

应用与数据集成平台

最佳实践

文档版本 04
发布日期 2024-10-15



版权所有 © 华为云计算技术有限公司 2024。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

1 ROMA Connect 最佳实践汇总	1
2 传统停车场管理系统数字化改造	3
2.1 方案概述	3
2.2 在 ROMA Connect 注册设备	4
2.3 配置设备数据转发 MQS	7
2.4 创建 MQS 到数据库的集成任务	9
2.5 创建并开放数据 API	13
2.6 设备接入 ROMA Connect	16
3 企业数据以 API 形式开放共享	18
3.1 方案概述	18
3.2 通过数据 API 开放数据	19
3.3 通过函数 API 开放数据	26
4 跨系统业务数据集成与转换	31
4.1 方案概述	31
4.2 配置系统间数据集成	32
5 搭建企业业务开放中台	37
5.1 方案概述	37
5.2 配置业务开放中台	39
6 使用 Filebeat+ELK 实现 Site 协同版实例的 API 日志监控	45
6.1 方案概述	45
6.2 资源规划	46
6.3 准备 Docker 环境和镜像	47
6.4 修改 Filebeat 和 ELK 配置	50
6.5 部署 Filebeat 和 ELK	54
6.6 配置索引生命周期策略（可选）	56
6.7 使用 Kibana 查看日志	58
7 使用自定义后端开发自定义认证	61
7.1 方案概述	61
7.2 开发自定义认证	61
7.3 使用自定义认证	65
8 MQS 消息堆积最佳实践	69

9 定时同步 MySQL 中的数据到 Oracle.....	71
9.1 方案概述.....	71
9.2 资源规划.....	71
9.3 创建 MySQL 连接器.....	72
9.4 创建 Oracle 连接器.....	72
9.5 选择模板创建组合应用.....	73
9.6 数据同步验证.....	76

1 ROMA Connect 最佳实践汇总

本文汇总了基于ROMA Connect常见应用场景的操作实践，为每个实践提供详细的方案描述和操作指导，帮助用户轻松构建基于ROMA Connect的应用与数据集成业务。

表 1-1 ROMA Connect 最佳实践一览表

最佳实践	说明
传统停车场管理系统数字化改造	介绍如何通过ROMA Connect对传统停车场的管理系统进行数字化改造，实现停车场车辆进出、计费、停车位状态的智能化管理。
企业数据以API形式开放共享	介绍如何通过ROMA Connect的服务集成，快速把数据库开放为RESTful API，不同业务系统间的数据以API方式互相开放访问，同时在开放API时可选择多种安全认证方式，实现跨网络跨地域应用系统数据间的安全访问。
跨系统业务数据集成与转换	介绍如何通过ROMA Connect的数据集成，把业务系统A中的数据进行格式转换后，集成到业务系统B的数据库中使用。后续根据制定的同步计划，定时把业务系统A中新增的数据同步到业务系统B中，保证了两个系统间的数据同步。
搭建企业业务开放中台	介绍如何通过服务集成的API级联能力，把子公司A和子公司B ROMA Connect实例的API级联到公司总部的ROMA Connect实例上，把公司总部的ROMA Connect实例作为业务开放中台，所有数字资产都通过业务开放中台对外共享。
使用Filebeat+ELK实现Site协同版实例的API日志监控	通过在ROMA Site协同版本地部署Filebeat+ELK，直接获取ROMA Site协同版实例中的API调用日志，并通过ELK的Kibana可视化界面查看API的调用日志以及实现API调用日志的汇聚统计。

最佳实践	说明
使用自定义后端开发自定义认证	使用ROMA Connect的服务集成APIC，可以快速将已有后端服务、数据源和自定义函数封装成标准的RESTful API并对外开放，简化了业务系统间的对接过程，降低了对接成本。
MQS消息堆积最佳实践	消息堆积问题可以从消费者端、生产者端和服务端三个方面进行处理。
定时同步MySQL中的数据到Oracle	通过模板“定时同步MySQL中的数据到Oracle”创建组合应用，系统会自动抓取MySQL数据库中的最新数据记录，并将其无缝对接到Oracle数据库中，实现两个数据库之间的高效、准确的数据更新与同步。

2 传统停车场管理系统数字化改造

方案概述

在ROMA Connect注册设备

配置设备数据转发MQS

创建MQS到数据库的集成任务

创建并开放数据API

设备接入ROMA Connect

2.1 方案概述

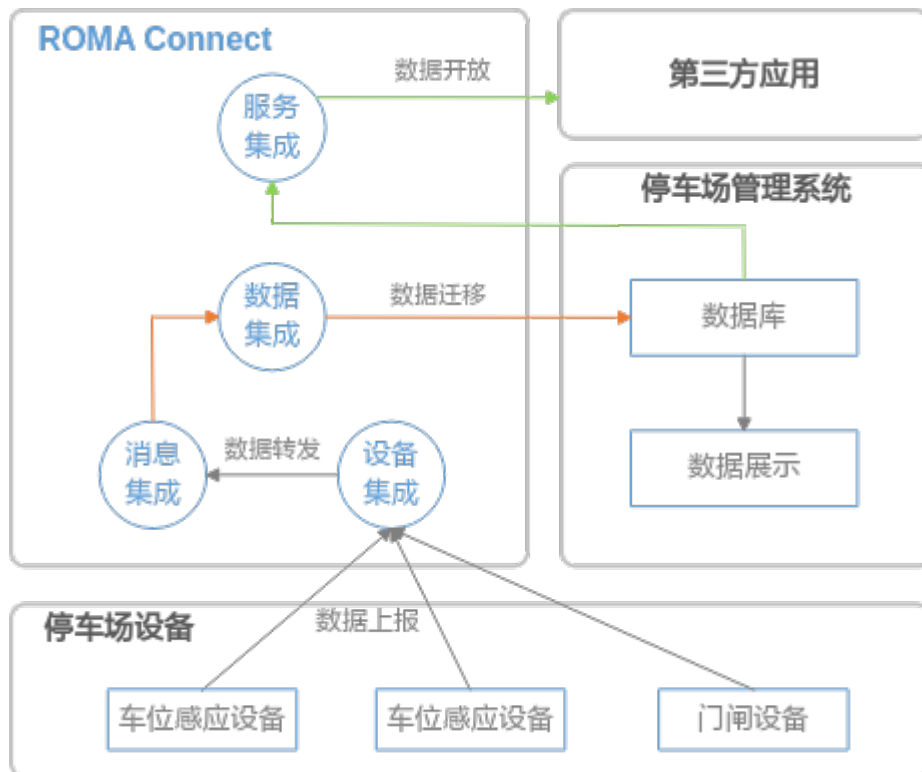
应用现状

随着城市汽车数量的增加，停车难的问题日益突出。

- 传统停车场靠人力管理，难以实时掌握停车场的车位全景，进出、缴费时间长，车位流转效率低下。
- 停车用户无法实时了解停车场的空余车位情况，在找车位过程中需要耗费大量时间，进一步增加道路交通压力。
- 交通部门无法实时掌握城市停车位使用情况，难以针对停车问题制定有效的交通缓解措施。

解决方案

本章节介绍如何通过ROMA Connect对传统停车场的管理系统进行数字化改造，实现停车场车辆进出、计费、停车位状态的智能化管理。



1. 停车场车位感应设备和进出门闸设备通过设备集成连接到ROMA Connect，实时上报车位状态信息和车辆出入信息。
2. 设备上报的数据经规则引擎转发到消息集成的Topic上存储。
3. 通过数据集成任务把Topic中的停车场数据实时写入停车场管理系统的数据库中，用于车位状态监测和停车计费，并可在IOC大屏上显示。
4. 通过数据API把数据库中的车位状态数据开放给第三方应用，供车主、城市管理者了解停车场车位情况。

使用ROMA Connect实现停车场管理系统数字化改造的业务流程如下：

1. [在ROMA Connect注册设备](#)
2. [配置设备数据转发MQS](#)
3. [创建MQS到数据库的集成任务](#)
4. [创建并开放数据API](#)
5. [设备接入ROMA Connect](#)

2.2 在 ROMA Connect 注册设备

概述

通过在ROMA Connect注册设备，为设备在ROMA Connect内分配唯一的ID和密码，使设备可以接入到ROMA Connect。

前提条件

- ROMA Connect与停车场设备和停车场管理系统的网络互通。若通过公网互通，ROMA Connect实例需绑定弹性IP。

- 停车场设备包含车位感应设备和进出门闸设备，配置过程类似，此处以车位感应设备的配置为例进行说明。

操作步骤

1. 创建集成应用。
 - a. 登录ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
 - b. 在左侧的导航栏选择“集成应用”，单击页面右上角的“创建集成应用”。
 - c. 在创建集成应用弹窗中填写集成应用的“名称”，然后单击“确认”。
2. 根据停车场设备的能力模型，在ROMA Connect上定义产品。
 - a. 在左侧的导航栏选择“设备集成 LINK > 产品管理”，单击页面右上角的“创建产品”。
 - b. 在创建产品弹窗中配置产品相关信息，完成后单击“确认”。

表 2-1 产品配置

参数	配置说明
产品类型	停车场设备通过网络直连ROMA Connect，此处选择“普通产品”。
协议类型	停车场设备通过集成MQTT客户端接入ROMA Connect，此处选择MQTT协议类型。
应用	选择1中创建的集成应用。
产品模板	本实践配置中未使用产品模板，此处不设置。
产品名称	填写产品的名称，自定义。
厂商名称	填写停车场设备的厂商名字。
厂商ID	填写停车场设备的厂商ID。
产品型号	填写停车场设备的产品型号。
设备类型	选择接入设备的类型，本实践配置中选择默认的“Default Type”。
模型版本	填写设备的模型版本，本实践配置中不设置。
描述	填写产品的描述信息。

- c. 在“产品管理”页面的“产品”页签下，单击2.b中创建的产品名称，进入产品详情页面。
- d. 在“物模型”页签下，单击“创建物模型服务”。
- e. 在创建物模型服务弹窗中配置物模型服务相关信息，完成后单击“确认”。

表 2-2 物模型服务配置

参数	配置说明
物模型服务名称	填写物模型服务的名称，自定义，例如“ParkingStatus”。
是否启用	是否启用该物模型服务，保持默认“启用”。
描述	填写物模型服务的描述信息。

- f. 选择2.e中创建的物模型服务，在右侧物模型服务的“属性”页签下单击“创建属性”。
- g. 在新增属性弹窗中配置属性相关信息，完成后单击“确认”。

此处以车位感应设备的配置为例进行说明，车位感应设备上报如下信息：

- 设备物理标识“deviceId”，数据类型为“String”。
- 车位状态属性数据“status”，数据类型为“String”，值为“0”表示未停车，值为“1”表示已停车。

表 2-3 服务属性配置

参数	deviceId属性配置说明	status属性配置说明
属性名称	设备上报属性参数的名称，此处填写为“deviceId”。	设备上报属性参数的名称，此处填写为“status”。
数据类型	设备上报属性参数的数据类型，此处选择“String”。	设备上报属性参数的数据类型，此处选择“String”。
是否必填	设备是否必须上报该属性，保持默认必填。	设备是否必须上报该属性，保持默认必填。
描述	填写属性的描述信息。	填写属性的描述信息。
最小值	填写物模型服务属性的最小值。	填写物模型服务属性的最小值。
最大值	填写物模型服务属性的最大值。	填写物模型服务属性的最大值。
步长	填写属性的步长。	填写属性的步长。
单位	填写属性的表达含义使用的单位。	填写属性的表达含义使用的单位。

3. 在ROMA Connect上注册设备。
每个停车场设备都需要在ROMA Connect上进行注册。
 - a. 在左侧的导航栏选择“设备集成 LINK > 设备管理”，单击页面右上角的“创建设备”。

- b. 在创建设备弹窗中填写设备相关配置信息，完成后单击“确认”。

表 2-4 设备配置

参数	配置说明
设备名称	填写设备的名称，自定义。
集成应用	选择1中创建的集成应用。
所属产品	选择2中所创建的产品。
设备标识	填写设备的物理标识，如设备的IMEI、MAC地址等。
设备密码	填写设备的接入密码，若不设置，则系统自动生成。
确认密码	填写确认密码，若不设置，则系统自动生成。
是否启用	选择是否启用设备，只有启用设备后，设备才可以接入ROMA Connect，保持默认启用。
设备标签	为设备设置标签信息，用于快速查找设备。
设备描述	填写设备的描述信息。

2.3 配置设备数据转发 MQS

概述

ROMA Connect不会直接存储设备上报的数据，用户需要通过配置数据转发规则，把设备数据转发到其他服务进行存储。本实践中，把设备数据转发到ROMA Connect的消息集成MQS。

操作步骤

1. 创建用于存储设备数据的消息Topic。
 - a. 在左侧的导航栏选择“消息集成 MQS > Topic管理”，单击页面右上角的“创建Topic”。
 - b. 在创建Topic弹窗中填写Topic相关配置信息，然后单击“确定”，创建Topic。

表 2-5 Topic 配置

参数	配置说明
Topic名称	填写Topic的名称，自定义。
集成应用	选择在ROMA Connect注册设备中创建的集成应用。
权限	为Topic所属的集成应用选择对Topic的操作权限，此处选择“发布+订阅”，即该Topic可用于生产和消费消息。

参数	配置说明
分区数	合理设置分区数量，可以提升消息生产与消费的并发性能。
副本数	ROMA Connect会自动在每个副本上备份数据，当其中一个副本故障时数据依然可用，Topic的副本数越多，可靠性越高。
老化时间（小时）	超过老化时间后，Topic中存储的消息将会被删除。
同步复制	客户端向Topic生产消息时，是否把消息复制给所有副本，然后才向消息客户端返回响应，此处不启用。
同步落盘	消息客户端向Topic生产的每条消息是否立即写入磁盘，此处不启用。
标签	添加Topic的标签信息，用于快速过滤和查找Topic。
敏感字段	填写Topic的消息敏感字段。若向Topic生成的消息中包含敏感字段，则包含敏感字段的整条消息内容会被屏蔽。此处不添加。
描述	填写Topic的描述信息。

2. 配置数据转发规则。

- 在左侧的导航栏选择“设备集成 LINK > 规则引擎”，单击页面右上角的“创建规则”。
- 在创建规则弹窗中填写规则相关配置信息，完成后单击“确认”。

表 2-6 规则配置

参数	配置说明
应用名称	选择在ROMA Connect注册设备中创建的集成应用。
规则名称	填写规则的名称，自定义。
规则描述	填写规则的描述信息。
状态	是否启用该规则，保持默认开启。

- 在规则列表中单击2.b中创建的规则名称，进入规则详情页面。
- 在“数据源端”下单击“创建数据源端”，设置数据源端相关配置，然后单击“保存”。

表 2-7 数据源端配置

参数	配置说明
产品名称	选择在ROMA Connect注册设备中所创建的产品。

参数	配置说明
设备名称	选择“全部设备”。
Topic名称	选择设备发送消息的Topic，格式为：{产品标识}/out/+。
Topic级别	Topic的级别，根据“设备名称”的选择自动适配，“设备名称”不做选择，默认为“产品级别”；“设备名称”选择具体设备则默认匹配为“设备级别”。
base64编码	是否对转发的设备数据进行base64编码，此处不开启。
包含设备信息	转发的设备数据是否包含设备信息，为方便后续识别转发数据所属的设备，此处开启。

- e. 在“数据目的端”下单击“创建数据目的端”，设置数据目的端相关配置，然后单击“保存”，完成数据转发规则的配置。

表 2-8 数据目的端配置

参数	配置说明
数据目的端	本实践中把数据转发到ROMA Connect的消息集成MQS，此处选择“ROMA MQS”。
连接地址	选择当前ROMA Connect实例的“消息集成 MQS内网连接地址”，您可以在ROMA Connect实例控制台的“实例信息”页面中查看得到。
Topic名称	选择1中创建的消息Topic。
用户名	仅当ROMA Connect实例的“MQS SASL_SSL”已开启时需要配置。 填写“Topic名称”中Topic所属集成应用的Key。
密码	仅当ROMA Connect实例的“MQS SASL_SSL”已开启时需要配置。 填写“Topic名称”中Topic所属集成应用的Secret。

