

1. 引言	3
1.1. 项目背景	3
1.2. 术语	3
1.3. 参考资料	3
2. 总体设计	3
2.1. 设计目标	3
2.2. 运行环境	4
2.3. 总体框架	4
3. 数据结构设计	4
3.1. 表 Project	4
3.2. 表 ServiceInstance	5
3.3. 表 Subscriber	5
3.4. 表 SubscriberGroup	6
3.5. 表 ServicePackage	6
3.6. 表 ServiceTemplate	6
3.7. 表 PolicyGroup	7
3.8. 表 PolicyTemplate	7
3.9. 表 PolicyEntry	7
3.10. 表关系	8
4. 模块设计	8
4.1. 项目管理（Project Management）	8
4.1.1. 功能描述	8
4.1.2. 关键流程	9
4.1.3. 界面设计	9
4.2. 用户管理（Subscriber Management）	10
4.2.1. 功能描述	10
4.2.2. 关键流程	10
4.3. 业务管理（Service Management）	10
4.4. 功能描述	10
4.5. 策略管理（Policy Management）	10

4.5.1. 功能描述.....	10
4.5.2. 界面设计.....	10
4.6. 资源管理（Resource Management）	10
4.6.1. 功能描述.....	10
4.6.2. 界面设计.....	11
4.7. 系统监控（System Monitor）	11
4.7.1. 功能描述.....	11
4.7.2. 界面设计.....	11
4.8. 系统配置（System Setting）	11
4.8.1. 功能描述.....	11
4.8.2. 界面设计.....	11
4.9. OpenDaylight（PolicyManager）	11
4.9.1. 功能描述.....	11
4.9.2. 关键流程.....	11
5. 界面设计.....	11
导航图.....	11
主页.....	12
通用增删改页面.....	12
6. 接口设计.....	12
7. 事件响应设计.....	12
8. 系统出错处理设计.....	12
9. 系统部署.....	12

