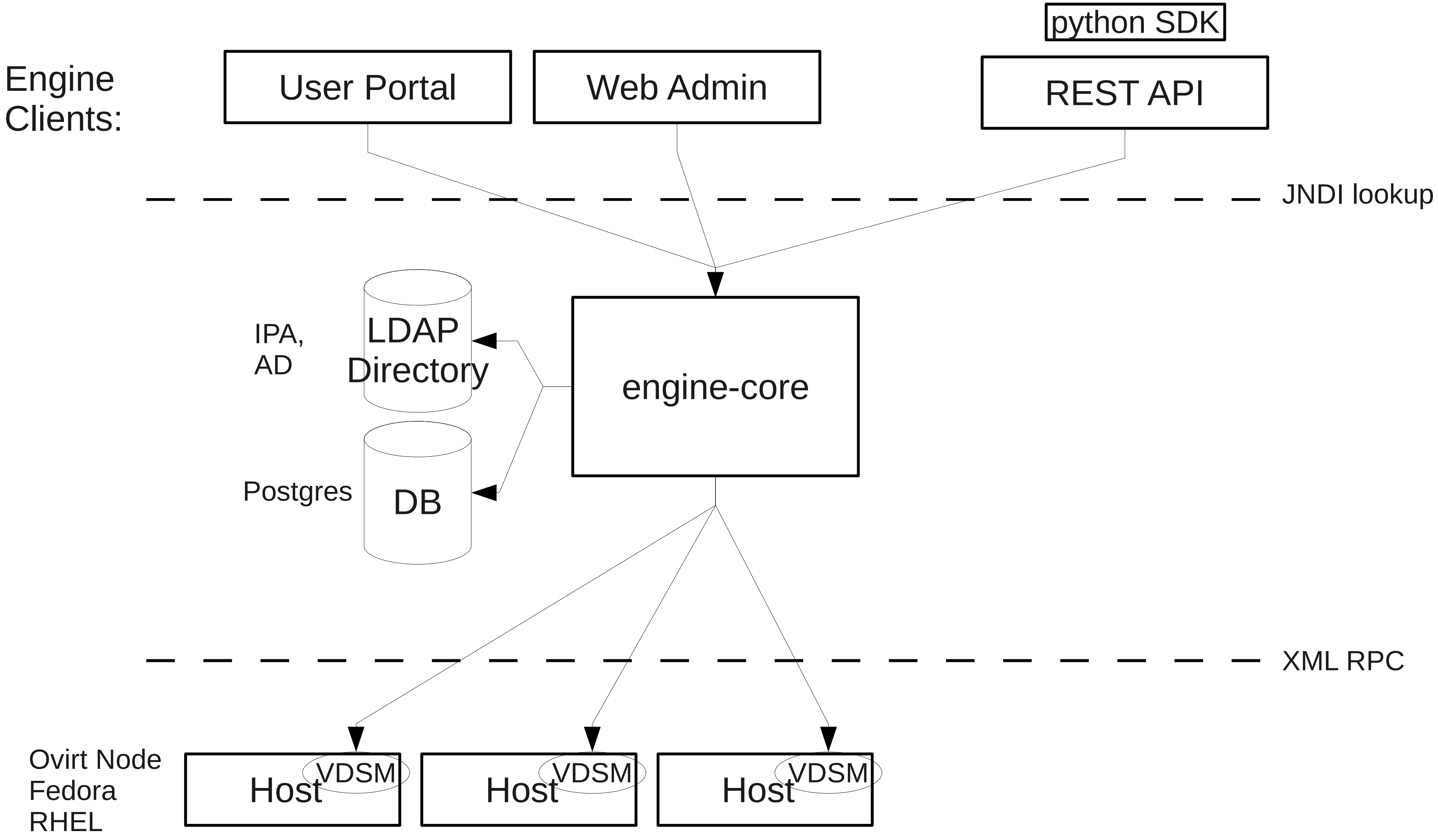
What is the engine core?