2.4 创建 MQS 到数据库的集成任务

概述

通过数据集成任务，把MQS中的设备数据进行数据结构转换并写入停车场管理系统的数据库中。此处假设停车场管理系统的数据库为MySQL。

操作步骤

1. 接入MQS数据源。
 - a. 在左侧的导航栏选择“数据源管理”，单击页面右上角的“接入数据源”。

- b. 在接入数据源页面的“默认数据源”页签下，选择“MQS”数据源类型，然后单击“下一步”。
- c. 在页面中填写MQS数据源的相关配置信息。

表 2-9 MQS 数据源配置

参数	配置说明
数据源名称	填写MQS数据源的名称，自定义。
集成应用	选择在ROMA Connect注册设备中所创建的集成应用。
描述	填写数据源的描述信息。
连接地址	选择当前ROMA Connect实例的“消息集成 MQS内网连接地址”，您可以在ROMA Connect控制台的实例信息页面查看该地址。
是否SSL	仅当ROMA Connect实例的“MQS SASL_SSL”已开启且“VPC内网明文访问”未开启时，选择“是”，其他情况下选择“否”。
SSL用户名/应用Key	仅当“是否SSL”选择“是”时需要配置。 填写配置设备数据转发MQS中创建的Topic所属集成应用的Key。
SSL密码/应用Secret	仅当“是否SSL”选择“是”时需要配置。 填写配置设备数据转发MQS中创建的Topic所属集成应用的Secret。

- d. 完成MQS数据源参数配置后，单击“开始检测”，进行数据源的连接测试
 - 若测试结果为“数据源连接成功！”，则继续下一步。
 - 若测试结果为“数据源连接失败！”，则检查和修改数据源连接参数配置，然后单击“重新检测”，直到连接成功为止。
 - e. 单击“创建”，完成MQS数据源的接入。
2. 接入MySQL数据源。
- a. 在“数据源管理”页面单击右上角的“接入数据源”。
 - b. 在接入数据源页面的“默认数据源”页签下，选择“MySQL”数据源类型，然后单击“下一步”。
 - c. 在页面中填写MySQL数据源的相关配置信息。

表 2-10 数据源配置参数

参数	配置说明
数据源名称	填写数据源的名称，自定义。
集成应用	选择在ROMA Connect注册设备中所创建的集成应用。
描述	填写数据源的描述信息。

参数	配置说明
连接模式	选择“默认”方式。
连接地址	填写MySQL数据库的访问IP地址。
端口	填写MySQL数据库的访问端口，MySQL的默认访问端口为3306，以数据库的实际配置为准。
数据库名	填写数据表所在的数据库名称。
编码格式	数据库所使用的编码格式。
超时时间（秒）	连接数据库的超时时间，单位为秒。
用户名	访问数据库的用户名，且该用户具备数据库的读写权限。
密码	访问数据库的用户密码。

- d. 完成数据源参数配置后，单击“开始检测”，进行数据源的连接测试
 - 若测试结果为“数据源连接成功！”，则继续下一步。
 - 若测试结果为“数据源连接失败！”，则检查和修改数据源连接参数配置，然后单击“重新检测”，直到连接成功为止。
 - e. 单击“创建”，完成MySQL数据源的接入。
3. 创建MQS到MySQL的数据集成任务。
- a. 在左侧的导航栏选择“数据集成 FDI > 任务管理”，单击“创建普通任务”。
 - b. 在创建任务页面中填写数据集成任务的相关配置信息。
 - i. 配置任务基本信息。

表 2-11 任务基本信息

参数	配置说明
任务名称	填写任务的名称，自定义。
描述	填写任务的描述信息。
集成模式	选择数据集成的模式。使用MQS作为源端数据源时仅支持实时任务，此处选择“实时”。
任务标签	添加任务标签，用来对任务进行分类，方便用户快速查询。为简单起见，此处不设置。
企业项目	选择任务所属的企业项目，保持默认设置“default”。

- ii. 配置源端MQS数据源信息。

表 2-12 源端配置信息

参数	配置说明
源端实例	选择当前正在使用的ROMA Connect实例。
源端集成应用	选择在ROMA Connect注册设备中所创建的集成应用。
源端数据类型	选择“MQS”。
数据源名称	选择1中所配置的MQS数据源。
Topic名称	选择配置设备数据转发MQS中所创建的Topic。
是否解析	源端MQS与目标端MySQL的数据结构不一样，需要对数据进行解析和转换，此处选择“是”。
数据根字段	指从源端获取到的JSON格式数据中，元数据集上层公共字段的路径。此处不设置。
数据类型	选择从MQS数据源获取的数据格式类型，需要与MQS实际存储的数据格式一致。存储到MQS的设备数据为JSON格式，此处选择“JSON”。
消费偏移设置	选择“最新”，即集成最新获取的消息数据。
元数据	指从源端获取到的JSON格式数据中，要集成到目标端的数据元素。 以车位感应设备的数据为例，需要把车位感应设备信息和车位状态信息集成到目标端数据库中。 <ul style="list-style-type: none">别名：deviceld，类型：String，解析路径：data.deviceld别名：status，类型：String，解析路径：data.status
时区	选择MQS数据源使用的时区，以便ROMA Connect识别数据的时间戳。

iii. 配置目标端MySQL数据源信息。

表 2-13 目标端配置信息

参数	配置说明
目标端实例	选择当前正在使用的ROMA Connect实例。
目标端集成应用	选择在ROMA Connect注册设备中所创建的集成应用。
目标端数据类型	选择“MySQL”。
数据源名称	选择2中所配置的MySQL数据源。

参数	配置说明
目标端表	选择MySQL数据库中集成数据要写入的数据表。 选择数据表后，单击“选择表字段”，选择车位感应设备信息和车位状态信息要写入数据库的列字段。
批次号字段	此处不设置。

- iv. 配置MQS到MySQL的数据映射规则。
单击“自动Mapping”，自动建立源端和目标端数据字段间的映射规则。如果两端的数据表字段不一致，您需要为目标端字段选择对应的源端字段。
- c. 单击“保存”，完成数据集成任务的创建。
- d. 在“任务管理”页面，在刚创建的数据集成任务右侧单击“启动”，启动任务。

2.5 创建并开放数据 API

概述

通过创建数据API，把停车场管理系统中的车位状态数据开放给第三方应用，供车主、城市管理者了解停车场车位情况。

操作步骤

1. 创建后端。
 - a. 在左侧的导航栏选择“服务集成 APIC > 自定义后端”，在“后端列表”页签中单击“创建后端”。
 - b. 在创建后端页面填写后端相关配置信息，完成后单击“立即创建”。

表 2-14 后端配置

参数	配置说明
后端名称	填写后端的名称，自定义。
集成应用	选择在ROMA Connect注册设备中创建的集成应用。
后端请求方法	选择后端的请求方法，此处选择“GET”。
后端请求路径	填写后端的请求路径，格式如：/serviceName/interfaceName。
后端安全认证	选择后端的安全认证方式。本实践中开放的API与后端服务在同一实例内，此处选择“无认证”。
描述	填写后端的描述信息。
高级设置	高级设置保持默认设置。

后端创建完成后，页面自动跳转到该后端的编辑器页面，后端类型默认为数据后端。

2. 配置和部署数据后端。
 - a. 在编辑器页面左侧单击“添加数据源”。
 - b. 在添加数据源弹窗中配置数据源信息，完成后单击“立即添加”。

表 2-15 数据源配置

参数	配置说明
选择数据源	选择 创建MQS到数据库的集成任务 中接入的MySQL数据源。
选择执行语句类型	选择执行语句的类型，此处选择“SQL”。
高级设置	高级设置保持默认设置。

- c. 添加数据源后，在编辑器左侧选择数据源，然后您可以在右侧的语句编辑框中为数据源添加如下执行语句，从数据库中获取当前空余车位数量。
假设MySQL数据库中存储车位信息和车位状态信息的数据表名为“ParkingData”，表中车位状态字段为“Status”。

```
SELECT COUNT(Status) AS FreeNo FROM ParkingData WHERE Status="0";
```
 - d. 完成后在页面右上角单击“保存”，保存数据后端配置。
 - e. 在页面右上角单击“测试”，然后在下方的测试窗口中单击“立即测试”。
在“执行结果”栏查看后端的响应结果是否为如下JSON数据，其中xxx为统计返回的空余车位数量值。

```
{"default":[{"FreeNo":xxx}]}
```
 - f. 后端测试完成后，在页面右上角单击“部署”，然后在弹窗中单击“确定”，部署数据后端。
3. 发布数据API。
 - a. 后端部署完成后，在页面右上角单击“发布”。
 - b. 在发布弹窗中填写发布相关参数，完成后单击“立即发布”，为该后端创建一个对应的前端API，并发布到环境上。
API发布完成后，自动跳转到对应前端API的API运行页面。

表 2-16 发布后端

参数	配置说明
所属分组	选择前端API所属的API分组。若没有可用的API分组，可单击右侧的“创建分组”，创建一个API分组。
发布环境	选择默认的“RELEASE”环境

参数	配置说明
前端安全认证	选择前端API的安全认证方式，推荐使用APP认证方式。 <ul style="list-style-type: none"> APP认证：表示由ROMA Connect对API请求进行安全认证。用户调用API时，使用集成应用的Key和Secret进行API请求的安全认证。 华为IAM认证：表示由IAM对API请求进行安全认证。用户调用API时，使用Token或AK/SK进行API请求的安全认证。 自定义认证：表示使用自定义的函数API对API请求进行安全认证。 无认证：表示API请求不需要认证。
自定义认证	仅当“前端安全认证”选择“自定义认证”时需要配置。 选择已创建的前端类型自定义认证。
前端请求协议	选择前端API使用的请求协议，可选择“HTTP”、“HTTPS”和“HTTP&HTTPS”，传输重要或敏感数据时推荐使用HTTPS。
后端超时 (ms)	后端服务请求的超时时间，默认为60000ms。
重试次数	ROMA Connect调用后端服务失败后的重试次数。 <ul style="list-style-type: none"> 值为-1时，表示不开启重试功能。 值为0-10时，表示开启重试功能，并根据设置的值执行重试。
高级设置	
前端请求方法	选择前端API的请求方法。“ANY”表示该API支持任意请求方法。
前端请求路径	填写前端API的请求路径，格式如：/getUserInfo/userId。 请求路径中的内容区分大小写。
支持跨域 (CORS)	是否支持跨域访问API，此处选择不启用。

4. 为数据API绑定独立域名。
 - a. 在页面左上角选择“分组信息”，在“域名管理”下单击“绑定独立域名”。
 - b. 在绑定独立域名弹窗中填写域名配置信息，完成后单击“确定”。

表 2-17 独立域名配置

参数	配置说明
域名	填写要绑定的域名。

参数	配置说明
支持最小TLS版本	选择域名访问所使用的最小TLS版本。
支持http to https自动重定向	是否支持HTTP到HTTPS的重定向。 仅当API的请求协议选择“HTTPS”或“HTTP&HTTPS”，且独立域名已绑定SSL证书时重定向生效。 说明 由于浏览器限制，非GET或非HEAD方法的重定向可能导致数据丢失，因此API请求方法限定为GET或HEAD。

- c. （可选）如果数据API支持HTTPS请求协议，则需要为独立域名绑定SSL证书。否则跳过此步骤。
 - i. 在已绑定的独立域名右侧单击“选择SSL证书”。
 - ii. 在选择SSL证书弹窗中勾选要绑定的SSL证书，完成后单击“确定”，完成SSL证书的绑定。
若证书列表中无可用的SSL证书，可单击“创建SSL证书”，新增SSL证书。
5. 在页面左上角选择“API运行”，在页面中获取数据API的请求方法和URL，并提供给第三方应用使用。

2.6 设备接入 ROMA Connect

概述

停车场的车位感应设备通过集成开源MQTT客户端接入ROMA Connect，实现车位状态信息的上报。

操作步骤

1. 获取MQTT客户端。
根据您所使用的编程语言获取3.1或3.1.1版本的[Eclipse paho MQTT Client](#)。
2. 获取设备的接入信息。
 - a. 登录ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
 - b. 在左侧的导航栏选择“设备集成 LINK > 设备管理”，在页面中获取设备的连接信息。
 - MQTT/MQTTS连接地址：在设备管理页面上方获取设备的接入地址，如果使用MQTT接入，则获取“MQTT连接地址”；如果使用MQTTS接入，则获取“MQTTS连接地址”，并下载SSL证书。
 - 客户端ID/用户名/密码：在设备页签下找到对应的设备，并获取设备的“客户端ID”、“用户名”和“密码”。

- 设备的Topic信息：单击设备的名称进入设备详情页面，在Topic列表页签下，获取设备上报（发布）消息的Topic信息，Topic名称格式为：{产品标识}/out/{设备标识}。
3. 设备集成MQTT客户端。

对车位感应设备进行开发并集成Eclipse paho MQTT Client，在集成开发过程中写入设备的接入信息，并设置当车位状态变化时，车位感应设备上报设备信息和车位状态信息。设备上报的数据为JSON格式，格式如下所示。

deviceId为设备的物理标识信息，数据类型为String；status为车位状态信息，数据类型为String，值为0（未停车）或1（已停车）。

```
{
  "deviceId": "xxxxxx",
  "status": "0/1"
}
```

具体开发过程请参考[设备集成开发](#)。
 4. 设备接入ROMA Connect。

设备完成集成开发后，上电并联网，接入ROMA Connect。此时，在ROMA Connect实例控制台的“设备集成 LINK > 设备管理”页面中，可以看到设备的“状态”为“在线”。
 5. 设备上报车位状态数据。

当车位状态变化时，车位感应设备上报车位状态数据。

 - 在ROMA Connect实例控制台选择“消息集成MQS > 消息查询”，选择[配置设备数据转发MQS](#)中所创建的Topic，可看到转发到MQS的设备数据记录。
 - 在ROMA Connect实例控制台选择“数据集成 FDI > 任务管理”，单击[创建MQS到数据库的集成任务](#)中创建的任务名称，在任务信息的“日志 > 运行日志”页签下，可看到任务执行成功的记录。
 - 在停车场管理系统的数据库中，可查看写入数据表的车位状态数据。

3 企业数据以 API 形式开放共享

方案概述

通过数据API开放数据

通过函数API开放数据

3.1 方案概述

应用现状

随着信息化技术的不断发展，企业的业务系统越来越多，各业务系统间需要进行数据的互联互通，以提升企业的运作效率。如何实现企业内新老业务系统之间的数据安全互通，甚至是跨企业业务系统的数据安全互通，成为企业越来越重视的问题。

