

# Quber Device Management Service (QDMS)

Paul

07.Jun.2018

Quber Inc.

# Contents

- Overview
  - Target Market
  - Objectives
  - Features
- System Composition
  - System Overview
  - Server Type
- Co-work with CNU
  - Project Scope

# Overview

- Objectives
- Features
- Target market

# Objectives

- In Technology view,
  - Server Development Infra 구축
  - Cloud-based Server 개발 infra 구축
  - Docker 기반의 Server 개발 경험 획득
- In Business chance view,
  - Device Management Service에 대한 자체 solution 확보
  - 향후 Server 관련 시장 요구사항에 대한 대응방안 확보

# Features

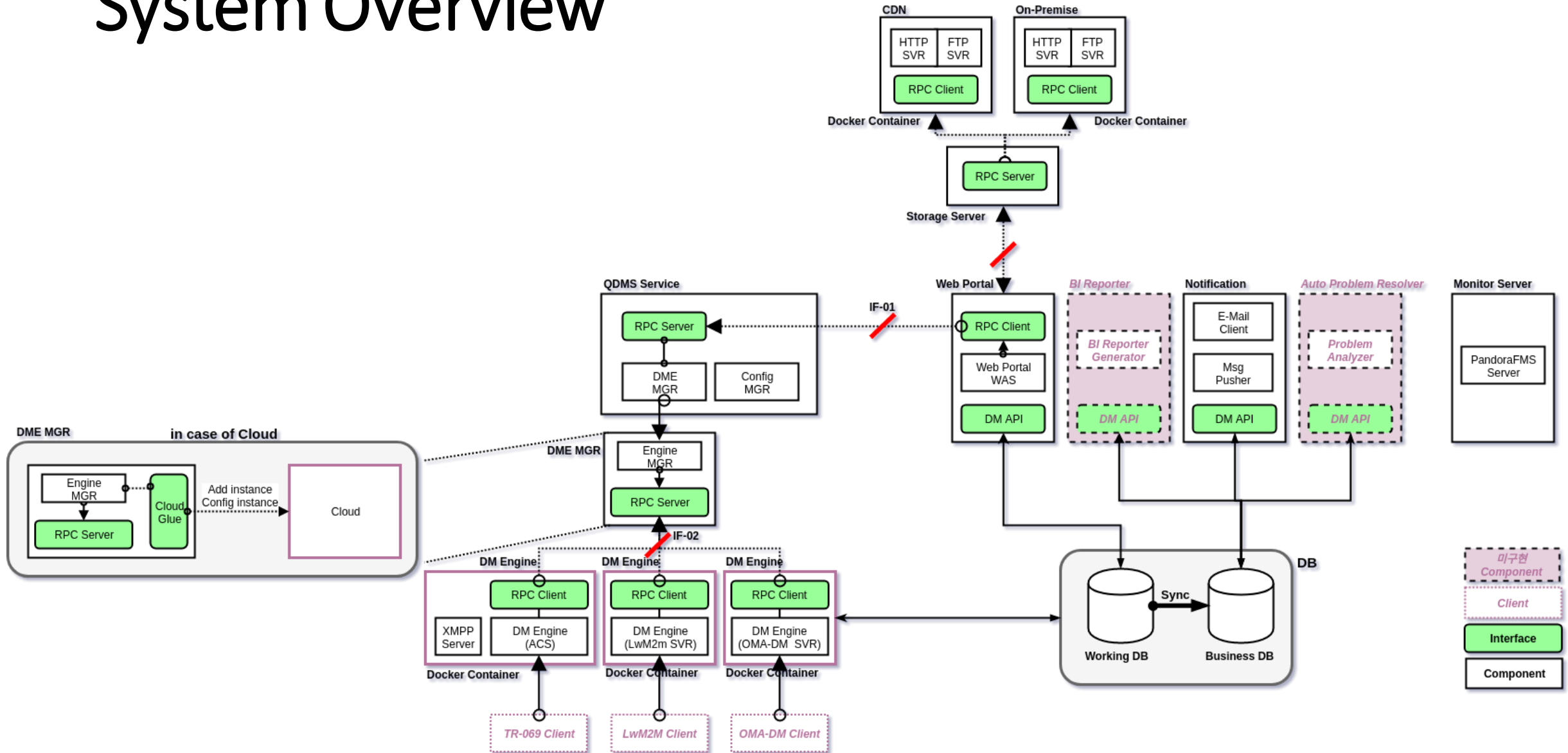
- **Device Management의 용이함**
  - 직관적인 Web Page를 통해 쉬운 Device Management 환경 제공
- **시장의 다양한 Device Management Protocol 수용 가능한 구조**
  - TR-069 / OMA-DM / LwM2M / SNMP / MQTT / etc.
- **서버 증설의 용이함**
  - Docker container 기반의 Device Manamgent Engine 사용
- **사업자에게 다양한 BI정보 전달**
  - 단순 DMS를 넘어 BI 정보원으로 동작
- **High QoE(Quality of Experience) 제공**
  - 가입자의 문제에 대한 선제적인 대응
- **시스템 문제에 대한 다양한 방법의 Alert 방안 제공**
  - 핸드폰 / E-mail / Web log-in화며

