### 业务子系统交互详细设计

### 功能说明

YUEP统一网管平台中存在如下子系统交互：

1. 客户端和服务端之间的交互
2. 服务端和下级网管的交互
3. 服务端和SBI的交互

其中后两种交互称为业务子系统交互，这两种交互有一些共同点：

* 子系统的访问都需要通过Mo来进行选择，访问SBI通过网元Mo来选择管理该网元的SBI子系统；下级网管通过下级网管的Mo来选择对应的网管服务器，总结起来一个子系统对应于多个Mo。
* 子系统创建、删除过程是相同的，SBI是由用户创建管理的，下级网管也是由用户添加、删除的。
* 子系统的消息订阅过程也是相同

业务子系统交互的不同点：

* 服务端获取服务处理不同，SBI需要根据网元类型和网元版本组装服务名称来获取真正的服务
* 服务端获取下级网管服务需要支持注入Adapter来支持MoNaming的转化

### 设计思路

业务子系统交互框架设计思路如下：

* 提供创建、删除、查询、修改子系统代理的接口
* SBI管理模块根据SBI创建、删除、修改来操作子系统代理服务接口
* NMS管理模块根据Mo创建、删除等消息来操作子系统代理服务接口
* 其他业务模块获取根据Mo通过子系统代理服务接口来获取服务，订阅消息
* 由于SBI和下级网管获取服务方式有所不同，子系统代理模块需要提供服务调用的扩展，

开发者可以根据子系统类型的不同注入不同的服务调用实现

SBIProxy模块设计思路：

由于SBI端业务Adapter模块暴露接口的时候暴露为服务名+网元类型+版本，所以服务端通过SBIProxy获取服务的时候，需要将服务名进行装换。

NMSProxy模块设计思路：

下级网管的管理对象MoNaming和这些管理对象在本级网管的MoNaming是不同的，当本级网管向下级网管进行服务请求的时候，需要在的服务参数中的MoNaming装换为下级网管MoNaming，返回值则需要将下级网管的MoNaming转为本级网管MoNaming。

### 子模块划分



MoCore：Mo核心模块(详细设计见Mo模块详细设计文档);

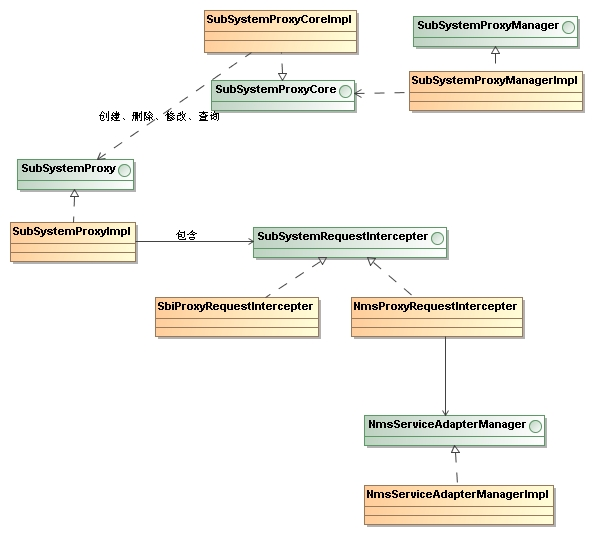
SubSystemProxyCore：子系统代理核心模块，主要提供创建、修改、删除、查询子系统代理服务，其他模块要和对应的子系统交互的时候，需要通过该模块提供的服务获取到相应的子系统代理;

SubSystemProxyManager：提供给其他模块创建、删除、绑定子系统代理使用服务。

SBIProxy:扩展了SBI端Adapter服务的获取;

NMSProxy:支持正对特定的下级网管服务注入Apdater，实现上下级网管MoNaming的转换;