- Business logic
- Central component of the oVirt platform
- Process user requests
- Scheduling
- Monitor host agents (vdsm) and vms
- Handle error flows
- Multi-level administration

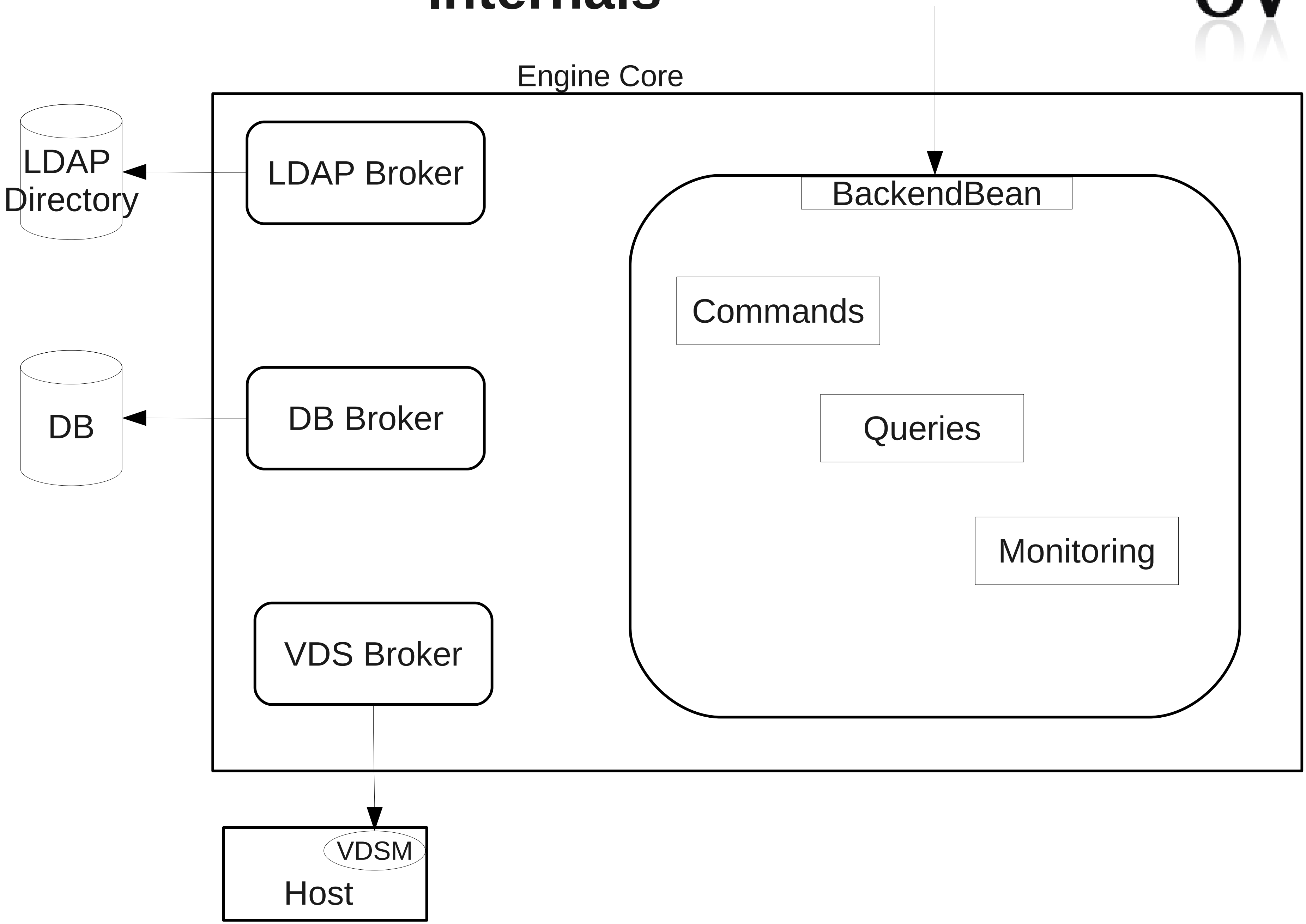
Technologies

- J2EE – runs on JBoss AS 7
- PostgreSQL
- Rest API
- XML-RPC
- Python utilities – mange-domains, configuration..