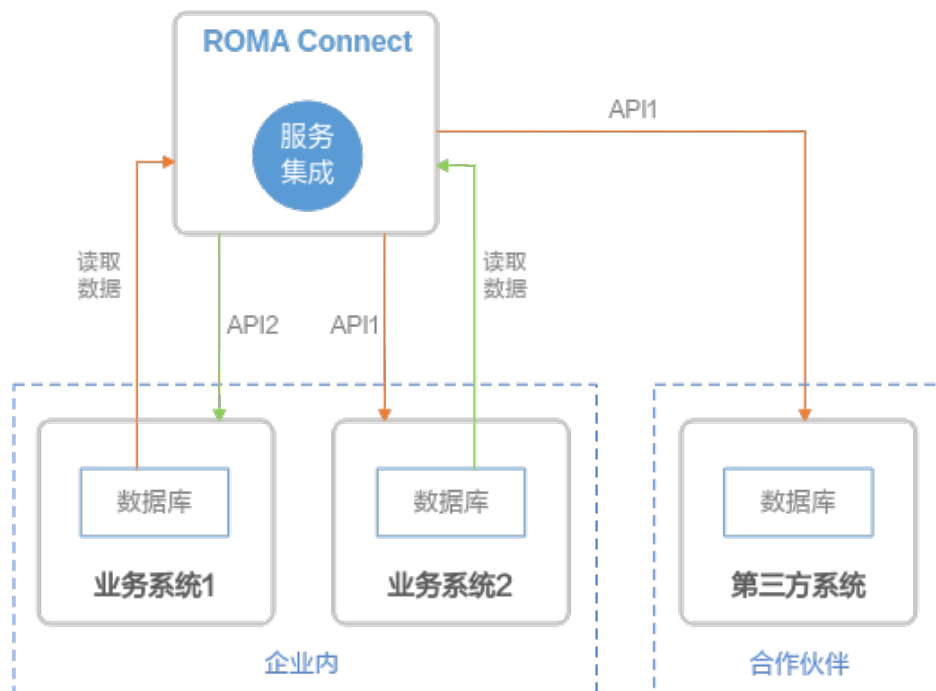
- 随着企业的跨区域发展，企业的业务系统也随之部署到各区域子公司中，总公司与各区域子公司的业务系统间需要时常进行业务交互。若不同业务系统间直接访问对方的数据库，操作方式过于复杂，且容易发生数据库信息泄露风险。
- 企业在日常运营的过程中，会持续的生产和累积数据资产，大部分企业的数据资产仅在企业内部使用，资产复用率低。若直接把数据库提供给合作伙伴或第三方访问，又难以保证数据库的对外安全访问问题。

解决方案

服务集成APIC是ROMA Connect的API集成组件，支持将已有后端服务、数据源和自定义函数封装成标准的RESTful API，并对外开放，简化分享数据或提供服务的过程，降低企业之间对接的成本。

本章节介绍如何通过ROMA Connect的服务集成，快速把数据库开放为RESTful API，不同业务系统间的数据以API方式互相开放访问，同时在开放API时可选择多种安全认证方式，实现跨网络跨地域应用系统数据间的安全访问。

- 对数据提供方而言，使用RESTful API形式对外开放其数据库中的数据，通过简单的操作，即可快速、低成本、低风险的开放数据或服务，提供业务增值能力。
- 对数据使用方而言，不再需要定制各种连接客户端，只需要使用一个具备简单RESTful请求发送能力的客户端，即可轻松获取各种数据库的数据。



ROMA Connect的服务集成直接读取数据库中的数据，把读取到的数据封装为一个RESTful API，并开放给企业内或企业外的其他业务系统。其他业务系统通过调用该API，即可获得该业务系统数据库中的数据。

ROMA Connect的服务集成把数据库数据封装为RESTful API有两种实现方式：

- 数据API：通过编写SQL脚本方式，读取数据库中的数据。操作简单方便，但灵活性较低。
- 函数API：通过编写JavaScript脚本方式，读取数据库中的数据。ROMA Connect提供了用于读取数据库数据的Java类DataSourceClient，在读取数据后可以通过JavaScript脚本对数据进行编排适配。操作较复杂，但灵活性更高。

3.2 通过数据 API 开放数据

前提条件

- 确保业务系统所在网络与ROMA Connect的网络互通。
若通过公网互通，ROMA Connect实例需绑定弹性IP。
- 确保业务系统的数据库为ROMA Connect支持的数据源类型。
数据API支持的数据源请参见[服务集成支持的数据源](#)。
- 准备一个可用的独立域名，用于作为API的访问域名。
确保已完成独立域名到“服务集成APIC连接地址”的域名解析配置，具体请参考[增加A类型记录集](#)。
若还没有独立域名，您可以通过[域名注册服务](#)申请域名，并通过[备案中心](#)完成域名备案。
- 如果数据API使用HTTPS请求协议，您还需要为独立域名申请SSL证书，并获取pem格式的SSL证书内容和密钥。

业务系统开放数据 API

1. 创建集成应用。

ROMA Connect实例中的资源都要归属到某个集成应用下，在创建其他资源前，您需要确保有一个集成应用。若已有可用的集成应用，可跳过此步骤。


- a. 登录ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
- b. 在左侧的导航栏选择“集成应用”，单击页面右上角的“创建集成应用”。
- c. 在创建集成应用弹窗中填写集成应用的“名称”，然后单击“确认”。

2. 接入数据源。

配置ROMA Connect接入业务系统的数据库，确保可以正常从数据库读取数据。不同类型数据源的接入配置不同，此处以MySQL数据库为例，其他类型数据库请参考[接入数据源](#)。

- a. 在左侧的导航栏选择“数据源管理”，单击页面右上角的“接入数据源”。
- b. 在接入数据源页面的“默认数据源”页签下，选择“MySQL”数据源，然后单击“下一步”。
- c. 在页面中配置数据源的连接信息。

表 3-1 数据源连接信息

参数	配置说明
数据源名称	填写数据源的名称，根据规划自定义。建议您按照一定的命名规则填写数据源名称，方便您快速识别和查找。
集成应用	选择数据源所归属的集成应用。
描述	单击“  ”，编辑数据源的描述信息。
连接模式	选择连接数据库模式。 <ul style="list-style-type: none">• 默认：由系统根据用户配置自动拼接数据源的连接字符串。• 专业：由用户自己输入JDBC格式的数据源连接字符串。
连接地址	仅当“连接模式”选择“默认”时需要配置。填写数据库的连接IP地址和端口号。
数据库名	仅当“连接模式”选择“默认”时需要配置。填写要接入的数据库名。
编码格式	仅当“连接模式”选择“默认”时可配置。数据库所使用的编码格式。
超时时间	仅当“连接模式”选择“默认”时可配置。连接数据库的超时时间，单位为秒。

参数	配置说明
连接字符串	仅当“连接模式”选择“专业”时需要配置。 填写MySQL数据库的JDBC格式连接串，例如： jdbc:mysql://{hostname}:{port}/{dbname}。 <ul style="list-style-type: none">• {hostname}为数据库的连接地址。• {port}为数据库的连接端口号。• {dbname}为要接入的数据库名。
用户名	填写连接数据库的用户名。
密码	填写连接数据库的用户密码。

- d. 完成数据源参数配置后，单击“开始检测”，进行数据源的连接测试。
- 若测试结果为“数据源连接成功！”，则继续下一步。
 - 若测试结果为“数据源连接失败！”，则检查和修改数据源连接参数配置，然后单击“重新检测”，直到连接成功为止。
- e. 单击“创建”，完成数据源的接入。
3. 创建数据后端。
- 通过自定义后端，把业务系统的数据转换为一个后端服务。
- a. 在左侧的导航栏选择“服务集成 APIC > 自定义后端”，在“后端列表”页签中单击“创建后端”。
- b. 在创建后端页面配置后端信息，完成后单击“立即创建”。

表 3-2 后端请求配置

参数	配置说明
后端名称	填写后端的名称，根据规划自定义。建议您按照一定的命名规则填写后端名称，方便您快速识别和查找。
集成应用	选择后端所属的集成应用。
后端请求方法	选择后端的请求方法，可选择“GET”、“POST”、“PUT”和“DELETE”。
后端请求路径	填写后端的请求路径，格式如：/getUserInfo/userId。 请求路径中的内容区分大小写。
后端安全认证	选择后端的安全认证方式。 <ul style="list-style-type: none">• 签名密钥：表示使用签名密钥的方式，对请求发送方进行认证。若使用签名密钥进行认证，调用该后端服务的API，也需要绑定相同的签名密钥。• 无认证：表示不对调用请求进行认证。
描述	填写后端的描述信息。
高级设置	

参数	配置说明
版本号	填写后端的版本号，自定义，用于区分不同版本的后端服务，如V1.0。
入参定义	根据实际需要定义后端服务的请求参数。 在“入参定义”下单击“新增入参定义”，添加自定义后端的请求参数。 <ul style="list-style-type: none">名称：请求参数的名称，自定义。参数位置：请求参数在后端服务请求中的位置，可选择“Headers”和“Parameters”。测试界面默认值：仅用于后续的自定义后端测试步骤，作为参数的默认测试值。在自定义后端部署时不生效。必填：在后端服务请求中，请求参数是否必填。描述：填写参数的描述信息。
返回类型	选择后端的响应数据格式，可选择“JSON”、“XML”和“STREAM”。
是否格式化	是否将响应消息体按选择的返回类型进行格式化。

后端创建完成后，页面自动跳转到该后端的编辑器页面，后端类型默认为数据后端。

- 在编辑器页面左侧单击“文件>新建数据后端>添加数据源”。
- 在添加数据源弹窗中配置数据源信息，完成后单击“立即添加”。

表 3-3 数据源配置

参数	配置说明
选择数据源	选择接入数据源中已创建的数据源。
选择执行语句类型	选择执行语句的类型，可选择“SQL”和“SP”（Stored Procedure）。 如果是Redis或MongoDB数据源，选择“SQL”，实际执行语句为NoSQL。
高级设置	
返回对象	填写返回对象的名称，执行语句的执行结果将封装在该对象中返回。
结果分页选项	执行语句的执行结果是否分页返回。 若同一个数据后端添加多个数据源时，不支持配置“结果分页选项”。
预编译	是否对执行语句进行预编译，可以防止SQL注入风险。

- e. 添加数据源后，在编辑器左侧选择数据源，然后您可以在右侧的语句编辑框中添加执行语句，通过执行语句获取要返回的数据。
 - 数据后端支持标准SQL语法，部分语法差异请参见[自定义后端开发（数据后端）](#)。
 - 如果是Redis或MongoDB数据源，实际为Redis或MongoDB的数据处理命令。

说明

若在SQL语句中同时引用多种数据类型的后端请求参数时，系统会默认把传入的参数转换为String类型。因此在执行SQL语句时，需要调用相应的函数对非String类型参数进行数据类型转换。

- f. 完成后在页面右上角单击“保存”，保存数据后端配置。
 - g. 测试后端功能。

在页面右上角单击“测试”，在下方的“测试参数”处，根据后端服务的定义添加请求参数，然后单击“立即测试”，发送请求。在“执行结果”栏查看后端的响应是否返回所期望的数据。
 - h. 后端测试完成后，在页面右上角单击“部署”，在确认弹窗中单击“确定”，部署后端服务。
4. 发布数据API。

把数据后端发布为数据API并部署到环境中，即可供其他业务系统调用。

 - a. 在页面右上角单击“发布”。
 - b. 在发布弹窗中配置前端API的相关信息，完成后单击“立即发布”，为该后端创建一个对应的前端API，并发布到环境上。

API发布完成后，自动跳转到对应前端API的API运行页面。

表 3-4 发布配置

参数	配置说明
所属分组	选择前端API所属的API分组。若没有可用的API分组，可单击右侧的“创建分组”，创建一个API分组。
发布环境	选择前端API要发布的环境，对外正式开放API时，应选择默认环境“RELEASE”。
前端安全认证	选择前端API的安全认证方式，推荐使用APP认证方式。 <ul style="list-style-type: none">• APP认证：表示由ROMA Connect对API请求进行安全认证。用户调用API时，使用集成应用的Key和Secret进行API请求的安全认证。• 华为IAM认证：表示由IAM对API请求进行安全认证。用户调用API时，使用Token或AK/SK进行API请求的安全认证。• 自定义认证：表示使用自定义的函数API对API请求进行安全认证。• 无认证：表示API请求不需要认证。

参数	配置说明
自定义认证	仅当“前端安全认证”选择“自定义认证”时需要配置。 选择已创建的前端类型自定义认证。
前端请求协议	选择前端API使用的请求协议，可选择“HTTP”、“HTTPS”和“HTTP&HTTPS”，传输重要或敏感数据时推荐使用HTTPS。
后端超时 (ms)	后端服务请求的超时时间，默认为60000ms。
重试次数	ROMA Connect调用后端服务失败后的重试次数。 <ul style="list-style-type: none">值为-1时，表示不开启重试功能。值为0-10时，表示开启重试功能，并根据设置的值执行重试。
高级设置	
前端请求方法	选择前端API的请求方法。“ANY”表示该API支持任意请求方法，用于获取数据库数据的API一般使用GET方法即可。
前端请求路径	填写前端API的请求路径，格式如：/getUserInfo/userId。 请求路径中的内容区分大小写。
支持跨域 (CORS)	是否支持跨域访问API。 浏览器出于安全性考虑，限制从页面脚本内发起的跨域请求，此时页面只能访问同源的资源。而CORS允许浏览器向跨域服务器发送XMLHttpRequest请求，从而实现跨域访问。跨域访问API请参见 配置跨域访问API 。

5. 为数据API绑定独立域名。
 - a. 在页面左上角选择“分组信息”，在“域名管理”下单击“绑定独立域名”。
 - b. 在绑定独立域名弹窗中填写域名配置信息，完成后单击“确定”。

表 3-5 独立域名配置

参数	配置说明
域名	填写要绑定的域名。
支持最小TLS版本	选择域名访问所使用的最小TLS版本。

参数	配置说明
支持http to https自动重定向	<p>是否支持HTTP到HTTPS的重定向。 仅当API的请求协议选择“HTTPS”或“HTTP&HTTPS”，且独立域名已绑定SSL证书时重定向生效。</p> <p>说明 由于浏览器限制，非GET或非HEAD方法的重定向可能导致数据丢失，因此API请求方法限定为GET或HEAD。</p>

- c. (可选) 如果数据API支持HTTPS请求协议，则需要为独立域名绑定SSL证书。否则跳过此步骤。
 - i. 在已绑定的独立域名右侧单击“选择SSL证书”。
 - ii. 在选择SSL证书弹窗中勾选要绑定的SSL证书，完成后单击“确定”，完成SSL证书的绑定。
若证书列表中无可用的SSL证书，可单击“创建SSL证书”，新增SSL证书。

其他业务系统调用 API

1. 向API提供方获取API的调用信息。
 - 获取API的请求信息。
API提供方可在ROMA Connect控制台选择“服务集成 APIC > API列表”，在列表中单击API的名称进入API运行页面，在页面中可获得API的请求方法和URL，在“前端配置”页签下，可获得API的请求参数。
 - 获取API的认证信息。
根据API的安全认证方式不同，需要获取不同的认证信息。
 - APP认证：
 - 签名认证：向API提供方获取该API所授权凭据的Key和Secret，以及用于调用API的SDK。
 - 简易认证：向API提供方获取该API所授权凭据的AppCode。
 - 其他认证：向API提供方获取该API所授权凭据的Key和Secret。
 - 华为IAM认证：通过云服务平台的账号凭证（账号密码或者AK/SK）进行认证。如果使用AK/SK进行认证，还需要向API提供方获取用于调用API的SDK。
 - 自定义认证：向API提供方获取请求参数中要携带的自定义认证信息。
 - 无认证：无需认证信息。
2. 根据获取的API调用信息，拼接并发送API请求，获取所需数据。

3.3 通过函数 API 开放数据

前提条件

- 确保业务系统所在网络与ROMA Connect的网络互通。
若通过公网互通，ROMA Connect实例需绑定弹性IP。
- 确保业务系统的数据库为ROMA Connect支持的数据源类型。
函数API支持的数据源请参见[服务集成支持的数据源](#)。
- 准备一个可用的独立域名，用于作为API的访问域名。
确保已完成独立域名到“服务集成APIC连接地址”的域名解析配置，具体请参考[增加A类型记录集](#)。
若还没有独立域名，您可以通过[域名注册服务](#)申请域名，并通过[备案中心](#)完成域名备案。
- 如果函数API使用HTTPS请求协议，您还需要为独立域名申请SSL证书，并获取pem格式的SSL证书内容和密钥。

