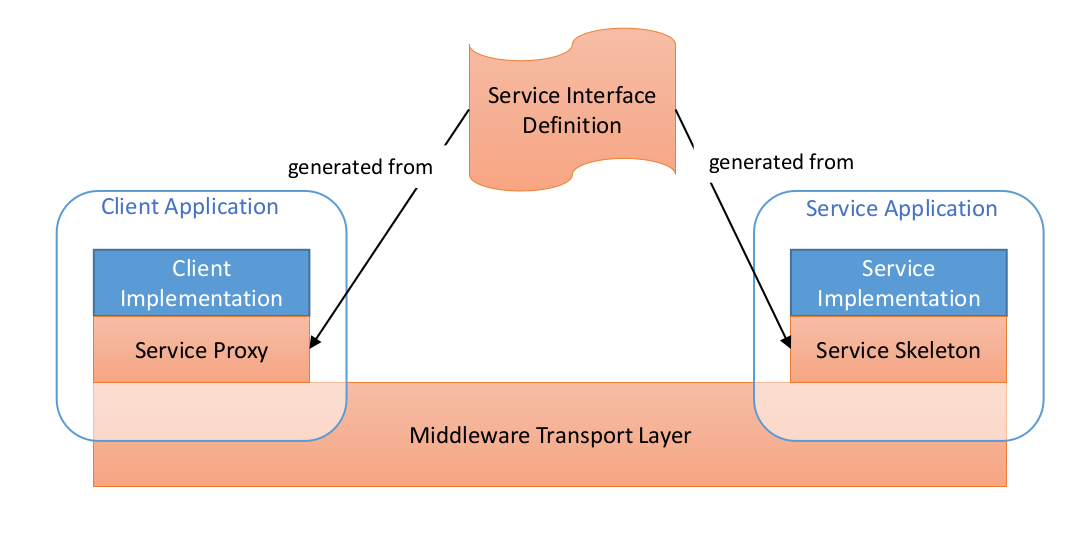
通信管理

# 1 代理/骨架结构

图1.1 Proxy Skeleton Pattern

Middleware Transport Layer: 由一个动态库、相应头文件和一个实现SOME/IP-SD的实例进程组成。

Service Proxy: 这部分代码由代码生成工具根据服务接口的定义生成。

Service Skeleton: 这部分代码由代码生成工具根据服务接口的定义生成。

# 2 API

## 2.1 客户端API

提供给客户端的接口有ServiceProxy类，Event类，Method类，其中Event类，Method类作为ServiceProxy类的成员的方式提供给开发者。

### 2.1.1 ServiceProxy类

#### 2.1.1.1 StartFindService

static ara::com::FindServiceHandle StartFindService(

ara::com::FindServiceHandler<RadarServiceProxy::HandleType> handler,

ara::com::InstanceIdentifier instance = ara::com::InstanceIdentifier::Any);

功能描述：类的静态方法，向通信管理注册的回调函数，当通信管理检测到匹配的实例时，调用回调函数。

#### 2.1.1.2 StopFindService

static void StopFindService(ara::com::FindServiceHandle handle);

功能描述：类的静态方法，让通信管理停止调用由StartFindService注册的回调函数。

#### 2.1.1.3 FindService

static ara::com::ServiceHandleContainer<RadarServiceProxy::HandleType>

FindService(

ara::com::InstanceIdentifier instance = ara::com::InstanceIdentifier::Any);

功能描述：类的静态方法，用于获取服务，创建代理实例前可调用此方法。

### 2.1.1.4 ServiceProxy

ServiceProxy(HandleType &handle)

功能描述：是类的构造方法，用于创建代理实例，参数由FindService的返回值提供

### 2.1.2 Event类

#### 2.1.2.1 Subscribe

void Subscribe(ara::com::EventCacheUpdatePolicy policy, size\_t cacheSize)

功能描述：订阅此Event，指定缓存更新的规则和缓存的大小

#### 2.1.2.2 GetSubscriptionState

ara::com::SubscriptionState GetSubscriptionState()

功能描述：获取订阅状态

#### 2.1.2.3 Unsubscribe

void Unsubscribe();

功能描述：取消订阅

#### 2.1.2.4 SetReceiveHandler

void SetReceiveHandler(ara::com::EventReceiveHandler handler)

功能描述：向通信管理注册回调函数，当收到Event时，通信管理调用此回调函数。

#### 2.1.2.5 UnsetReceiveHandle

void UnsetReceiveHandler();

功能描述：取消回调函数

#### 2.1.2.6 SetSubscriptionStateChangeHandler

void SetSubscriptionStateChangeHandler(

ara::com::SubscriptionStateChangeHandler handler)

功能描述：向通信管理注册回调函数，当订阅状态变更时，通信管理调用此回调函数

#### 2.1.2.7 UnsetSubscriptionStateChangeHandler

void UnsetSubscriptionStateChangeHandler();

功能描述：取消回调函数

#### 2.1.2.8 Update

bool Update(ara::com::FilterFunction<SampleType> filter = {})

功能描述：更新本地缓存的，调用此函数后，GetCachedSamples接口才可以获取新数据

#### 2.1.2.9 GetCachedSamples

ara::com::SampleContainer<ara::com::SamplePtr<const SampleType>> &

GetCachedSamples()

功能描述：获取缓存数据

#### 2.1.2.10 Cleanup

void Cleanup()

功能描述：清空缓存数据

### 2.1.3 Method类

#### 2.1.3.1 operator()

operator()

功能描述：Method类的（）操作符，参数和返回值由具体的服务接口定义，例如ara::com::Future<int32> operator()(int32 p);

### 2.1.4 样例代码

## 2.2 服务端API

提供给服务端的接口有ServiceSkeleton类，Event类，其中Event类作为ServiceProxy类的成员的方式提供给开发者。Method是以ServiceSkeleton类的纯虚成员方法提供，由开发者实现相应的Method.

### 2.2.1 ServiceSkeleton类

#### 2.2.1.1 ServiceSkeleton

ServiceSkeleton( ara::com::InstanceIdentifier instance,

ara::com::MethodCallProcessingMode mode = ara::com::MethodCallProcessingMode::kEvent)

功能描述：类的构造方法，用于创建实例

#### 2.2.1.2 OfferService

void OfferService()

功能描述：提供服务信息

#### 2.2.1.3 StopOfferService

void StopOfferService()

功能描述：停止提供服务信息

#### 2.2.1.4 ProcessNextMethodCall

ara::com::Future<bool> ProcessNextMethodCall()

功能描述：当mode为kPoll时，调用这个接口来获取要处理的方法

### 2.2.2 Event类

#### 2.2.2.1 Send

void Send(const SampleType &data)

功能描述：发送数据，发送后，数据还可访问

#### 2.2.2.2 Allocate

ara::com::SampleAllocateePtr<SampleType> Allocate()

功能描述：分配内存，由Send(ara::com::SampleAllocateePtr<SampleType> data)接口释放。

#### 2.2.2.3 Send

void Send(ara::com::SampleAllocateePtr<SampleType> data)

功能描述：发送数据，发送后，数据不可访问

### 2.2.3 样例代码