Overview


python SDK
Engine
Clients:
User Portal
Web Admin
REST API
JNDI lookup
IPA,
AD
LDAP
Directory
engine-core
Postgres
DB
XML RPC
Ovirt Node
Fedora
RHEL
Host
VDSM
Host
VDSM
Host
VDSM

Internals


Engine Core
LDAP Directory
LDAP Broker
BackendBean
Commands
DB
DB Broker
Queries
Monitoring
VDS Broker
VDSM
Host

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

Run Action

Query Lifecycle

- Authorization
- Concrete query logic

Run Query

- 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
  - 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

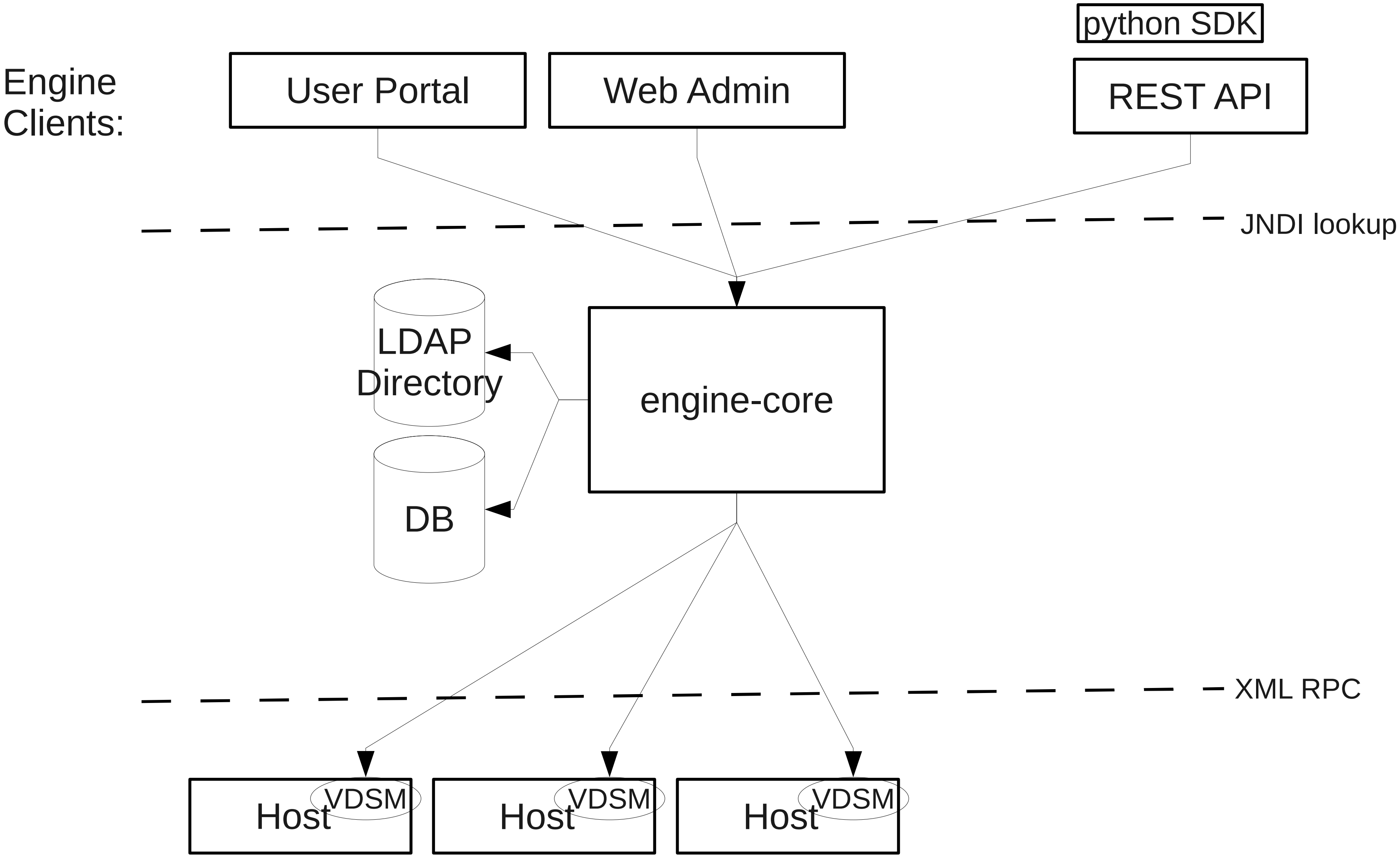
Authentication module

- 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

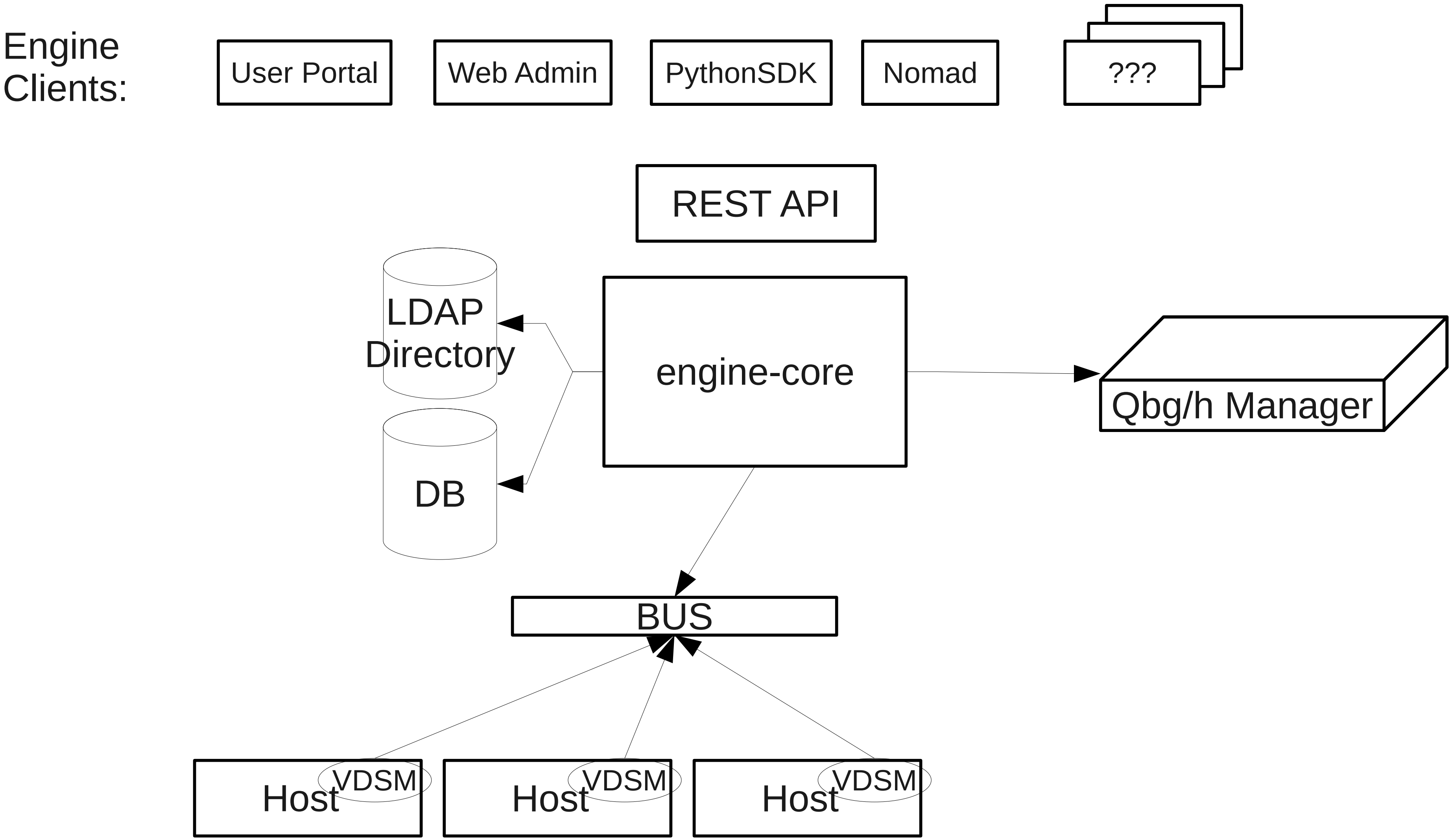
Multi Level Administration

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

Road Map - Today



Road Map - Tomorrow



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/Buildig_oVirt_engine

- IRC Channel
  - #ovirt on oftc

THANK YOU !

http://www.ovirt.org
ofrenkel@redhat.com