# Target Market

- 규모 : 중/소규모
- 용도
  - Device Management를 필요로 하는 사업자
  - 서버관리를 위한 전문인력 유지가 부담스러운 사업자

# System Composition

# System Overview



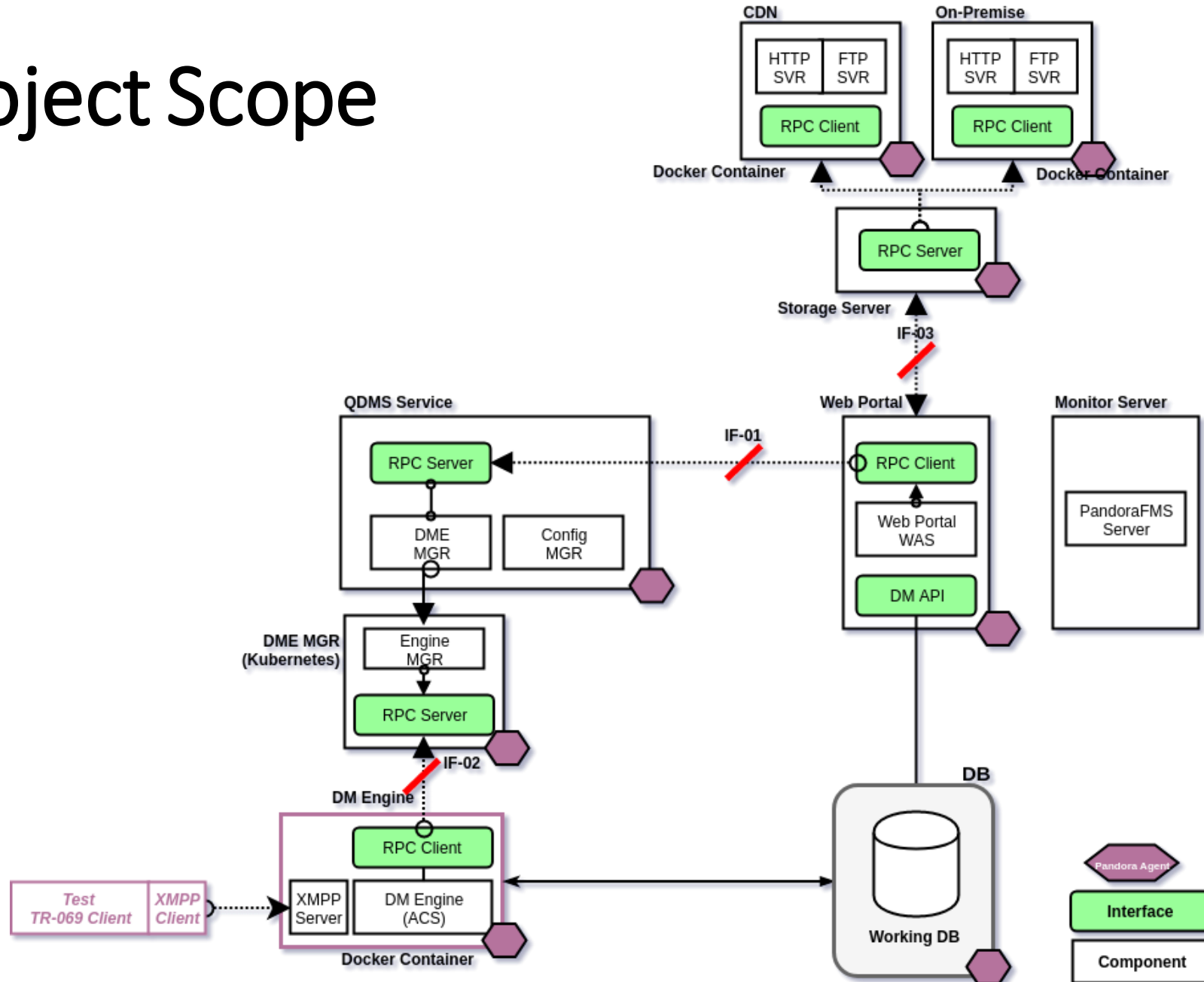


# Server Type

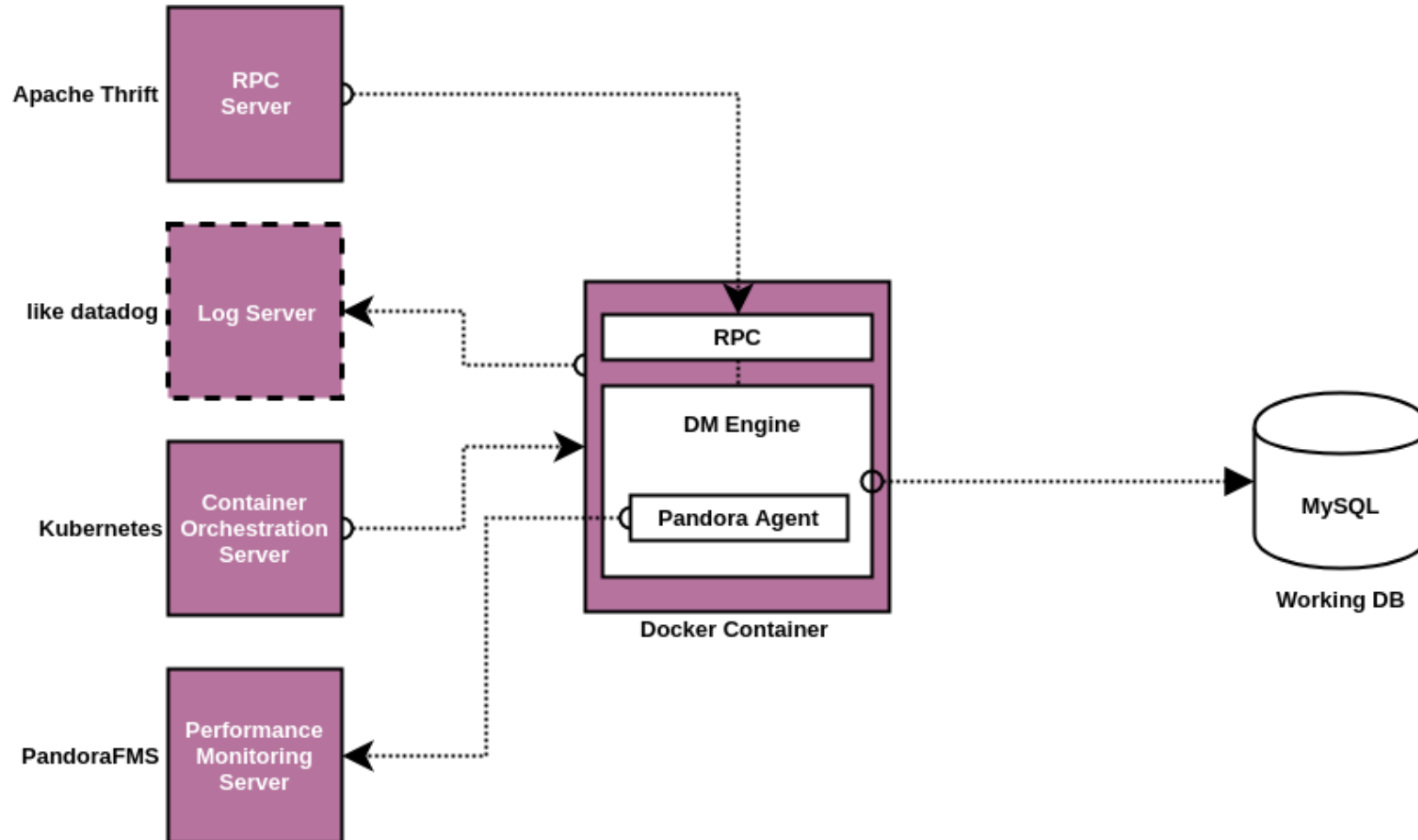
- QDMS main Server
  - QDMS main service
  - DM Manager
- DM Engine Group
  - TR-069(w/ XMPP Server) / OMA-DM / LwM2M / SNMP / etc.
- Web Portal & BI Server
- DB Server
- Storage Server
  - (S)FTP Server / HTTPS Server
- Monitoring Server
  - PandoraFMS client/server