业务系统开放函数 API

1. 创建集成应用。
ROMA Connect实例中的资源都要归属到某个集成应用下，在创建其他资源前，您需要确保有一个集成应用。若已有可用的集成应用，可跳过此步骤。
 - a. 登录ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
 - b. 在左侧的导航栏选择“集成应用”，单击页面右上角的“创建集成应用”。
 - c. 在创建集成应用弹窗中填写集成应用的“名称”，然后单击“确认”。
2. 创建函数后端。
通过自定义后端，把业务系统的数据转换为一个后端服务。
 - a. 在左侧的导航栏选择“服务集成 APIC > 自定义后端”，在“后端列表”页签中单击“创建后端”。
 - b. 在创建后端页面配置后端信息，完成后单击“立即创建”。

表 3-6 后端请求配置

参数	配置说明
后端名称	填写后端的名称，根据规划自定义。建议您按照一定的命名规则填写后端名称，方便您快速识别和查找。
集成应用	选择后端所属的集成应用。
后端请求方法	选择后端的请求方法，可选择“GET”、“POST”、“PUT”和“DELETE”。
后端请求路径	填写后端的请求路径，格式如：/getUserInfo/userId。 请求路径中的内容区分大小写。

参数	配置说明
后端安全认证	选择后端的安全认证方式。 <ul style="list-style-type: none">• 签名密钥：表示使用签名密钥的方式，对请求发送方进行认证。若使用签名密钥进行认证，调用该后端服务的API，也需要绑定相同的签名密钥。• 无认证：表示不对调用请求进行认证。
描述	填写后端的描述信息。
高级设置	
版本号	填写后端的版本号，自定义，用于区分不同的后端服务版本，如V1.0。
入参定义	根据实际需要定义后端服务的请求参数。 在“入参定义”下单击“新增入参定义”，添加自定义后端的请求参数。 <ul style="list-style-type: none">• 名称：请求参数的名称，自定义。• 参数位置：请求参数在后端服务请求中的位置，可选择“Headers”和“Parameters”。• 测试界面默认值：仅用于后续的自定义后端测试步骤，作为参数的默认测试值。在自定义后端部署时不生效。• 必填：在后端服务请求中，请求参数是否必填。• 描述：填写参数的描述信息。
返回类型	选择后端的响应数据格式，可选择“JSON”、“XML”和“STREAM”。
是否格式化	是否将响应消息体按选择的返回类型进行格式化。

后端创建完成后，页面自动跳转到该后端的编辑器页面，后端类型默认为数据后端。

- c. 在编辑器的左上角单击“文件 > 新建函数后端 > 调用Mysql数据源示例”，在弹窗中单击“确定”，切换后端类型为函数后端。
如果是Redis或MongoDB数据源，可选择“调用Redis数据源示例”。
- d. 在右侧的语句编辑框中根据系统提供的函数脚本示例，修改数据源相关接入配置。

```
importClass(com.roma.apic.livedata.client.v1.DataSourceClient);
importClass(com.roma.apic.livedata.config.v1.DataSourceConfig);
function execute(data){
var config = new DataSourceConfig()
config.setType("mysql")
config.setUrl("jdbc:mysql://127.0.0.1:3306/db?allowPublicKeyRetrieval=true")
config.setUser("xxxx")
config.setPassword("xxxx")
var ds = new DataSourceClient(config)
return ds.execute("SELECT * FROM person where name = ? and age = ?","Tom",20);
}
```

- 在setType、setUrl、setUser和setPassword处配置数据源的连接信息，具体参数配置说明请参考[DataSourceConfig类说明](#)。
 - 在ds.execute处设置SQL执行语句，通过执行语句获取要返回的数据。
- e. 完成后在页面右上角单击“保存”，保存函数后端配置。
- f. 测试后端功能。
在页面右上角单击“测试”，在下方的“测试参数”处，根据后端服务的定义添加请求参数，然后单击“立即测试”，发送请求。在“执行结果”栏查看后端的响应是否返回所期望的数据。
- g. 后端测试完成后，在页面右上角单击“部署”，在确认弹窗中单击“确定”，部署后端服务。
3. 发布函数API。
把函数后端发布为函数API并部署到环境中，即可供其他业务系统调用。
- a. 在页面右上角单击“发布”。
- b. 在发布弹窗中配置前端API的相关信息，完成后单击“立即发布”，为该后端创建一个对应的前端API，并发布到环境上。
API发布完成后，自动跳转到对应前端API的API运行页面。

表 3-7 发布配置

参数	配置说明
所属分组	选择前端API所属的API分组。若没有可用的API分组，可单击右侧的“创建分组”，创建一个API分组。
发布环境	选择前端API要发布的环境，对外正式开放API时，应选择默认环境“RELEASE”。
前端安全认证	选择前端API的安全认证方式，推荐使用APP认证方式。 <ul style="list-style-type: none">• APP认证：表示由ROMA Connect对API请求进行安全认证。用户调用API时，使用集成应用的Key和Secret进行API请求的安全认证。• 华为IAM认证：表示由IAM对API请求进行安全认证。用户调用API时，使用Token或AK/SK进行API请求的安全认证。• 自定义认证：表示使用自定义的函数API对API请求进行安全认证。• 无认证：表示API请求不需要认证。
自定义认证	仅当“前端安全认证”选择“自定义认证”时需要配置。 选择已创建的前端类型自定义认证。
前端请求协议	选择前端API使用的请求协议，可选择“HTTP”、“HTTPS”和“HTTP&HTTPS”，传输重要或敏感数据时推荐使用HTTPS。
后端超时 (ms)	后端服务请求的超时时间，默认为60000ms。

参数	配置说明
重试次数	ROMA Connect调用后端服务失败后的重试次数。 <ul style="list-style-type: none">值为-1时，表示不开启重试功能。值为0-10时，表示开启重试功能，并根据设置的值执行重试。
高级设置	
前端请求方法	选择前端API的请求方法。“ANY”表示该API支持任意请求方法，用于获取数据库数据的API一般使用GET方法即可。
前端请求路径	填写前端API的请求路径，格式如：/getUserInfo/userId。 请求路径中的内容区分大小写。
支持跨域 (CORS)	是否支持跨域访问API。 浏览器出于安全性考虑，限制从页面脚本内发起的跨域请求，此时页面只能访问同源的资源。而CORS允许浏览器向跨域服务器发送XMLHttpRequest请求，从而实现跨域访问。跨域访问API请参见 配置跨域访问API 。

4. 为函数API绑定独立域名。

- 在页面左上角选择“分组信息”，在“域名管理”下单击“绑定独立域名”。
- 在绑定独立域名弹窗中填写域名配置信息，完成后单击“确定”。

表 3-8 独立域名配置

参数	配置说明
域名	填写要绑定的域名。
支持最小TLS版本	选择域名访问所使用的最小TLS版本。
支持http to https自动重定向	是否支持HTTP到HTTPS的重定向。 仅当API的请求协议选择“HTTPS”或“HTTP&HTTPS”，且独立域名已绑定SSL证书时重定向生效。 说明 由于浏览器限制，非GET或非HEAD方法的重定向可能导致数据丢失，因此API请求方法限定为GET或HEAD。

- (可选) 如果函数API支持HTTPS请求协议，则需要为独立域名绑定SSL证书。否则跳过此步骤。
 - 在已绑定的独立域名右侧单击“选择SSL证书”。
 - 在选择SSL证书弹窗中勾选要绑定的SSL证书，完成后单击“确定”，完成SSL证书的绑定。

若证书列表中无可用的SSL证书，可单击“创建SSL证书”，新增SSL证书。

其他业务系统调用 API

1. 向API提供方获取API的调用信息。
 - 获取API的请求信息。

API提供方可在ROMA Connect控制台选择“服务集成 APIC > API列表”，在列表中单击API的名称进入API运行页面，在页面中可获得API的请求方法和URL，在“前端配置”页签下，可获得API的请求参数。
 - 获取API的认证信息。

根据API的安全认证方式不同，需要获取不同的认证信息。

 - APP认证：
 - 签名认证：向API提供方获取该API所授权凭据的Key和Secret，以及用于调用API的SDK。
 - 简易认证：向API提供方获取该API所授权凭据的AppCode。
 - 其他认证：向API提供方获取该API所授权凭据的Key和Secret。
 - 华为IAM认证：通过云服务平台的账号凭证（账号密码或者AK/SK）进行认证。如果使用AK/SK进行认证，还需要向API提供方获取用于调用API的SDK。
 - 自定义认证：向API提供方获取请求参数中要携带的自定义认证信息。
 - 无认证：无需认证信息。
2. 根据获取的API调用信息，拼接并发送API请求，获取所需数据。

4 跨系统业务数据集成与转换

方案概述

配置系统间数据集成

4.1 方案概述

应用现状

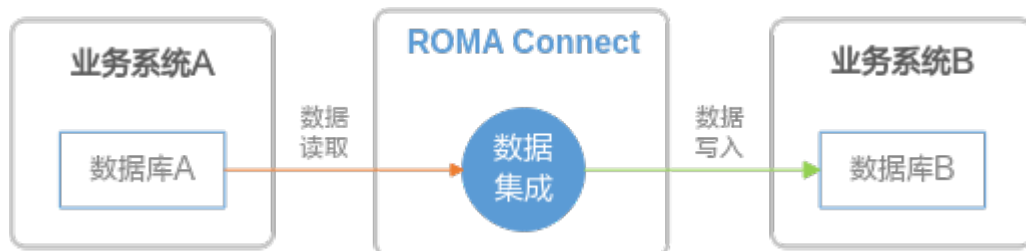
在数字化转型过程中，一些大型企业的业务系统数据格式多样化，难以在业务系统间形成有效信息传输。如何实现多种数据类型之间灵活、快速的数据集成和转换，成为企业数字化转型所面临的重要问题。

比如企业中不同业务系统数据的存储格式、存储位置各不相同，当一个业务系统需要获取另一个业务系统的某些数据时，就会出现获取的数据内容不符合要求，无法直接与自己数据结合使用等问题，不同业务系统间的数据共用难度大，效率低，影响业务发展。

解决方案

数据集成FDI是ROMA Connect的数据集成组件，支持多种数据类型之间灵活、快速、无侵入式的数据集成和转换，为业务系统之间建立有效的数据连接。

本章节介绍如何通过ROMA Connect的数据集成，把业务系统A中的数据进行格式转换后，集成到业务系统B的数据库中使用。后续根据制定的同步计划，定时把业务系统A中新增的数据同步到业务系统B中，保证了两个系统间的数据同步。



ROMA Connect的数据集成直接读取业务系统A中数据库的数据，并根据集成任务的配置，筛选出指定数据并进行数据转换，然后把转换后的数据写入到业务系统B的数据库中。

4.2 配置系统间数据集成

前提条件

- 确保业务系统数据库所在网络与ROMA Connect的网络互通。
若通过公网互通，ROMA Connect实例需绑定弹性IP。
- 确保源端和目标端业务系统的数据库为ROMA Connect支持的数据源类型
数据集成支持的数据源请参见[数据集成支持的数据源](#)。
- ROMA Connect具备向目标端数据库写入数据的权限。

配置数据集成任务

1. 创建集成应用。

ROMA Connect实例中的资源都要归属到某个集成应用下，在创建其他资源前，您需要确保有一个集成应用。若已有可用的集成应用，可跳过此步骤。

- a. 登录ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
- b. 在左侧的导航栏选择“集成应用”，单击页面右上角的“创建集成应用”。
- c. 在创建集成应用弹窗中填写集成应用的“名称”，然后单击“确认”。

2. 接入数据源。

配置ROMA Connect接入业务系统的数据库，确保可以正常从数据库读取和写入数据。

不同类型数据源的接入配置不同，此处以Kafka作为源端数据库，MySQL作为目标端数据库为例进行说明，其他类型数据库请参考[接入数据源](#)。

接入源端Kafka数据源：

- a. 在左侧的导航栏选择“数据源管理”，单击页面右上角的“接入数据源”。
- b. 在接入数据源页面的“默认数据源”页签下，选择“Kafka”类型的数据源，然后单击“下一步”。
- c. 在页面中配置数据源的连接信息。

表 4-1 数据源连接信息

参数	配置说明
数据源名称	填写数据源的名称，根据规划自定义。建议您按照一定的命名规则填写数据源名称，方便您快速识别和查找。
集成应用	选择数据源所归属的集成应用。
描述	编辑数据源的描述信息。
连接地址	填写Kafka的连接地址和端口号。 如果Kafka有多个副本（Broker），可单击“添加地址”填写多个连接地址。
是否SSL	ROMA Connect与Kafka的连接是否使用SSL认证加密。

参数	配置说明
SSL用户名/应用Key	仅当“是否SSL”选择“是”时需要配置。 SSL认证所使用的用户名。
SSL密码/应用Secret	仅当“是否SSL”选择“是”时需要配置。 SSL认证所使用的用户密码。

- d. 完成数据源接入配置后，单击“开始检测”，进行数据源的连接测试。
- 若测试结果为“数据源连接成功！”，则继续下一步。
 - 若测试结果为“数据源连接失败！”，则检查数据源状态和数据源连接参数配置，然后单击“重新检测”，直到连接成功为止。
- e. 单击“创建”，完成数据源的接入。

接入目标端MySQL数据源：

- a. 在“数据源管理”页面单击右上角的“接入数据源”。
- b. 在接入数据源页面的“默认数据源”页签下，选择“MySQL”数据源，然后单击“下一步”。
- c. 在页面中配置数据源的连接信息。

表 4-2 数据源连接信息

参数	配置说明
数据源名称	填写数据源的名称，根据规划自定义。建议您按照一定的命名规则填写数据源名称，方便您快速识别和查找。
集成应用	选择数据源所归属的集成应用。
描述	编辑数据源的描述信息。
连接模式	选择连接数据库模式。 <ul style="list-style-type: none">• 默认：由系统根据用户配置自动拼接数据源的连接字符串。• 专业：由用户自己输入JDBC格式的数据源连接字符串。
连接地址	仅当“连接模式”选择“默认”时需要配置。 填写数据库的连接IP地址和端口号。
数据库名	仅当“连接模式”选择“默认”时需要配置。 填写要接入的数据库名。
编码格式	仅当“连接模式”选择“默认”时可配置。 数据库所使用的编码格式。
超时时间	仅当“连接模式”选择“默认”时可配置。 连接数据库的超时时间，单位为秒。

参数	配置说明
连接字符串	仅当“连接模式”选择“专业”时需要配置。 填写MySQL数据库的JDBC格式连接串，例如： jdbc:mysql://{hostname}:{port}/{dbname}。 <ul style="list-style-type: none">• {hostname}为数据库的连接地址。• {port}为数据库的连接端口号。• {dbname}为要接入的数据库名。
用户名	填写连接数据库的用户名。
密码	填写连接数据库的用户密码。

- d. 完成数据源参数配置后，单击“开始检测”，进行数据源的连接测试
- 若测试结果为“数据源连接成功！”，则继续下一步。
 - 若测试结果为“数据源连接失败！”，则检查数据源状态和数据源连接参数配置，然后单击“重新检测”，直到连接成功为止。
- e. 单击“创建”，完成源端数据源的接入。
3. 创建数据集成任务。