1. 引言

1.1. 项目背景

{项目背景}

1.2. 术语

{术语}

SCP

SCM

PM

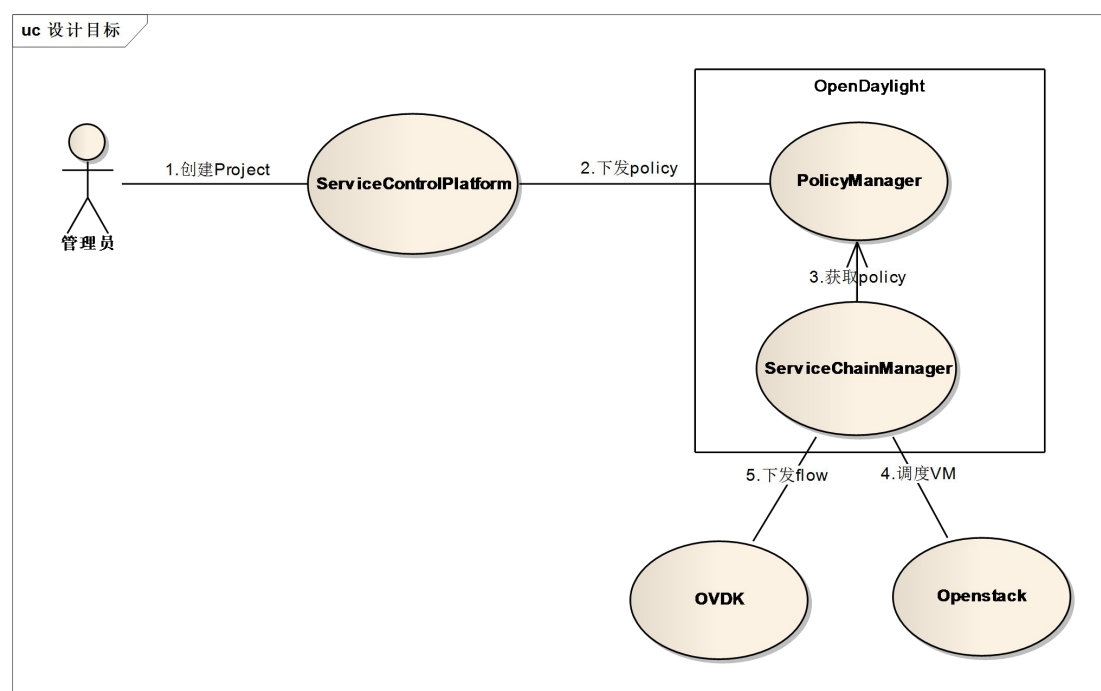
Package

1.3. 参考资料

{参考资料}

2. 总体设计

2.1. 设计目标



1.创建 Project，绑定用户组、资源池、服务套餐

2.SCP 收到 Project 创建后，将其转换为 policy，发送给 PM 保存起来

3.SCM 收到 Pack-in 消息后，获取 policy

4.SCM 调用 Openstack 的 API，进行 VM 调度

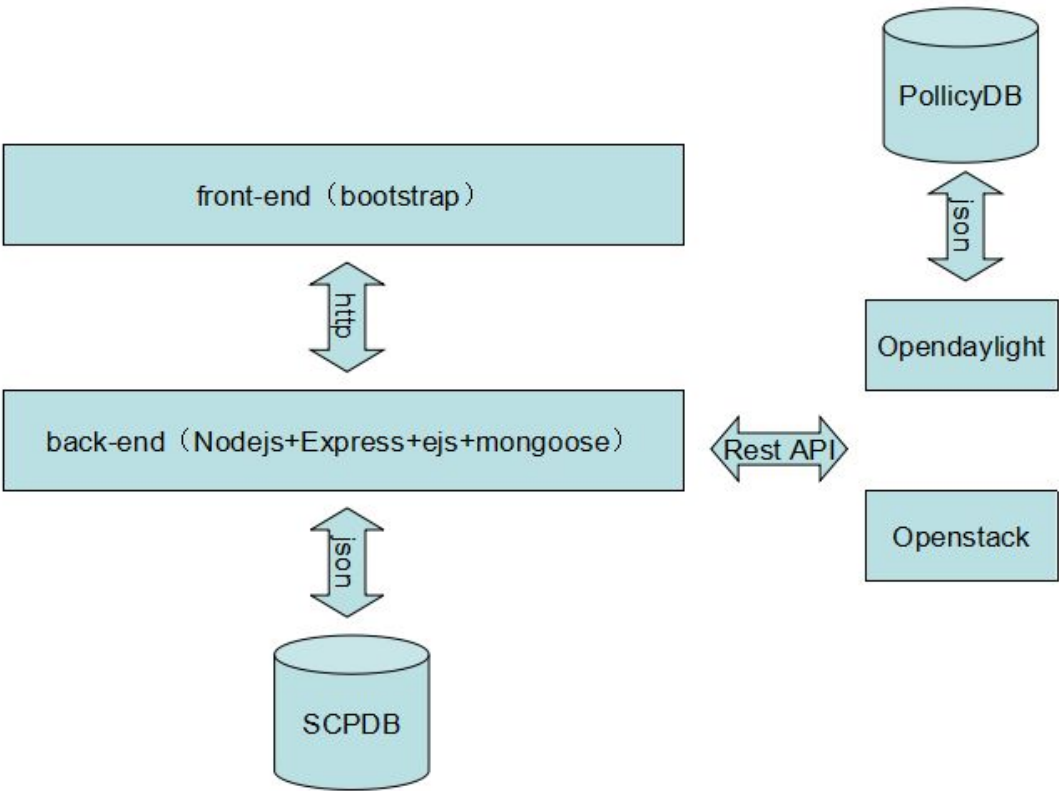
5.根据拓扑发送 flow 到对应的 OVDK

2.2. 运行环境

操作系统：Linux window （推荐 linux）

基础软件：nodejs mongodb

2.3. 总体框架



3. 数据结构设计

3.1. 表 Project

类型	描述	列名
ProjectID	数值类型	每个 Project 记录的 UUID，直接与 Mongodb 创建记录时自动生成的 UUID 关联
ProjectName	字符串类型	每个项目的名称

SubscriberGroupIDServicePackageIDPair	数值类型对	用户组（SubscriberGroupID）与用户组绑定的服务包（ServicePackage）的 ID 键值对,当有多个服务包时存储多个键值对，用于启动服务实例（ServiceInstance）时绑定并生成策略（Policy）
---------------------------------------	-------	--