### 类图



|  |  |
| --- | --- |
| 类名称 | 描述 |
| SubSystemProxy | 子系统代理接口 |
| SubSystemProxyImpl | SubSystemProxy接口实现类 |
| SubSystemProxyCore | 子系统代理核心接口，提供创建、删除、修改、查询子系统代理服务。支持根据不同类型的子系统注入SubSystemRequestIntercepter，来实现不同服务请求 |
| SubSystemProxyCoreImpl | SubSystemProxyCore接口实现类 |
| SubSystemProxyManager | 子系统代里管理接口，提供给SBI、NMS模块管理SubSystemProxy。SBI和NMS模块根据SBI或者Mo的创建、删除、修改来添加、删除、修改SubSystemProxy。 |
| SubSystemRequestIntercepter | 子系统服务请求拦截接口，可以拦截对子系统的服务请求，然后实现自己的服务请求 |
| SbiProxyRequestIntercepter | SBI服务请求拦截，实现了根据网元类型+网元版本向SBI请求Adapter服务的功能 |
| NMSProxyRequestIntercepter | NMS服务请求拦截，实现了根据特定服务名称注入Adapter，来实现MoNaming的转换 |

### 消息

SystemProxyMessage 子系统代理模块消息

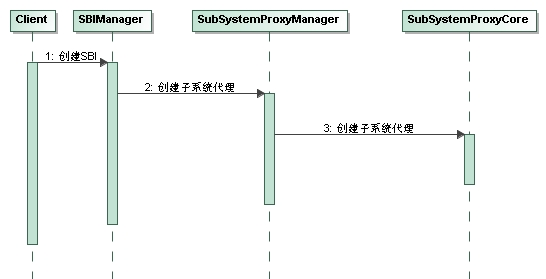
CREATE\_SUBSYSTEMPROXY：创建子系统代理

UPDATE\_SUBSYSTEMPROXY:更新子系统代理

DELETE\_SUBSYSTEMPEOXY：删除子系统代理

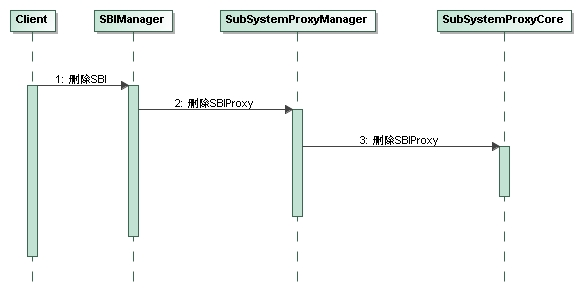
### 典型时序图

* SBI服务请求时序图
* 创建SBIProxy



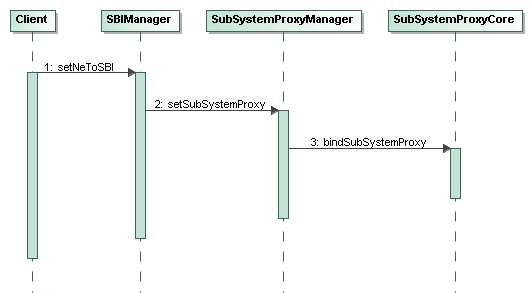
1. 客户端调用SBIManager创建SBI服务
2. SBIManager创建SBI，并且调用SubSystemProxyManager创建SubSystemProxy
3. SubSystemProxyManager调用SubSystemProxyCore创建SubSystemProxy