ROMA Connect通过数据集成任务，读取源端数据库中的数据，进行数据结构转换后，写入目标端数据库中。

- a. 在左侧的导航栏选择“数据集成 FDI > 任务管理”，单击页面的“创建普通任务”。
- b. 在创建任务页面中配置任务基本信息。

表 4-3 任务基本信息

参数	配置说明
任务名称	填写任务的名称，根据规划自定义。建议您按照一定的命名规则填写任务名称，方便您快速识别和查找。
描述	填写任务的描述信息。
集成模式	选择数据集成的模式。 <ul style="list-style-type: none">• 定时：指数据集成任务根据任务计划，在指定的时间执行任务，将源端数据集成到目标端。• 实时：指数据集成任务不断检测源端数据的变更，并实时将新增的数据集成到目标端。 使用Kafka作为源端数据源时仅支持实时任务，此处选择“实时”。
任务标签	添加任务标签，用来对任务进行分类，方便用户快速查询。
企业项目	选择任务所属的企业项目，保持默认设置“default”。

c. 配置源端数据信息。

表 4-4 源端信息

参数	配置说明
源端实例	选择当前正在使用的ROMA Connect实例。
源端集成应用	选择源端Kafka数据源所属的集成应用，在 接入数据源 时已配置。
源端数据类型	选择“Kafka”。
数据源名称	选择 接入数据源 中已配置的Kafka数据源。
Topic名称	选择要读取数据的Topic名称。
是否解析	<p>获取到的源端数据是否由ROMA Connect做进一步的数据解析。</p> <ul style="list-style-type: none">若选择是，则ROMA Connect根据配置的解析规则，对获取到的源端数据解析后再集成到目标端。若选择否，则ROMA Connect会直接透传获取到的源端数据，并集成到目标端。 <p>本实践中需要对源端数据进行数据结构转换后，再写入目标端数据库，此处选择“是”。</p>
数据根字段	指从源端获取到的JSON或XML格式数据中，元数据集上层公共字段的路径。此处不设置。
数据类型	选择从Kafka数据源获取的数据格式类型，需要与Kafka实际存储的数据格式一致。
消费偏移设置	选择集成最早获取的消息数据还是最新获取的消息数据。
元数据	<p>指从源端获取到的JSON或XML格式数据中，要集成到目标端的每一个底层key-value型数据元素。</p> <ul style="list-style-type: none">别名：对元数据的自定义名称。类型：元数据的数据类型，需要与源端数据中对应字段的数据类型一致。解析路径：由于数据根字段不设置，此处填写元数据的完整路径。 <p>以JSON格式数据<code>{"a": {"b": "xx", "c": "xx"}}</code>为例，参数b和c为底层数据元素，其解析路径分别为a.b和a.c。</p>
时区	选择Kafka数据源使用的时区，以便ROMA Connect识别数据的时间戳，默认为GMT+8:00时区（北京时间）。

d. 配置目标端数据信息。

表 4-5 目标端信息

参数	配置说明
目标端实例	配置为当前正在使用的ROMA Connect实例，源端实例配置后目标端自动关联，无需配置。
目标端集成应用	选择目标端MySQL数据源所属的集成应用，在 接入数据源 时已配置。
目标端数据类型	选择“MySQL”。
数据源名称	选择 接入数据源 中已配置的MySQL数据源。
目标端表	选择集成数据要写入MySQL数据库中的数据表。 选择数据表后，单击“选择表字段”，可以选择要写入数据的列字段。
批次号字段	选择目标表中类型为String并且字段长度大于14的字段作为批次号字段，且该批次号字段不能和Mapping信息中的目标字段重复。 该字段值将为一个随机数，用来标识同一批次的的数据，同一批次插入的数据具有相同的批次号，表示这些数据是同一批次插入的，可以用来定位或者解析回滚。
是否清空表	开启后，每次调度任务会先清空目标端表。

- e. 配置源端到目标端的数据映射规则。
单击“自动Mapping”，自动建立源端和目标端数据字段间的映射规则。如果两端的数据表字段不一致，您需要为目标端字段选择对应的源端字段。
- f. 单击“保存”，完成数据集成任务的创建。

启动数据集成任务

数据集成任务创建后，“任务状态”为“停止”状态，此时任务不会执行，需先启动任务。

- 实时任务启动后，ROMA Connect会不断检测源端数据变更。首次执行时，会把所有符合条件的源端数据集成到目标端，后续若检测到新增数据，则将其集成到目标端。
 - 定时任务启动后，ROMA Connect按照任务计划进行数据集成。首次执行时，会把所有符合条件的源端数据集成到目标端，后续根据任务配置，集成符合条件的全量数据或只集成增量数据。
1. 启动数据集成任务。
在数据集成任务左侧勾选上要启动的任务，单击任务列表上方的“启动”，启动任务。任务启动成功后，“任务状态”变为“启动”。
 2. 任务启动后，任务状态为“执行中”，待任务状态为“执行成功”，则表示该数据集成任务执行完成。
 3. 执行完成后，可在目标端数据库的数据表中查看集成同步过来的数据。

5 搭建企业业务开放中台

方案概述

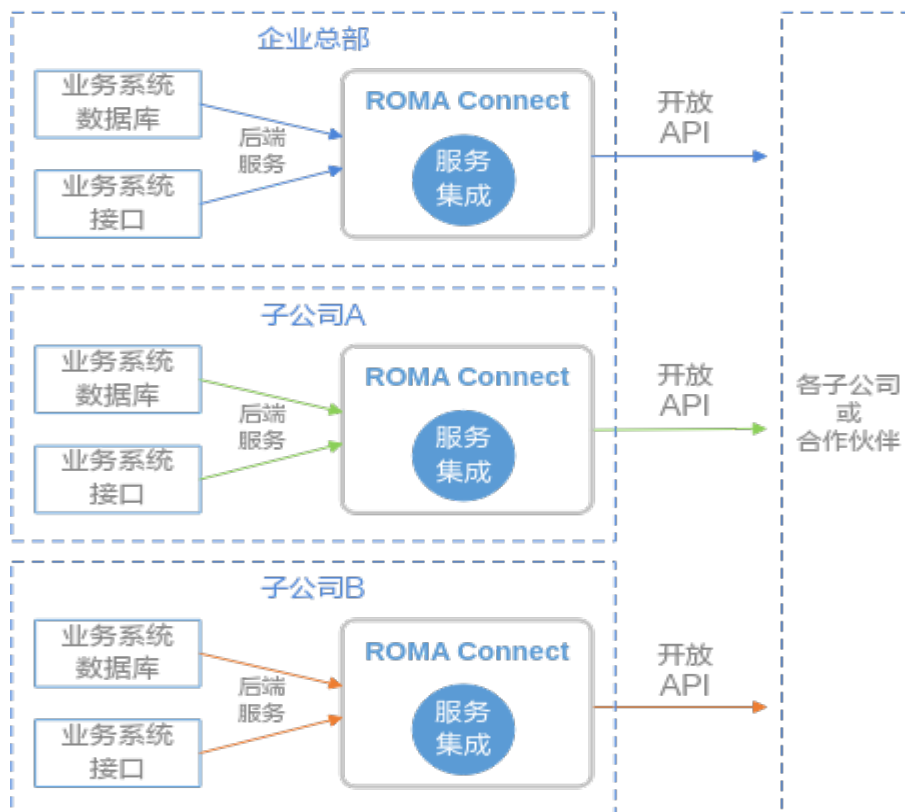
配置业务开放中台

5.1 方案概述

应用现状

使用ROMA Connect进行业务系统数字化改造的一些大型企业中，不同的子公司或部门基于独立维护的要求，都有自己的ROMA Connect实例。各子公司或部门的业务系统都对接到自己的ROMA Connect实例，通过API的形式实现数字资产的互通共享。

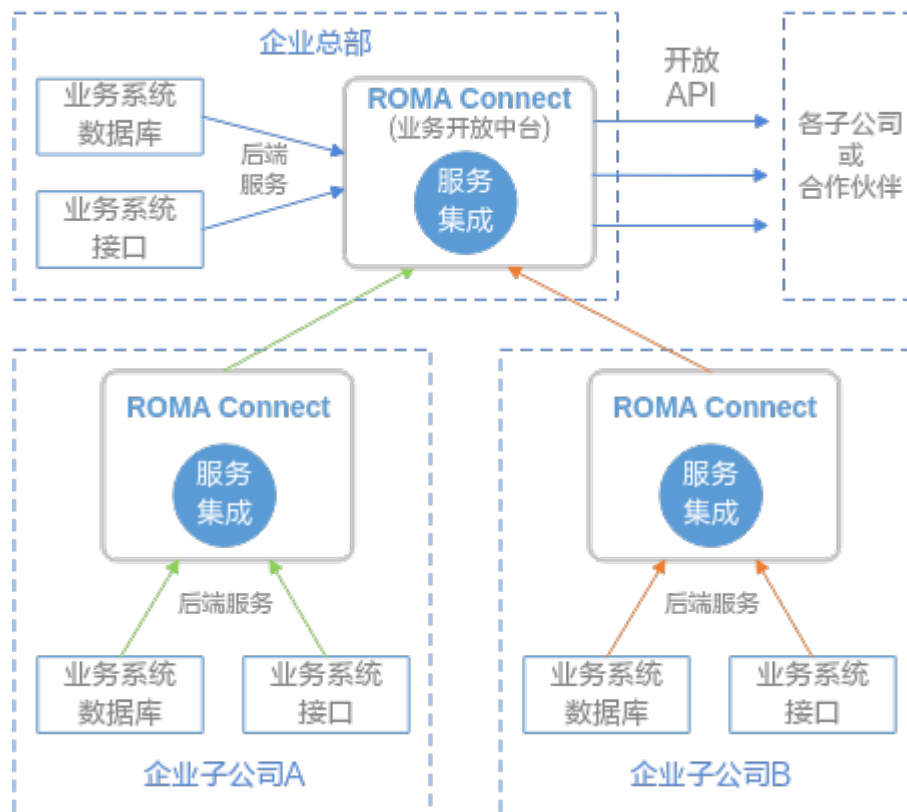
- 由于各子公司或部门在运作上相互独立，其ROMA Connect实例可能会归属到不同的VPC上，而且每个ROMA Connect实例都有自己的访问地址。某个子公司或部门要通过内网访问其他子公司或部门的开放API，就需要打通到多个不同VPC的网络，访问多个不同的地址，在操作配置上十分复杂。
- 不同子公司或部门在开放API时，定义的API认证方式、格式等会各不相同。某个子公司或部门要访问其他子公司或部门的开放API时，需要使用不同的认证信息，增加了子公司或部门间数字资产的共享难度。



解决方案

ROMA Connect的服务集成APIC提供了API级联的能力，通过API级联能力把企业内所有子公司或部门要开放的API级联到一个ROMA Connect实例上，把该ROMA Connect实例作为企业内的业务开放中台。所有的子公司或部门都调用业务开放中台的API来获取共享的数字资产，实现了调用入口的统一，认证信息的统一，提升了数字资产的共享效率，降低共享难度。

本章节介绍如何通过服务集成的API级联能力，把子公司A和子公司B ROMA Connect实例的API级联到公司总部的ROMA Connect实例上，把公司总部的ROMA Connect实例作为业务开放中台，所有数字资产都通过业务开放中台对外共享。



5.2 配置业务开放中台

前提条件

- 级联的实例间网络可以互通。
若级联的两个实例在不同VPC，可以通过创建VPC对等连接，将两个VPC的网络打通，实现跨VPC访问，具体请参考[VPC对等连接说明](#)。
- 级联的实例间跨网互通时，如果经过网闸设备，需要将中台实例和子公司实例的地址和端口提供给网闸设备，并在网闸设备中采用TCP协议摆渡方式打通网络路径。也可以通过专用VPN或隧道来实现跨网互通。

配置实例间的级联关系

配置中台实例到各子公司实例的级联关系，使中台实例可使用各子公司实例的开放API作为后端服务。

1. 在子公司实例配置级联功能。
 - a. 登录子公司实例A所在的ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
 - b. 在“实例信息”页面选择“配置参数”页签，并找到级联功能开关参数“cascade”。
 - c. 单击参数右侧的“编辑”，把参数运行值设置为“on”，然后单击“保存”，打开级联功能开关。
 - d. 单击参数左侧的“∨”，展开并配置级联相关配置参数。

参数	参数默认值	参数范围	参数运行值	更新时间	操作
cascade_auth_key			2021/11/03 09:43:52 GMT+08:00	2021/11/03 09:43:52 GMT+08:00	编辑
cascade_instance_ids			2021/11/03 09:43:52 GMT+08:00	2021/11/03 09:43:52 GMT+08:00	编辑

表 5-1 子公司实例配置

参数	配置说明
cascade_auth_key	级联认证的加密密钥，用于级联API之间的调用认证。子公司实例和中台实例的cascade_auth_key值必须配置一致。
cascade_instance_ids	填写中台实例的实例ID。

- e. 重复1.a-1.d，配置在子公司B的ROMA Connect实例上配置级联功能。
2. 在中台实例配置级联功能。
 - a. 登录中台实例所在的ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
 - b. 在“实例信息”页面选择“配置参数”页签，并找到级联功能开关参数“cascade”。
 - c. 单击参数右侧的“编辑”，把参数运行值设置为“on”，然后单击“保存”，打开级联功能开关。
 - d. 单击参数左侧的“∨”，展开并配置级联相关配置参数。

参数	参数默认值	参数范围	参数运行值	更新时间	操作
cascade_auth_key			2021/11/03 09:43:52 GMT+08:00	2021/11/03 09:43:52 GMT+08:00	编辑
cascade_instance_ids			2021/11/03 09:43:52 GMT+08:00	2021/11/03 09:43:52 GMT+08:00	编辑

表 5-2 中台实例配置

参数	配置说明
cascade_auth_key	级联认证的加密密钥，用于级联API之间的调用认证。公司实例和中台实例的cascade_auth_key值必须配置一致。
cascade_instance_ids	中台实例中无需配置该参数。

配置级联实例间的负载通道

配置中台实例到各子公司实例的负载通道，当中台实例调用子公司实例的API作为后端服务时，可使用专属的认证通道，避免了认证冲突问题。

1. 在中台实例控制台左侧的导航栏选择“服务集成 APIC > API策略”，在“负载通道”页签中单击“创建负载通道”。

2. 在新建负载通道页面配置负载通道基本信息。

表 5-3 基本信息配置

参数	配置说明
通道名称	填写负载通道的名称，根据规划自定义。建议您按照一定的命名规则填写负载通道名称，方便您快速识别和查找。
端口	填写负载通道的服务器访问端口号。根据子公司实例中API使用的请求协议来确定端口号，HTTP协议使用80端口，HTTPS协议使用443端口。
分发算法	选择后端服务请求的分发算法，负载通道通过分发算法确定请求被发送到哪台服务器上。 <ul style="list-style-type: none">● 加权轮询：结合弹性服务器权重值，将请求轮流转发到每一台服务器。● 加权最小连接：结合弹性服务器权重值，将请求转发到具有最少活跃连接数的那台服务器。● 源地址哈希：由请求的来源IP地址决定请求被转发到哪一台云服务器，相同源地址的请求始终会转发到同一台服务器，除非该云服务器不可用。● URI哈希：由请求的路径决定请求被转发到哪一台云服务器，相同路径的请求始终会转发到同一台服务器，除非该云服务器不可用。

3. 配置通道内服务器。
 - a. 选择“服务器添加方式”，访问子公司实例的API时需要选择“指定服务器地址”。
 - b. 单击“创建服务器分组”，在创建服务器分组弹窗中配置分组信息，完成后单击“确定”。

您可以根据需要把服务器划分到多个分组中。

表 5-4 服务器分组配置

参数	配置说明
分组名称	填写服务器分组的名称，根据规划自定义。建议您按照一定的命名规则填写分组名称，方便您快速识别和查找。
权重	填写服务器分组的权重值，权重值越大，转发到该分组下服务器的请求数量越多。
描述	填写分组的描述信息。