3.2. 表 ServiceInstance

列名	类型	描述
ServiceInstanceID	数值类型	每个 ServiceInstance 记录的 UUID，直接与 MongoDB 创建记录时自动生成的 UUID 关联
ProjectID	数值类型	关联创建该实例（ServiceInstance）时的项目（Project）UUID
SubscriberGroupID	数值类型	关联创建该实例（ServiceInstance）时的用户组（SubscriberGroup）UUID
AppType	字符串类型	标识当前实例（ServiceInstance）使用的 App
ServicePackageID	数值类型	关联创建该实例（ServiceInstance）时的服务包（ServicePackage）UUID
ServiceTemplateID	数值类型	当前 ServicePackage 下的 ServiceTemplate 的对应 UUID 冗余存储多个 ServiceTemplateID
PolicyGroupID	数值类型	关联当前 ServiceTemplate 下的 PolicyGroupID
PolicyEntryID	数值类型 + 字符串类型	保存这个 PolicyEntryID 以及 PolicyEntryContent 以 json 格式冗余存储包括，流表信息（Flows），五元组（Stuples）

3.3. 表 Subscriber

列名	类型	描述
SubscriberID	数值类型	每个 SubscriberID 记录的 UUID，直接与 MongoDB 创建记录时自动生成的 UUID 关联
AccountName	字符串类型	账户名

IPAddr	字符串类型	账户名对应的 IP 地址
--------	-------	--------------

3.4. 表 SubscriberGroup

列名	类型	描述
SubscriberGroupID	数值类型	每个 SubscriberGroupID 记录的 UUID，直接与 MongoDB 创建记录时自动生成的 UUID 关联
SubscriberGroupName	字符串类型	标识这个 SubscriberGroup 的名称
SubscriberID	数值类型	冗余存储所有被拖入这个 SubscriberGroup 中的 SubscriberID

3.5. 表 ServicePackage

列名	类型	描述
ServicePackageID	数值类型	每个 ServicePackage 记录的 UUID，直接与 MongoDB 创建记录时自动生成的 UUID 关联
ServicePackageName	字符串类型	每个 ServicePackage 记录的名字
ServiceTemplateID	数值类型	关联当前 ServicePackage 下包含的 ServiceTemplate 的 UUID，冗余存储每一条 ServiceTemplateID

3.6. 表 ServiceTemplate

列名	类型	描述
ServiceTemplateID	数值类型	每个 ServiceTemplate 记录的 UUID，直接与 MongoDB 创建记录时自动生成的 UUID 关联
ServiceTemplateName	字符串类型	每个 ServiceTemplate 的名称
ServiceTemplateContent	字符串类型	描述该 ServiceTemplate 特殊的一些属性如下备注中所描述的类型的信息
PolicyGroupID	数值类型	冗余存储当前 ServiceTemplate 下的 PolicyGroupID

3.7. 表 PolicyGroup

列名	类型	描述
PolicyGroupID	数值类型	每个 PolicyGroupID 记录的 UUID，直接与 Mongodb 创建记录时自动生成的 UUID 关联
PolicyGroupName	字符串类型	每个 PolicyGroup 的名字
PolicyTemplateID	数值类型+字符串类型	每个 PolicyTemplateID，和其带有的模版数据，用于在自动生成一些 policyentry 时使用
PolicyentryID	数值类型+字符串类型	每个 PolicyEntry 及其 PolicyEntryContent，用于一些可以重用的基础性策略，已经手动填写好数据