* 删除SBIProxy



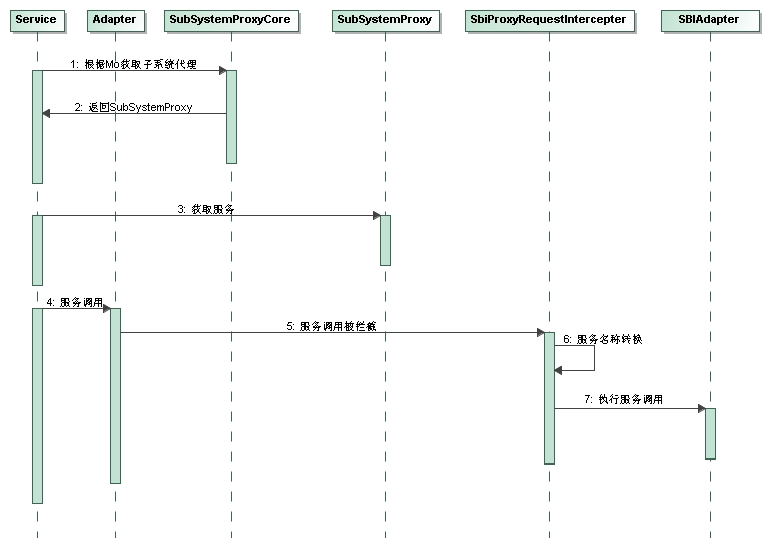
1. 客户端调用SBIManager服务删除SBI
2. SBIManager创建SBI，并且调用SubSystemproxyManager删除SBIProxy
3. SubSystemproxyManager调用SubSystemProxyCore删除SBIProxy

* 网元绑定到SBI



1. 客户端调用SBIManager的 setNeToSBI服务
2. SBIManager调用SubSystemProxyManager的SetSubSystemProxy方法将网元绑定到SBIProxy
3. SubSystemProxyManager调用SubSystemProxyCore的bindSubSystemProxy将网元绑定到SBIProxy

* 获取SBI服务



网管系统中某个服务需要调用SBI端的服务执行过程如下：

根据服务请求的调用SubSystemProxyCore的getSubSystemProxy方法，获取子系统代理

返回SubSystemProxy对象

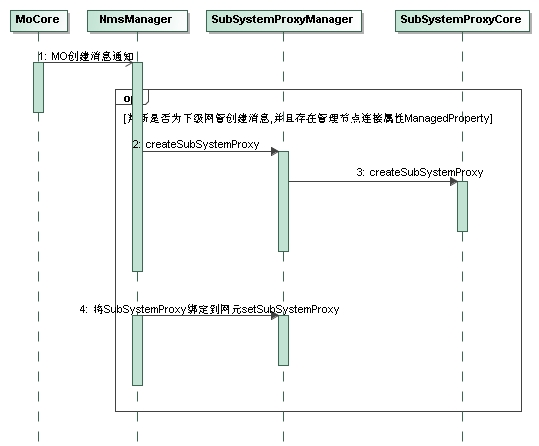
调用SubSystemProxy的getService方法获取SBI端的服务Adapter