- c. 单击“添加后端服务器地址”。
- d. 在页面中配置后端服务器信息。

表 5-5 后端服务器配置

参数	配置说明
后端服务器地址	填写子公司实例的API访问地址。 <ul style="list-style-type: none">● 若填写IP地址：<ul style="list-style-type: none">- 级联的实例间基于弹性IP实现公网互通时，则填写子公司实例的“弹性IP地址”。- 级联的实例间基于VPC内网地址实现内网互通时，则填写子公司实例的“服务集成 APIC连接地址”。● 若填写域名，则填写API的访问域名地址。
是否备用节点	开启后对应后端服务器为备用节点，仅当非备用节点全部故障时工作。
端口	填写后端服务器的访问端口号，端口为0时使用负载通道的端口。
启停状态	选择是否启用服务器，只有启用后，请求才会分发到该服务器上。

4. 配置健康检查配置。

健康检查配置默认打开，若不需要执行健康检查，可关闭健康检查配置。

表 5-6 健康检查配置

参数	配置说明
协议	选择健康检查所使用的协议，可选择“TCP”、“HTTP”和“HTTPS”。
双向认证	仅当“协议”选择“HTTPS”时可配置。 选择是否在ROMA Connect和后端服务器间启用双向认证。
路径	仅当“协议”选择“HTTP”或“HTTPS”时需要配置。 填写健康检查的URL地址路径。
请求类型	仅当“协议”选择“HTTP”或“HTTPS”时需要配置。 选择健康检查所使用的HTTP请求方法，可选择“GET”和“HEAD”。
检查端口	健康检查的目标端口，默认使用负载通道配置的端口号。
正常阈值	执行健康检查时，判定服务器状态正常的连续检查成功次数。例如“正常阈值”设置为“2”，则在连续2次检查成功时，判定服务器的状态正常。
异常阈值	执行健康检查时，判定服务器状态异常的连续检查失败次数。例如“异常阈值”设置为“5”，则在连续5次检查失败时，判定服务器的状态异常。

参数	配置说明
超时时间	健康检查的响应超时时间，单位为秒，超过时间无响应则认为检查失败。
间隔时间	连续两次检查的间隔时间，单位为秒。
HTTP响应码	仅当“协议”选择“HTTP”或“HTTPS”时需要配置。 服务器在返回指定的HTTP响应码时，判定响应成功。可同时指定多个响应码。

- 单击“完成”，完成负载通道的创建。
- 重复1-5配置中台实例到其他子公司实例的负载通道。

在中台实例创建 API

在中台实例创建并开放API，使用子公司实例中的API作为后端服务。各子公司或合作伙伴通过调用中台实例开放的API，实现对各子公司能力的调用。

创建API的过程请参考[创建API](#)，仅后端配置有如下区别。

表 5-7 API 后端服务配置

参数	配置说明
后端服务类型	选择后端服务的类型。使用子公司实例的API作为后端服务时，需选择“HTTP/HTTPS”。
负载通道	是否使用负载通道访问后端服务。使用子公司实例的API作为后端服务时，需选择“使用”。
URL	配置后端服务的URL。 <ul style="list-style-type: none">请求方法：选择后端服务的请求方法，根据子公司实例中API的请求方法选择。请求协议：选择后端服务使用的请求协议，根据子公司实例中API的请求协议选择。负载通道：选择访问后端服务所使用的负载通道，选择配置级联实例间的负载通道所创建的负载通道。路径：填写后端服务的请求路径，格式如：/getUserInfo/{userId}。请求路径中可包含Path参数，以{参数名}形式表示。 如果请求路径中需要携带环境变量，则使用“#变量名#”的形式将环境变量添加到请求路径中，如/#path#。环境变量名严格区分大小写，支持添加多个环境变量，如/#path##request#。
级联标识	是否使用级联方式访问后端服务，选择开启。
自定义host头域	自定义后端服务请求中的Host头域。若 配置级联实例间的负载通道 中创建负载通道时，“后端服务器地址”填写了IP地址，则此处需要填写子公司实例API的域名。

参数	配置说明
后端超时 (ms)	后端服务请求的超时时间，默认为5000ms。
重试次数	ROMA Connect调用后端服务失败后的重试次数。 <ul style="list-style-type: none">• 值为-1时，表示不开启重试功能。• 值为0-10时，表示开启重试功能，并根据设置的值执行重试。重试次数不能大于负载通道中已启用的后端服务器个数。
TLS双向认证	仅当“协议”选择“HTTPS”时可配置。 选择是否在ROMA Connect和后端服务间启用双向认证。使用子公司实例的API作为后端服务时，不启用双向认证。
后端认证	选择是否启用后端认证。使用子公司实例的API作为后端服务时，不启用后端认证。

6 使用 Filebeat+ELK 实现 Site 协同版实例的 API 日志监控

- 方案概述
- 资源规划
- 准备Docker环境和镜像
- 修改Filebeat和ELK配置
- 部署Filebeat和ELK
- 配置索引生命周期策略（可选）
- 使用Kibana查看日志

6.1 方案概述

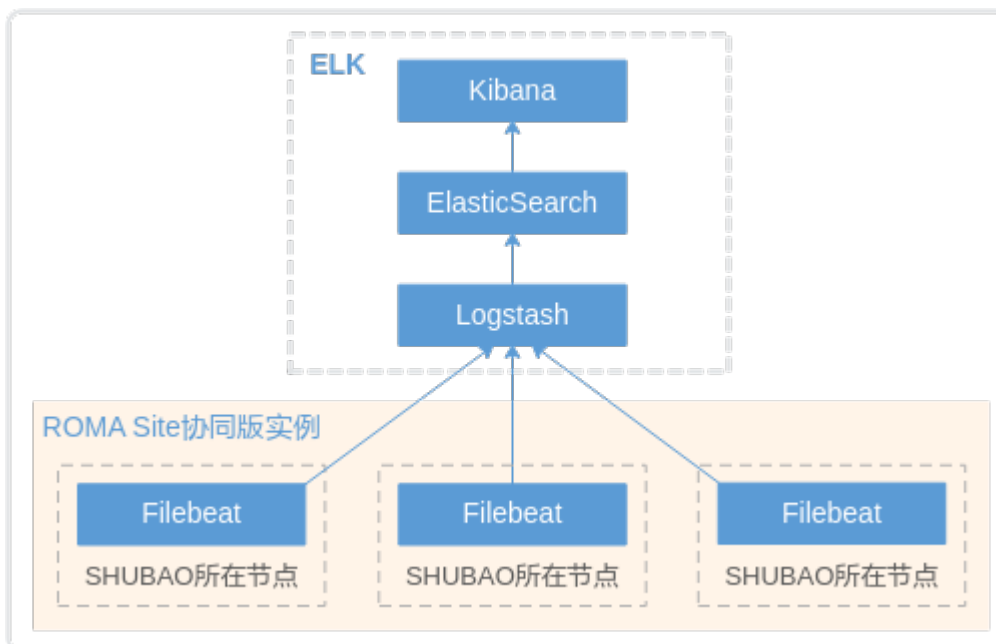
应用场景

ROMA Site协同版是ROMA Connect的边缘形态，部署于用户IDC和第三方公有云，与云端ROMA Connect反向互联，不破坏企业的安全边界。

服务集成APIC的API监控分析能力依赖于云日志服务（LTS）的日志统计能力，由于ROMA Site协同版实例为边缘部署，需要将API的调用日志回传到华为云上才能实现API的监控分析。由于传输的数据量与API请求量相关，请求数据量过大，回传链路无法承载。因此ROMA Site协同版实例暂不支持查看API的调用日志和调用统计数据。

方案架构

通过在ROMA Site协同版本地部署Filebeat+ELK，直接获取ROMA Site协同版实例中的API调用日志，并通过ELK的Kibana可视化界面查看API的调用日志以及实现API调用日志的汇聚统计。



1. Filebeat部署在APIC的SHUBAO组件所在节点上，监控API调用日志的产生，并将其发送到ELK的Logstash进行处理。
2. ELK部署在独立服务器中，ELK的Logstash接收到日志数据后，根据设置的过滤解析规则，对日志数据进行拆分和过滤，然后将解析后的日志数据发送到ELK的ElasticSearch进行存储。
3. ELK的ElasticSearch接收到日志数据后，生成日志数据的索引并存储数据，为Kibana的可视化显示提供聚合搜索和数据分析的支撑。
4. 用户使用ELK的Kibana创建日志视图，基于Logstash拆分的日志属性进行API调用日志的查询和统计。

6.2 资源规划

Filebeat部署在APIC的SHUBAO组件所在节点上，ELK部署在独立服务器中。在搭建Filebeat+ELK系统前，需要对Filebeat和ELK所需的服务器资源进行评估，以准备相关服务器资源。

API调用日志文件最大为200MB，达到上限后会自动生成新的日志文件，以每天产生5个或10个新日志文件为例，Filebeat和ELK对服务器的资源需求如表6-1所示。

表 6-1 服务器资源需求

组件	日志文件数	CPU	内存	磁盘
Filebeat	5个	1核	256MB	当Filebeat已读取日志，但尚未发送至其它组件前，会把日志数据缓存到磁盘中，因此磁盘占用为已读取日志文件总大小，最大值为200MB*日志文件数。
	10个	2核	256MB	
ELK: Logstash	5个	2核	1GB	同Filebeat的磁盘要求。

组件	日志文件数	CPU	内存	磁盘
	10个	4核	2GB	
ELK: ElasticSearch	5个	4核	8GB	ElasticSearch的磁盘空间需求取决于数据量、副本数： <ul style="list-style-type: none">● 一般场景：数据量*(1+副本数)*1.7● 峰值场景：数据量*(1+副本数)*2.6 其中，数据量为存储的日志文件总大小，即200MB*日志文件数；本实践中，ElasticSearch的副本数为1。 ElasticSearch存储的数据，默认未设置过期时间，用户可根据磁盘空间情况，为ElasticSearch数据配置索引生命周期策略。
	10个	8核	16GB	
ELK: Kibana	5个	2核	256MB	Kibana的磁盘空间需求取决于Kibana面板数量、复杂度、更新频率、在ElasticSearch中创建的索引数量及大小，磁盘空间需求较少，约500MB。
	10个	4核	512MB	

6.3 准备 Docker 环境和镜像

本实践使用官方最新稳定版的Docker镜像（基于Docker、Docker-Compose）部署Filebeat和ELK。

前提条件

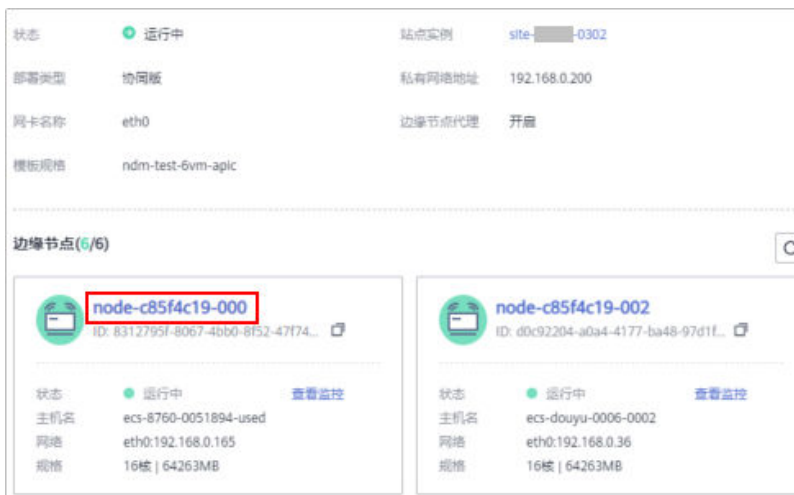
- 准备一台可连接公网的本地PC，系统为Linux，用于拉取Docker镜像。
- 已根据[资源规划](#)，完成服务器资源的准备，且ELK所部署的服务器与APIC的SHUBAO组件所在节点网络互通。

操作步骤

1. 确认SHUBAO组件所在节点。
 - a. 登录ROMA Connect控制台，在左侧导航栏选择“Site实例”。
 - b. 在要监控日志的Site协同版实例上，单击站点名称，打开站点信息弹窗。



- c. 在弹窗中单击边缘节点的名称，进入该边缘节点的详细信息页面。



- d. 在节点信息页面单击“应用”页签，如果存在名称为“shubao-xxxxxx”的容器应用，则该节点为SHUBAO组件所在节点，记录该节点的IP地址。



- e. 重复步骤1.c-1.d，确认并记录所有SHUBAO组件所在节点的IP地址。
2. 安装Docker环境。
 - 参考[安装Docker Engine](#)，为本地PC安装Docker。
 - 参考[安装Docker Engine](#)和[安装Docker Compose](#)为ELK所部署的服务器安装Docker和Docker Compose。

- 参考[安装Docker Compose](#)为APIC的SHUBAO组件所在节点安装Docker Compose。
3. 在本地PC上，切换root用户，并执行以下命令拉取Docker镜像。

```
docker pull docker.elastic.co/beats/filebeat:8.7.1
docker pull docker.elastic.co/logstash/logstash:8.7.1
docker pull docker.elastic.co/elasticsearch/elasticsearch:8.7.1
docker pull docker.elastic.co/kibana/kibana:8.7.1
docker pull python:3.8.17
```
 4. 执行以下命令查询3中拉取的Docker镜像的镜像ID。

```
docker images
```

执行命令后返回如下信息，其中IMAGE ID即为镜像ID。

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
docker.elastic.co/beats/filebeat	8.7.1	*****	*****	*****
docker.elastic.co/logstash/logstash	8.7.1	*****	*****	*****
docker.elastic.co/elasticsearch/elasticsearch	8.7.1	*****	*****	*****
docker.elastic.co/kibana/kibana	8.7.1	*****	*****	*****
python	3.8.17	*****	*****	*****
...				
 5. 修改python镜像。
 - a. 执行以下命令进入python镜像。

```
docker run -it python_image_id bash
```

其中，**python_image_id**为4中查询到的python镜像的镜像ID。
 - b. 执行以下命令在镜像内安装依赖。

```
pip install requests
```
 - c. 执行以下命令检查安装的依赖是否存在。

```
#启动Python交互式输入，若使用的Python 3，则执行的命令为python3
python
import requests
import json
import http
```

如果安装的依赖存在，则无回显；如果安装的依赖不存在，则返回“module not found”，需执行5.b重新安装依赖。
 - d. 执行以下命令退出Python交互式输入和退出镜像。

```
exit()
exit
```
 - e. 执行以下命令查询python镜像所对应的容器ID。

```
docker ps -a
```

执行命令后返回如下回显信息，其中CONTAINER ID即为容器ID，IMAGE为镜像名称或镜像ID。根据4中查询到的python镜像的镜像ID，找到其对应的容器ID。

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
*****	*****	!*****!	*****	*****	*****	*****
*****	*****	!*****!	*****	*****	*****	*****
...						
 - f. 执行以下命令保存镜像。

```
docker commit container_id python
```

其中，**container_id**为5.e中查询到的python镜像的容器ID。
 6. 执行以下命令导出镜像文件。

```
docker save filebeat_image_id > filebeat_image.tar
docker save logstash_image_id > logstash_image.tar
docker save elasticsearch_image_id > elasticsearch_image.tar
docker save kibana_image_id > kibana_image.tar
docker save python_image_id > python_image.tar
```

其中**filebeat_image_id**、**logstash_image_id**、**elasticsearch_image_id**、**kibana_image_id**和**python_image_id**为4中查询到的对应组件的镜像ID。

7. 把导出的镜像文件上传到服务器中。
 - 把镜像文件filebeat_image.tar上传到SHUBAO组件所在节点，步骤1查询到的所有SHUBAO所在的节点都要上传。
 - 把镜像文件logstash_image.tar、elasticsearch_image.tar、kibana_image.tar和python_image.tar上传到ELK所部署的服务器。
8. 登录SHUBAO组件所在节点，切换root用户，执行以下命令导入Docker镜像。步骤1查询到的所有SHUBAO所在的节点都要执行。