3.8. 表 PolicyTemplate

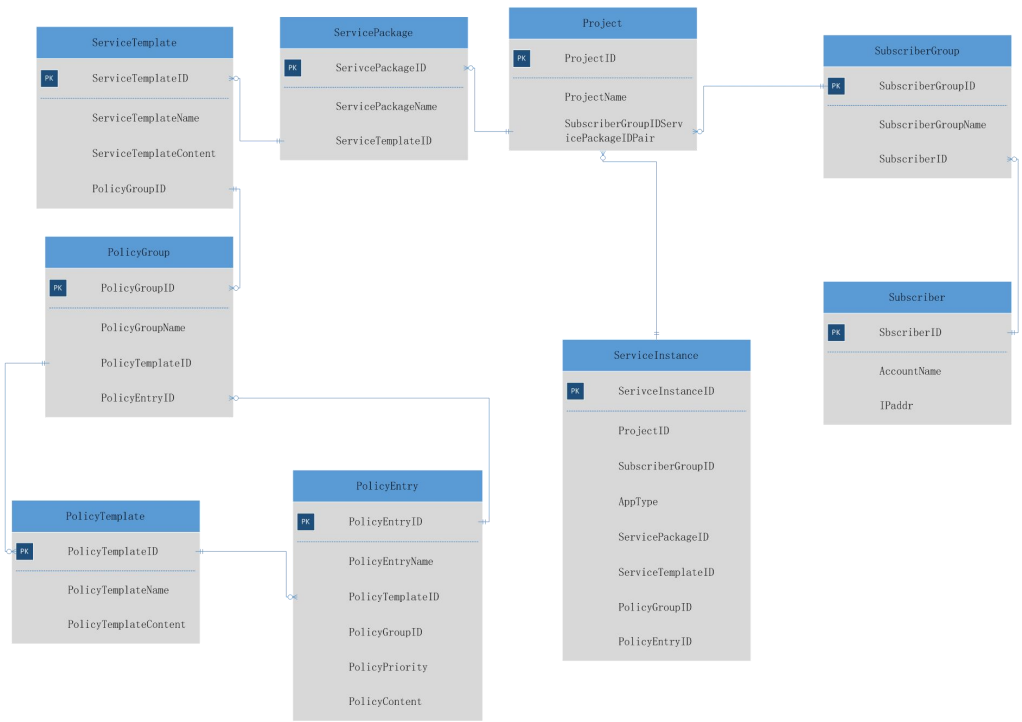
列名	类型	描述
PolicyTemplateID	数值类型	每个 PolicyTemplateID 记录的 UUID，直接与 Mongodb 创建记录时自动生成的 UUID 关联
PolicyTemplateName	字符串类型	每个 PolicyTemplate 的名字
PolicyTemplateContent	字符串类型	每一种 PolicyTemplate 的具体内容详见备注

3.9. 表 PolicyEntry

列名	类型	描述
PolicyEntryID	数值类型	每个 PolicyEntryID 记录的 UUID，直接与 Mongodb 创建记录时自动生成的 UUID 关联
PolicyEntryName	字符串类型	每个 PolicyEntry 的名字
PolicyTemplateID	数值类型	描述该条 PolicyEntry 是由哪条 PolicyTemplate 生成的
PolicyGroupID	数值类型	描述该条 PolicyEntry 属于哪个 PolicyGroup

PolicyPriority	数值类型	标志该条 PolicyEntry 的优先级
PolicyContent	字符串类型	描述具体的策略内容，内容根据所选择的模版的不同而不同

3.10. 表关系



4. 模块设计

4.1. 项目管理（Project Management）

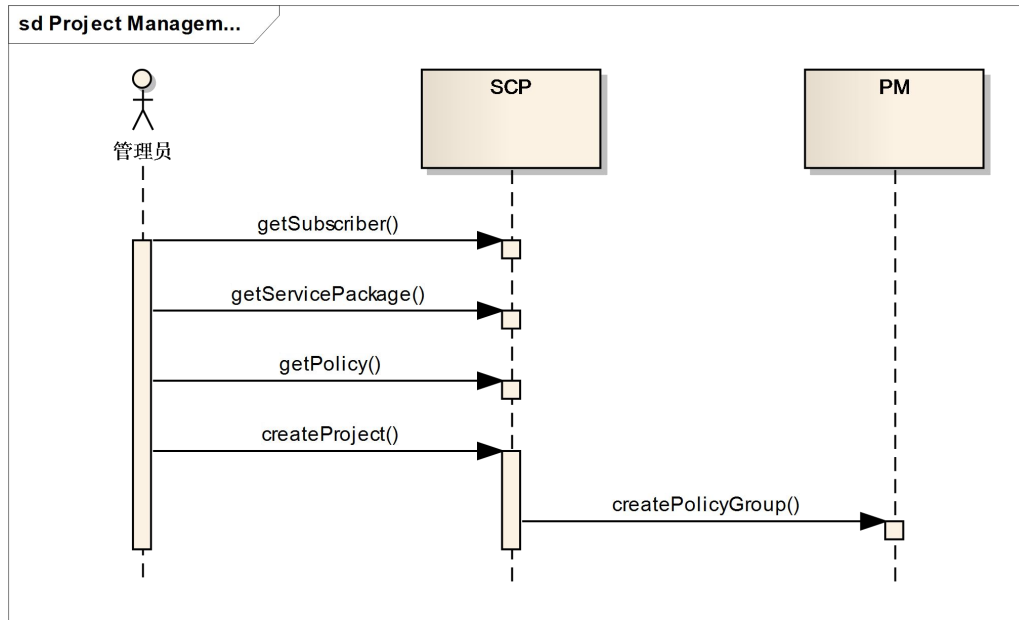
4.1.1. 功能描述

- 创建、维护 Project 信息
- Project 实例化
 - 绑定特定用户组
 - 绑定业务模板
 - 绑定策略
- Service Instance 监测
 - 显示 session 的状态、业务、策略组
 - 外部触发条件：资源、业务、用户的变化