获取到SBI端的服务之后，执行方法调用，服务调用被转发到SbiProxyRequestIntercepter对象上

SbiProxyRequestIntercepter根据服务请求的网元Mo获取网元类型、版本，然后进行服务名称装换

SbiProxyRequestIntercepter根据转换的服务名称执行真正的服务调用

* NMS服务请求时序图
* 创建NMSProxy



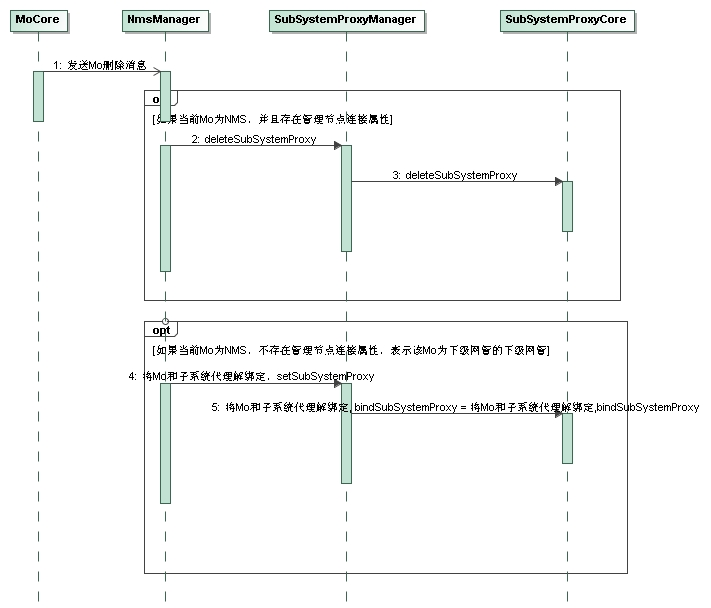
NmsManager监听MoCore发出的Mo创建消息

如果为下级网管NMS创建消息，则调用SubSystemProxyManager创建SubSystemProxy

SubSystemProxyManager调用SubSystemProxyCore创建SubSystemProxy

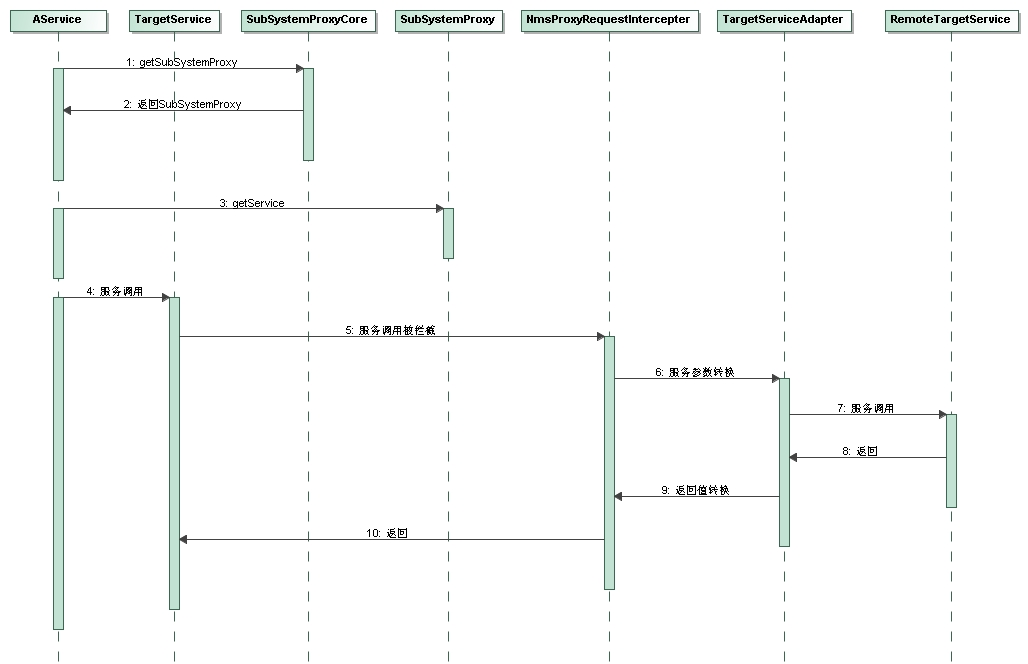
NmsManager调用SubSystemProxyManager的SetSubSystemProxy方法，将当前Mo绑定到SubSystemProxy

* 删除NMSProxy



1. NmsManager监听MoCore发出的Mo删除消息
2. 如果被删除的Mo为NMS对象，并且存在管理节点属性，调用SubSystemProxyManager的deleteSubSystemProxy服务删除SubSystemProxy
3. SubSystemProxyManager调用SubSystemProxyCore的deleteSubSystemProxy方法删除SubSystemProxy
4. 如果不存在管理节点属性，调用SubSystemProxyManager的setSubSystemProxy，将Mo和SubSystemProxy解绑定
5. SubSystemProxyManager调用SubSystemProxyCore的bindSubSystemProxy方法(subSystemId==null)，解绑定