假设镜像文件所在路径为/tmp。

```
cd /tmp
docker load < filebeat_image.tar
```

9. 登录ELK所部署的服务器，切换root用户，执行以下命令导入Docker镜像。

假设镜像文件所在目录为/tmp。

```
cd /tmp
docker load < logstash_image.tar
docker load < elasticsearch_image.tar
docker load < kibana_image.tar
docker load < python_image.tar
```

6.4 修改 Filebeat 和 ELK 配置

修改 Filebeat 配置

Filebeat目录和文件结构如下所示：

```
_ filebeat
  _ log
  _ docker-compose.yml
  _ filebeat.yml
```

1. 登录SHUBAO组件所在节点，切换root用户，执行以下命令创建Filebeat相关目录和配置文件。

用户可自定义配置文件所在路径，此处以在根目录下创建filebeat目录为例进行说明。

```
mkdir -p /filebeat/log
touch /filebeat/filebeat.yml
touch /filebeat/docker-compose.yml
```

其中，log目录用于存储Filebeat收集到的日志数据以及其它元数据信息，即使容器Filebeat被删除并重新启动，会继续读取该目录中的内容（如registry），并在收集数据时更新该目录，这样可确保Filebeat只收集新的日志数据，避免将已经发送过的数据再次发送。

2. 执行以下命令为log目录配置权限。
3. 编辑filebeat.yml文件，在文件中增加如下配置。

```
filebeat.inputs:
- type: filestream
  # id需要替换为数字，不同SHUBAO组件所在节点之间相互唯一。例如SHUBAO组件所在节点有3个，则id可分别配置为site_log_filebeat_01，site_log_filebeat_02，site_log_filebeat_03
  id: site_log_filebeat_id
  enabled: true
  paths:
  - /usr/share/filebeat/access_log/access_public.log
  fields:
  # index_name为在ElasticSearch存储和Kibana可视化显示时的指定索引名称，可自定义，若需要区分日志来源，则不同SHUBAO组件所在节点之间配置为不一样。
  es_index_name: index_name
```

```
fields_under_root: true
close_inactive: 1m
clean_inactive: 72h

output.logstash:
# 填写ELK组件所服务器的IP和Logstash对外开放的端口号（在ELK的logstash.conf文件中配置，默认为5044），需自行配置防火墙端口，开放访问权限。
hosts: ["ELK_LOGSTASH_IP:5044"]
```

4. 编辑docker-compose.yml文件，在文件中增加如下配置。

```
version: '3'

services:
  filebeat:
    image: docker.elastic.co/beats/filebeat:8.7.1
    network_mode: host
    restart: always
    container_name: filebeat
    volumes:
      - ./filebeat.yml:/usr/share/filebeat/filebeat.yml

# 如果需要把Filebeat收集到的日志数据保存到本地log目录下，则取消本行代码注释
# - ./log:/usr/share/filebeat/data

# filebeat读取本地日志，该本地日志地址需要根据实际待监控路径的地址进行配置，相对路径或绝对路径均可。
# 多地址监听：若在一台服务器中需要监听多个地址，则需在filebeat.yml的path和docker-compose.yml的volumes中增加新的配置，配置方式与该例子相同。
- /opt/roma/logs/apic-shubao/run/access_public.log:/usr/share/filebeat/access_log/access_public.log
```

5. 重复步骤1-4，登录其他SHUBAO组件所在节点修改Filebeat配置。

修改 ELK 配置

ELK目录和文件结构如下所示：

```
_ ELK
  |_ elasticsearch
  |   |_ elasticsearch.yml
  |_ kibana
  |   |_ kibana.yml
  |_ logstash
  |   |_ log
  |   |_ logstash.conf
  |   |_ logstash.yml
  |_ python_elk
  |   |_ send_msg.py
  |_ docker-compose.yml
```

1. 登录ELK所部署的服务器，切换root用户，执行以下命令创建ELK相关目录和配置文件。

用户可自定义配置文件所在路径，此处以在根目录下创建ELK目录为例进行说明。

```
mkdir -p /ELK/logstash/log /ELK/elasticsearch /ELK/kibana /ELK/python_elk
touch /ELK/logstash/logstash.yml
touch /ELK/logstash/logstash.conf
touch /ELK/elasticsearch/elasticsearch.yml
touch /ELK/kibana/kibana.yml
touch /ELK/python_elk/send_msg.py
touch /ELK/docker-compose.yml
```

其中，log目录用于挂载Logstash的数据目录，存储Logstash处理后的数据和元数据信息，即使容器Logstash被删除，日志数据也不会丢失，并且可以轻松的备份和恢复数据。

2. 执行以下命令为log目录配置权限。
chmod 777 /ELK/logstash/log
3. 编辑logstash.conf文件，在文件中增加如下配置。

```
input{
  beats {
    port => 5044
  }
}
# 过滤解析规则
filter {
  grok {
    match => { "message" => "%{DATA:my_remote_addr} %{DATA:request_id} %{DATA:api_id} %
{DATA:user_name} %{DATA:app_id} \[%{HTTPDATE:timestamp}\] %{NUMBER:request_time:float} %
{DATA:request_method} %{URIPROTO:scheme}://%{DATA:host} %{DATA:router_uri} %
{DATA:server_protocol} %{NUMBER:status:int} %{NUMBER:bytes_sent:int} %
{NUMBER:request_length:int} \"%{DATA:http_user_agent}\" \"%{DATA:http_x_forwarded_for}\" \"%
{DATA:upstream_addr}\" \"%{DATA:upstream_uri}\" \"%{DATA:upstream_status}\" \"%
{DATA:upstream_connect_time}\" \"%{DATA:upstream_header_time}\" \"%
{DATA:upstream_response_time}\" \"%{WORD}-%{WORD}-%{INT} %
{DATA:all_upstream_response_time} %{DATA:errorType} %{DATA:auth_type} %{DATA:access_model1}
%{DATA:access_model2} %{DATA:inner_time} %{DATA:proxy_protocol_vni} %
{DATA:proxy_protocol_vpce_id} \"%{DATA:proxy_protocol_addr}\" %{NUMBER:body_bytes_sent} %
{DATA:api_name} %{DATA:app_name} %{DATA:provider_app_id} %{DATA:provider_app_name} \"%
{DATA:custom_data_log1}\" \"%{DATA:custom_data_log2}\" \"%{DATA:custom_data_log3}\" \"%
{DATA:custom_data_log4}\" \"%{DATA:custom_data_log5}\" \"%{DATA:custom_data_log6}\" \"%
{DATA:custom_data_log7}\" \"%{DATA:custom_data_log8}\" \"%{DATA:custom_data_log9}\" \"%
{DATA:custom_data_log10}\" \"%{WORD:response_source}?" }
}
}

date {
  match => [ "timestamp", "dd/MMM/yyyy:HH:mm:ss Z" ]
  target => "@timestamp"
}

if [upstream_response_time] =~ /^d+(\.d+)?$/ {
  mutate {
    add_field => { "duration" => "%{upstream_response_time}" }
    convert => { "upstream_response_time" => "float" }
  }
}
}

output{
  elasticsearch {
    # ElasticSearch对外开放的端口默认为9200，如果修改了端口，此处需同步修改
    hosts => ["http://localhost:9200"]
    index => "site_access_log_%{es_index_name}_%{+YYYY-MM-dd}"

    # username和密码自定义，需与部署ELK时设置的用户名和密码保持一致
    user => "username"
    password => "password"
  }

  stdout {
    codec => rubydebug
  }

  # 当状态异常或调用超时，触发异常告警通知
  if ([status] == 504 or [status] == 502 or [status] == 404) or ([duration] and
[upstream_response_time] >= 60.0) {
    # http信息可根据实际需要修改
    http {
      retry_failed => "false"
      http_method => "post"
      # 配置告警信息监听地址，端口默认为9202
      url => "http://localhost:9202"
      format => "message"
      content_type => "application/x-www-form-urlencoded"
      # 自定义告警信息内容，可根据实际需要修改
      message => "ip: %{my_remote_addr}, 告警时间: %{timestamp}, 接口: %{router_uri}, 状态码: %
{status}, 调用时长: %{upstream_response_time}"
    }
  }
}
```



```
}  
}
```

4. 编辑logstash.yml文件，在文件中增加如下配置。

```
path.config: /usr/share/logstash/pipeline/*.conf  
xpack.monitoring.enabled: true  
# ElasticSearch对外开放的端口默认为9200，如果修改了端口，此处需同步修改  
xpack.monitoring.elasticsearch.hosts: ["http://localhost:9200"]
```

5. 编辑elasticsearch.yml文件，在文件中增加如下配置。

```
cluster.name: "elasticsearch"  
network.host: "127.0.0.1"  
http.cors.enabled: true  
http.cors.allow-origin: "*"   
discovery.type: "single-node"  
xpack.security.enabled: true
```

6. 编辑kibana.yml文件，在文件中增加如下配置。

```
# 0.0.0.0表示所有客户端地址均可通过用户名及密码访问Kibana可视化界面，如果仅允许指定客户端IP访问，则修改为指定的客户端IP  
server.host: "0.0.0.0"  
# Kibana端口号，默认为5601，如果需要修改端口号，则取消本行代码注释，并修改实际使用的端口号  
# server.port: 5601  
# ElasticSearch对外开放的端口默认为9200，如果修改了端口，此处需同步修改  
elasticsearch.hosts: [ "http://localhost:9200" ]  
monitoring.ui.container.elasticsearch.enabled: true  
# username和password自定义，需与部署ELK时设置的用户名和密码保持一致  
elasticsearch.username: 'username'  
elasticsearch.password: 'password'
```

7. 编辑docker-compose.yml文件，在文件中增加如下配置。

```
version: '3'  
  
services:  
  logstash:  
    image: docker.elastic.co/logstash/logstash:8.7.1  
    restart: always  
    container_name: logstash  
    network_mode: host  
  
    volumes:  
    - ./logstash/logstash.conf:/usr/share/logstash/pipeline/logstash.conf  
    - ./logstash/logstash.yml:/usr/share/logstash/config/logstash.yml  
  
    # 如果需要把Logstash处理后的日志数据保存到本地log目录下，则取消本行代码注释  
    # - ./logstash/log:/usr/share/logstash/data/:rw  
  
  elasticsearch:  
    image: docker.elastic.co/elasticsearch/elasticsearch:8.7.1  
    restart: always  
    container_name: elasticsearch  
    # 如果ElasticSearch占用资源过高，导致容器无法成功启动，可取消此处的代码注释，根据服务器情况  
    # 对资源使用进行限制  
    # deploy:  
    # resources:  
    #   limits:  
    #     memory: 4g  
    #     cpus: '0.8'  
    network_mode: host  
    volumes:  
    - ./elasticsearch/elasticsearch.yml:/usr/share/elasticsearch/config/elasticsearch.yml  
  
  kibana:  
    image: docker.elastic.co/kibana/kibana:8.7.1  
    restart: always  
    container_name: kibana  
    network_mode: host  
    volumes:  
    - ./kibana/kibana.yml:/usr/share/kibana/config/kibana.yml  
  
python_elk:
```

```
# python_image_id为python镜像的镜像ID
image: python_image_id
restart: always
container_name: python_elk
network_mode: host
volumes:
- ./python_elk/send_msg.py:/send_msg.py
build: .
command: python3 -u /send_msg.py
```

8. 编辑send_msg.py文件，在文件中增加如下配置。

以告警通知发送到一个API为例进行说明，且该API使用Token鉴权。

```
from http.server import BaseHTTPRequestHandler, HTTPServer
import json
import requests
import os

class RequestHandler(BaseHTTPRequestHandler):
    def do_POST(self):
        content_length = int(self.headers['Content-Length'])
        content = self.rfile.read(content_length).decode('utf-8')
        print("\n\n\n\nPath: {}\nHeaders: {}\nBody: {}".format(str(self.path), str(self.headers),
        content))

        # url_address为获取Token的URL地址，appid和appsecret为获取Token所需的鉴权信息。
        token_get_url = "url_address"
        params = {
            "appid": "appid",
            "appsecret": "appsecret"
        }
        response = requests.get(token_get_url, params=params)

        if response.status_code == 200:
            response_json = json.loads(response.content.decode('utf-8'))
            access_token = response_json['d']['access_token']

            # 配置要接收告警通知的API信息，msg_send_url为API的URL地址，identity、ucs_type和wid为
            # API的相关请求参数，可根据API实际情况增删和修改参数
            msg_send_url = "msg_send_url"
            identity = "identity"
            ucs_type = "ucs_type"
            wid = "wid"
            payload =
            "access_token={}&numbers[0]={}&ucs_type={}&content[msgtype]=text&content[text]
            [content]={}&wid={}".format(access_token, identity, ucs_type, content, wid)

            print("content: {}".format(content))
            print("payload: {}".format(payload))
            ans = os.system('curl {} -d {}'.format(msg_send_url, payload))
            print("os.system: ", ans)

# 告警信息监听端口默认为9202，如果修改了端口，此处需同步修改
def run_server(port=9202):
    with HTTPServer(("localhost", port), RequestHandler) as httpd:
        print("Server listening on port {}...", port)
        httpd.serve_forever()

if __name__ == "__main__":
    run_server()
```

6.5 部署 Filebeat 和 ELK

部署 Filebeat

1. 登录SHUBAO组件所在节点，切换root用户。

2. 执行以下命令对filebeat.yml文件进行授权。

```
cd /filebeat
sudo chown root filebeat.yml
```

3. 执行以下命令启动Filebeat。

```
docker-compose up -d
```

部署 ELK

1. 登录ELK所部署的服务器，切换root用户。

2. 执行以下命令启动ELK。

```
cd /ELK
docker-compose up -d
```

3. 执行以下命令，判断ElasticSearch是否启动成功。

```
curl localhost:9200
```

其中9200为[修改ELK配置](#)中设置的ElasticSearch对外开放端口号。

- 如果启动成功，会返回鉴权失败信息“missing authentication credentials for REST request”。
- 如果启动失败，会返回连接拒绝信息“connection refused”。

4. 为ElasticSearch设置密码校验。

- a. ElasticSearch成功启动后，执行以下命令进入ElasticSearch容器内部。

```
docker exec -it elasticsearch bash
```

进入容器内部后处于/usr/share/elasticsearch路径下。

- b. 执行以下命令为ElasticSearch生成节点证书。

```
bin/elasticsearch-certutil ca -out config/elastic-certificates.p12 pass ""
```

若要求输入密码，则输入与[修改ELK配置](#)时设置的密码一致。

- c. 执行以下命令新增用户。

```
bin/elasticsearch-users useradd username
```

其中，**username**设置为与[修改ELK配置](#)时设置的用户名一致。

在新增用户后，会自动提示设置密码，密码设置为与[修改ELK配置](#)时设置的密码一致。

- d. 执行以下命令为新增的用户授权，用于Logstash和Kibana访问ElasticSearch。

```
bin/elasticsearch-users roles -a superuser username
bin/elasticsearch-users roles -a kibana_system username
```

其中，**username**为步骤4.c新增的用户。

- e. 执行以下命令查看用户授权。

```
bin/elasticsearch-users roles -v username
```

其中，**username**为步骤4.c新增的用户。

- 如果授权正确，则返回如下信息。

```
username: kibana_system,superuser
```

- 如果授权错误，则执行以下命令删除授权，并重新为该用户授权。

```
bin/elasticsearch-users roles -r auth_name username
```

其中，**auth_name**为查询到授权错误的权限名称。

- f. 执行以下命令退出ElasticSearch容器。

```
exit
```

- g. 执行以下命令验证ElasticSearch用户是否配置正确。

```
curl -X GET "http://localhost:9200" -u username
```