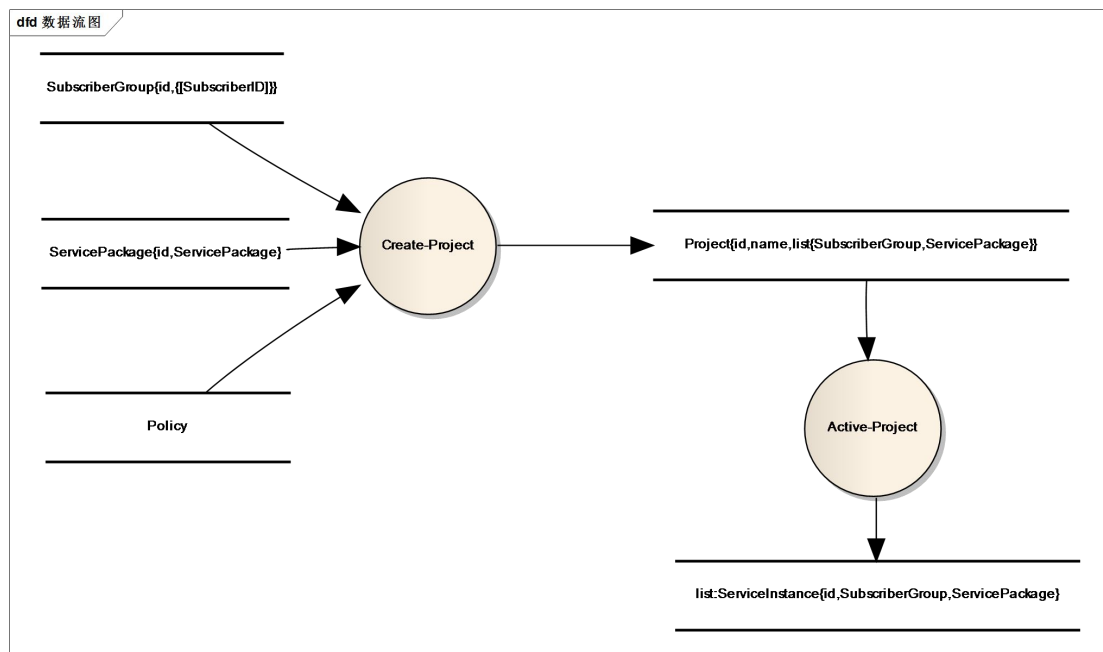
内部触发条件：PM 流表操作结果

4.1.2. 关键流程

创建 Project



数据流图



4.1.3. 界面设计

[ServiceControlPlatform](#)
[Link 1](#) | [Link 2](#) | [Link 3](#)

[One](#)
[Two](#)
[Three](#)

Project Management

CreateDelete

<input checked="" type="checkbox"/> Check box	Item 1	Item 2	#
<input checked="" type="checkbox"/> Check box	Item 3	Item 4	
<input checked="" type="checkbox"/> Check box	Item 5	Item 6	
<input checked="" type="checkbox"/> Check box	Item 1	Item 2	
<input checked="" type="checkbox"/> Check box	Item 3	Item 4	
<input checked="" type="checkbox"/> Check box	Item 5	Item 6	
<input checked="" type="checkbox"/> Check box	Item 1	Item 2	
<input checked="" type="checkbox"/> Check box	Item 3	Item 4	
<input checked="" type="checkbox"/> Check box	Item 5	Item 6	
<input checked="" type="checkbox"/> Check box	Item 5	Item 6	