* 下级网管服务调用

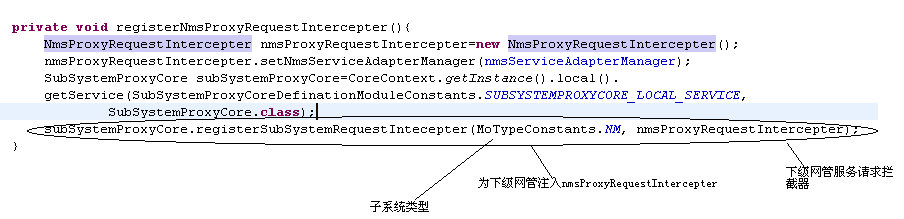


1. AService中调用SubSystemProxyCore的getSubSystemProxy方法获取下级网管SubSystemProxy
2. SubSystemProxyCore返回SubSystemProxy
3. 通过SubSystemProxy获取下级网管服务TargetService
4. 调用TargetService的服务
5. TargetService的服务调用被NmsProxyRequestIntercepter拦截
6. NmsProxyRequestIntercepter调用TargetService的适配器TargetServiceAdapter，TargetServiceAdapter将服务调用参数中的本级网管MoNaming转换为下级网管MoNaming
7. TargetServiceAdapter调用下级网管远程服务RemoteTargetService
8. RemoteTargetService将服务调用结果返回给TargetServiceAdapter
9. TargetServiceAdapter将返回值转换为本级 网管MoNaming
10. NmsProxyRequestIntercepter将服务调用结果返回给调用者

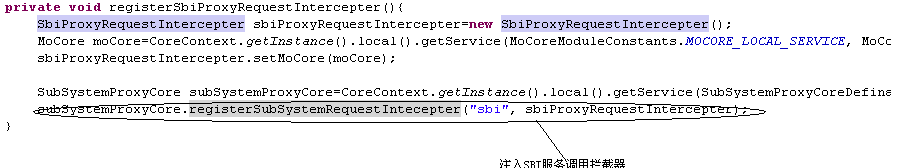
### 业务规则和约束

1. 如何为不同类型的SubSystemProxy注入SubSystemRequestIntercepter

* 注入下级网管服务拦截器

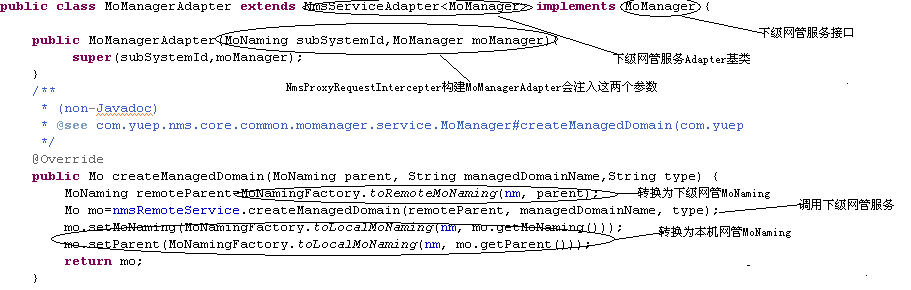


* 注入SBI服务拦截器

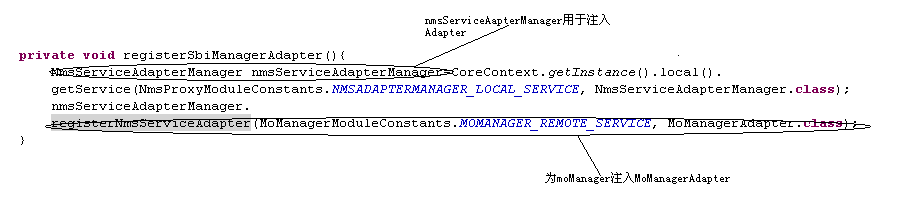


1. NMSProxy下级服务Adapter实现

NmsServiceAdapter类为下级网管服务的Adapter基类，通过下级网管服务Adapter类又必须和原有服务实现同一接口。



1. 如何注入远程服务Adapter



### 第三方库

无！