# Project Scope

# Project Scope



# DM Engine Node



# Components

Components	Related Tech. / To do
QDMS Service	<ul style="list-style-type: none"><li>• RPC Server module</li></ul>
DM Engine Manager	<ul style="list-style-type: none"><li>• RPC Server module</li><li>• Container Orchestration Server(Google Kubernetes)</li></ul>
DM Engine (ACS Server)	<ul style="list-style-type: none"><li>• Docker Container</li><li>• RPC Client module</li><li>• Monitoring Agent</li><li>• XMPP Server module</li></ul>
Web Portal	<ul style="list-style-type: none"><li>• RPC Client module</li><li>• Monitoring Agent</li><li>• Spring framework (System Admin/Vendor Admin) / Vue.js</li></ul>
Database	<ul style="list-style-type: none"><li>• <del>DB Synchronization tool</del></li></ul>
Storage Server	<ul style="list-style-type: none"><li>• RPC Server module</li></ul>
On-Premise File Server	<ul style="list-style-type: none"><li>• RPC Client module</li><li>• HTTPS / SFTP Server</li></ul>
Monitoring Server	<ul style="list-style-type: none"><li>• PandoraFMS</li></ul>
Test TR-069 Client	<ul style="list-style-type: none"><li>• Test TR-069 Client</li></ul>

# Interfaces

ID	Procedure Name	Parameters	Param.Desc.	Description
IF-01-01	StartQdmsService			QDMS 전체 서비스를 시작시킨다
IF-01-02	StopQdmsService			QDMS 전체 서비스를 중지시킨다. 중지시 운용중인 모든 DM Engine instance들의 동작이 중지된다
IF-01-03	ReportQdmsService			QDMS서비스의 동작상태를 조회한다(시작시간/운영시간/DM Engine의 개수/각 DM Engine {type/status}/Error여부)
IF-01-04	AddDmEngine	1:type 2:aliasEngine	1:engine type(tr069/LwM2m/SNMP/etc.) 2.engine alias	지정한 DM Engine instance를 추가한다
IF-01-05	RemoveDmEngine	1:type 2:aliasEngine	1:engine type(tr069/LwM2m/SNMP/etc.) 2.engine alias	지정한 DM Engine instance를 삭제한다
IF-01-06	StartDmEngine	1:type 2:DM Engine index	1:DM Engine 종류(TR-069/SNMP/etc.) 2:DM Engine index(0은 all을 의미)	DM Engine을 시작시킨다
IF-01-07	StopDmEngine	1:type 2:DM Engine index	1:DM Engine 종류(TR-069/SNMP/etc.) 2:DM Engine index(0은 all을 의미)	DM Engine을 종료시킨다
IF-01-08	QueryDmEngineSts	1:type 2:DM Engine index	1:DM Engine 종류(TR-069/SNMP/etc.) 2:DM Engine index	DM Engine의 동작상태(Active/Inactive)를 조회한다
IF-02-01	StartEngine			해당 DM Engine을 구동시킨다
IF-02-02	StopEngine			해당 DM Engine을 중지시킨다
IF-02-03	QuerySts			해당 DM Engine의 동작상태를 조회한다
IF-03-01	RegisterFile	1:type 2:provisionGroupId 3:updateGroupId 4:fileName 5:version 6:fileCrc	1: file type (fw / sw / config/etc.) 2: provision group id 3: update group id 4: file name 5: file version 6: file crc	file을 file server에 upload한다
IF-03-02	UnregisterFile	1:fileId	1: fileId	file을 file server에서 삭제한다
IF-03-03	QuerySts			file server들의 동작상태를 조회한다