4.2. 用户管理（Subscriber Management）

4.2.1. 功能描述

用户管理:用户增删改

用户组定义：指定用户的分类，标识用户组的流量，用子网区分

界面设计

4.2.2. 关键流程

4.3. 业务管理（Service Management）

4.4. 功能描述

为不同用户流量准备业务套餐，套餐内包含多项业务约定：包括控制策略、配额、时长限制

方便用户配置，提供各类业务模板，对 Service Chain 类业务，定义 App 类型、处理顺序、业务封装、转发策略模板等配置

界面设计

4.5. 策略管理（Policy Management）

4.5.1. 功能描述

4.5.2. 界面设计

4.6. 资源管理（Resource Management）

4.6.1. 功能描述

4.6.2. 界面设计

4.7. 系统监控（System Monitor）

4.7.1. 功能描述

4.7.2. 界面设计

4.8. 系统配置（System Setting）

4.8.1. 功能描述

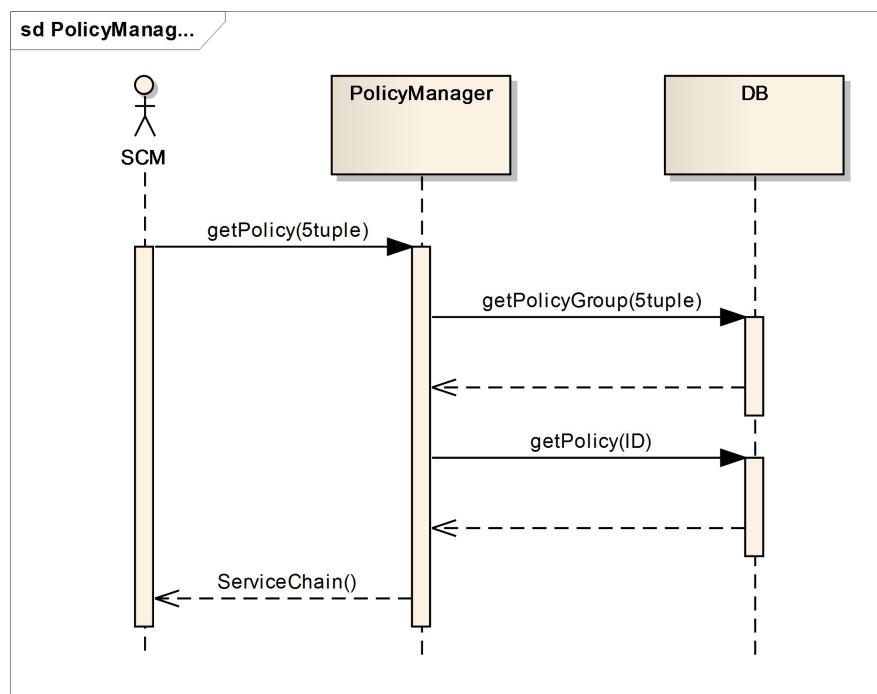
4.8.2. 界面设计

4.9. OpenDaylight（PolicyManager）

4.9.1. 功能描述

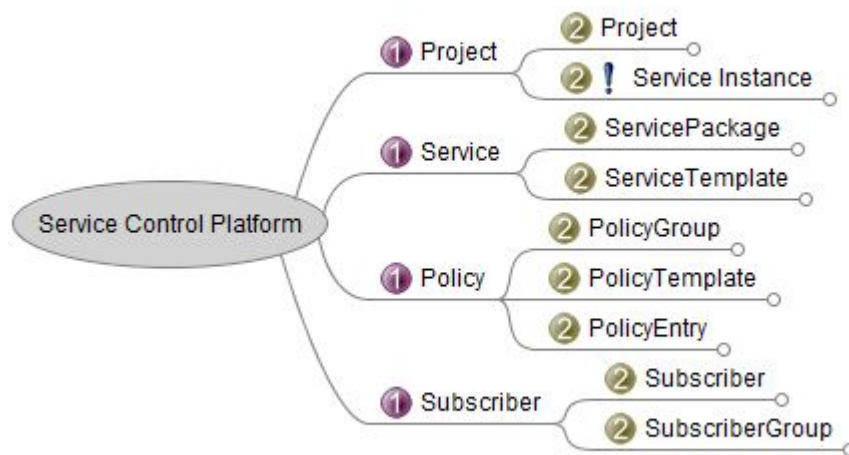
负责维护各类策略规则，以供 SCM 等模块调用

4.9.2. 关键流程



5. 界面设计

导航图



主页

通用增删改页面

6. 接口设计

7. 事件响应设计

8. 系统出错处理设计

9. 系统部署