其中，**username**为步骤4.c新增的用户。执行命令后，按提示输入步骤4.c设置的密码。

- 如果配置成功，则返回如下信息。

```
{
  "name" : "xxx",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "xxx",
  "version" : {
    "number" : "8.7.1",
    "build_flavor" : "default",
    "build_type" : "docker",
    "build_hash" : "xxx",
    "build_date" : "xxx",
    "build_snapshot" : "xxx",
    "lucene_version" : "xxx",
    "minimum_wire_compatibility_version" : "xxx",
    "minimum_index_compatibility_version" : "xxx"
  },
  "tagline" : "You Know, for Search"
}
```

- 如果配置失败，则返回如下错误信息。

```
{"error":{"root_cause":[{"type":"security_exception","reason":"missing authentication credentials for REST request [/]","header":{"WWW-Authenticate":["Basic realm=\"security\" charset=\"UTF-8\"","ApiKey"]}], "type":"security_exception","reason":"missing authentication credentials for REST request [/]","header":{"WWW-Authenticate":["Basic realm=\"security\" charset=\"UTF-8\"","ApiKey"]},"status":401}
```


5. 执行以下命令修改logstash.conf文件中**user**和**password**的值，修改为与步骤4.c中设置的用户名密码一致。
vim /ELK/logstash/logstash.conf
6. 执行以下命令修改kibana.yml文件中**elasticsearch.username**和**elasticsearch.password**的值，修改为与步骤4.c中设置的用户名密码一致。
vim /ELK/kibana/kibana.yml
7. 执行以下命令重启Logstash和Kibana。
docker restart logstash
docker restart kibana

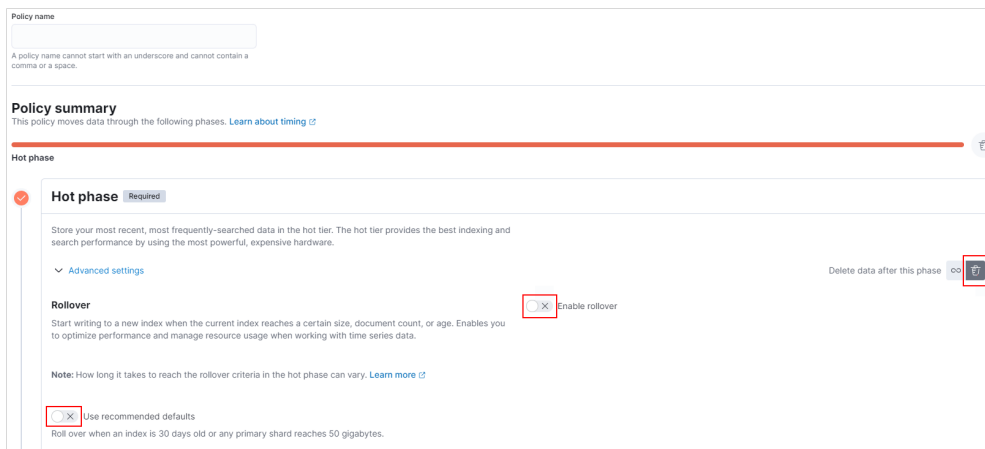
6.6 配置索引生命周期策略（可选）

ElasticSearch存储的数据，默认未设置过期时间，用户可根据磁盘空间情况，为ElasticSearch数据设置索引生命周期策略。

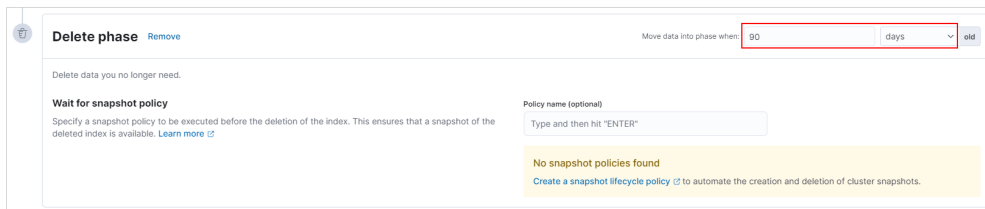
创建生命周期策略

1. 登录Kibana可视化界面。
 - 访问地址：**IP:Port**
其中，**IP**为ELK所部署服务器的IP；**Port**为Kibana端口号，默认为5601，也可以在**修改ELK配置**时自定义端口号。如果网络中部署了防火墙，需提前配置防火墙的端口访问权限。
 - 用户名密码：在**部署ELK**时自定义的用户名和密码。
2. 在页面左上角单击“☰”，然后选择“Management > Stack Management > Index Lifecycle Policies”。
3. 在页面中单击“Create policy”，创建生命周期策略。
4. 在创建策略页面配置策略相关信息。
 - Policy name：自定义策略名称。

- 在“Hot phase”处单击“Advanced Setting”，在右侧选择“”，并关闭“Use recommended defaults”和“Enable rollover”。




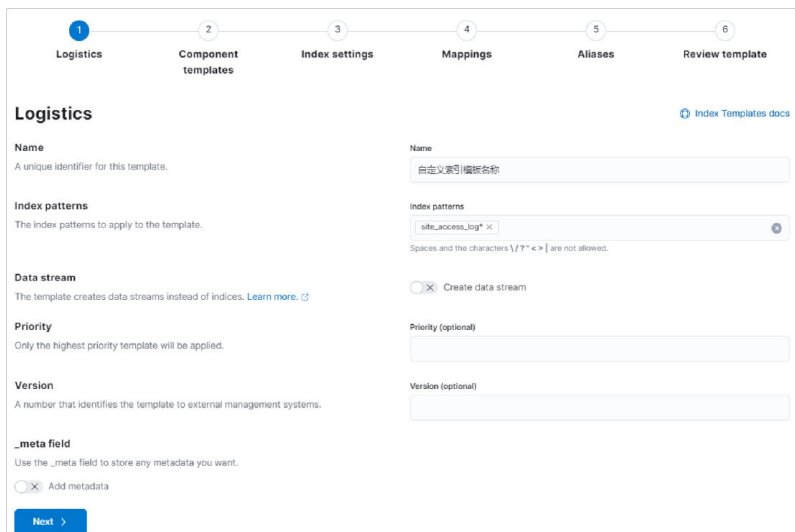
- 在“Delete phase”处设置数据的过期时间“Move data into phase when”。



5. 单击“Save policy”保存策略。

创建索引模板

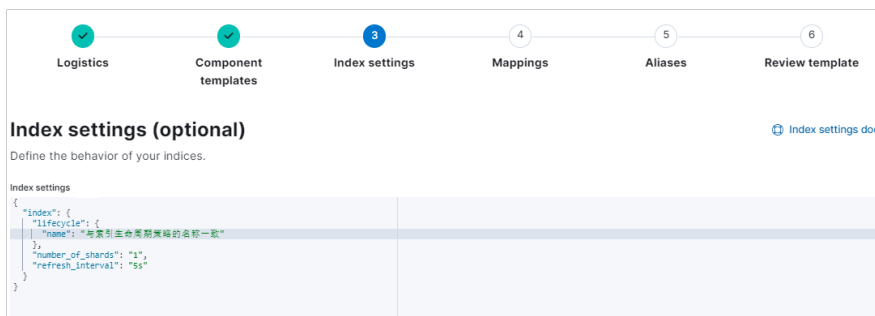
1. 在页面左上角单击“”，然后选择“Management > Stack Management > Index Management”。
2. 在页面中选择“Index Templates”页签，然后单击“Create template”，创建索引模板。
3. 在创建模板页面配置模板相关信息。
 - a. 配置Logistics，完成后单击“Next”。
 - Name：模板名称，自定义。
 - Index pattern：使用通配符形式匹配多个日志索引，例如“site_access_log_*”，表示匹配以site_access_log_开头的所有日志索引文件。
日志索引的名称格式为site_access_log_{es_index_name}_{current_date (YYYY-MM-dd)}，在编辑logstash.conf文件时设置。



- b. Component配置保持默认，直接单击“Next”。
- c. 配置Index settings，完成后单击“Next”。

在Index settings中配置模板关联的生命周期策略，其中“name”的值与创建索引模板中设置的“Policy name”一致。

```
{
  "index": {
    "lifecycle": {
      "name": "与索引生命周期策略的名称一致"
    },
    "number_of_shards": "1",
    "refresh_interval": "5s"
  }
}
```



- d. Mappings和Aliases配置保持默认，直接单击“Next”。
- e. 在Review template中确认模板的配置信息，然后单击“Create template”，完成索引模板的创建。

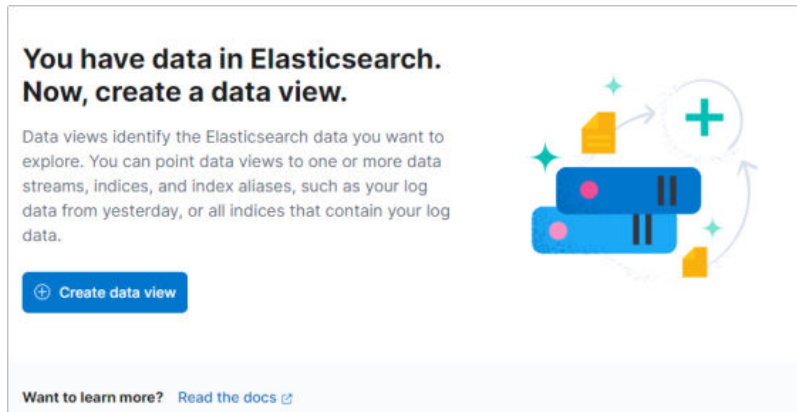
6.7 使用 Kibana 查看日志

1. 登录Kibana可视化界面。
 - 访问地址：IP:Port

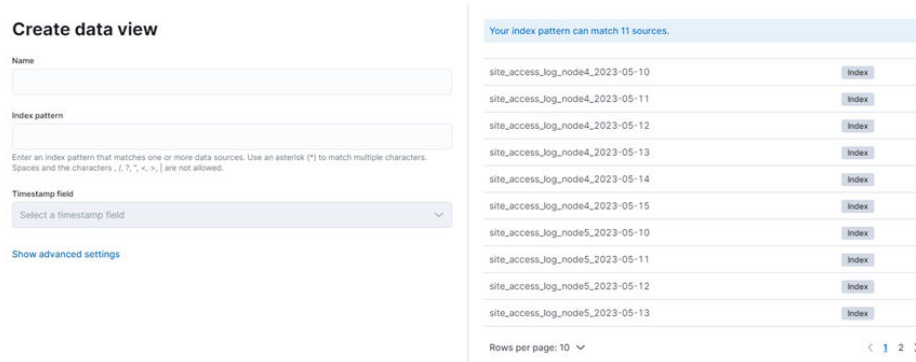
其中，IP为ELK所部署服务器的IP；Port为Kibana端口号，默认为5601，也可以在修改ELK配置时自定义端口号。如果网络中部署了防火墙，需提前配置防火墙的端口访问权限。

- 用户名密码：在部署ELK时自定义的用户名和密码。

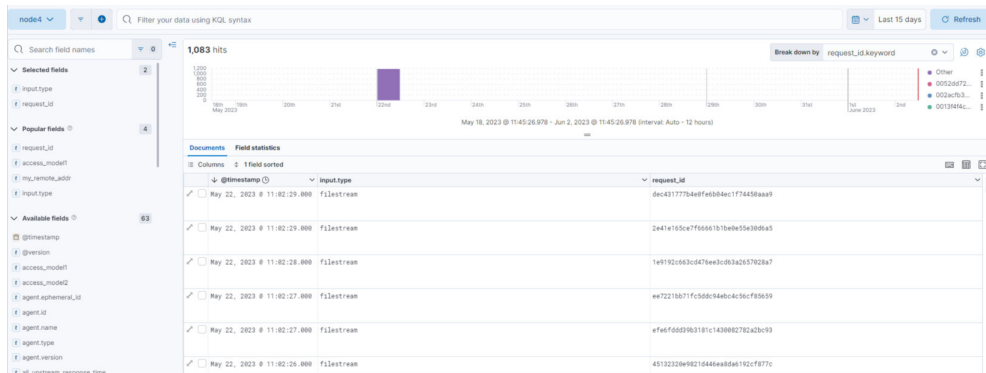
- 在页面左上角单击“☰”，然后选择“Analytics > Discover”。
- 在页面中单击“Create data view”，创建数据视图。



- 在创建数据视图页面中，配置视图信息，完成后单击“Save data view to Kibana”保存视图。



- Name: 数据视图名称，自定义。
 - Index pattern: 使用通配符形式匹配多个日志索引，例如“site_access_log_*”，表示匹配以site_access_log_开头的所有日志索引文件。
日志索引的名称格式为site_access_log_{es_index_name}_{current_date (YYYY-MM-dd)}，在编辑logstash.conf文件时设置。
- 在数据视图中查看日志数据和日志统计。
 - 在页面右上角设置查询时间范围，然后单击“Refresh”，可查看该时间范围内的日志数据。
 - 在页面左侧的字段列表中选择要查看的日志字段，筛选包含该字段的日志数据。
 - 在页面右侧的“Break down by”处选择日志字段，可查看该字段的聚合统计数据。



7 使用自定义后端开发自定义认证

[方案概述](#)
[开发自定义认证](#)
[使用自定义认证](#)

7.1 方案概述

应用场景

使用ROMA Connect的服务集成APIC，可以快速将已有后端服务、数据源和自定义函数封装成标准的RESTful API并对外开放，简化了业务系统间的对接过程，降低了对接成本。

APIC提供了APP认证、IAM认证等安全认证方式，确保API访问的安全。APIC还提供了自定义认证方式，方便用户使用自己的认证机制对API进行安全认证，以更好的兼容已有业务能力。

方案优势

使用自定义认证方式，业务系统可继承已有的认证机制，可以更好的兼容已有业务能力，降低开发成本。

约束与限制

APIC的自定义认证支持前端自定义认证和后端自定义认证，本实践中以前端自定义认证为例进行说明，后端自定义认证的使用请参见[创建后端自定义认证](#)。

7.2 开发自定义认证

创建用于前端认证的函数后端

通过自定义后端开发自定义认证函数，然后调用该函数进行访问认证，实现自定义认证。

1. 登录ROMA Connect控制台，在“实例”页面单击实例上的“查看控制台”，进入实例控制台。
2. 创建集成应用。
 - a. 在左侧的导航栏选择“集成应用”，单击页面右上角的“创建集成应用”。
 - b. 在创建集成应用弹窗中填写集成应用的“名称”，然后单击“确认”。
3. 在左侧的导航栏选择“服务集成 APIC > 自定义后端”，在“后端列表”页签中单击“创建后端”。
4. 在创建后端页面配置后端信息，完成后单击“立即创建”。

图 7-1 自定义后端配置

* 后端名称
支持汉字、英文、数字、中划线、下划线、点、斜杠、中英文格式下的小括号和冒号。

* 集成应用 [新建集成应用](#)

* 后端请求方法

* 后端请求路径
支持英文、数字、中划线、下划线、点等，且以斜杠 (/) 开头。2-256个字符

后端安全认证 签名密钥 无认证
无认证模式，安全级别低，所有用户均可访问，不推荐使用。

描述
0/1,000

表 7-1 后端配置

参数	说明
后端名称	填写后端的名称，自定义。
集成应用	选择2创建的集成应用。
后端请求方法	后端的请求方法必须选择“POST”。
后端请求路径	填写后端的请求路径，格式如：/getUserInfo/userId。
后端安全认证	选择后端的安全认证方式，此处选择“无认证”。
描述	填写后端的描述信息。
高级设置	高级设置保持默认设置。

后端创建完成后，页面自动跳转到该后端的编辑器页面，后端类型默认为数据后端。

5. 开发自定义认证函数的功能实现。

在编辑器的左上角单击“文件 > 新建函数后端 > 前端自定义认证示例”，在弹窗中单击“确定”，然后编写用于安全认证的函数脚本。

修改示例中的认证参数名称和认证参数值，此处使用API请求的Header参数“x-auth”作为认证参数，当“x-auth”的值为“user1:xxxx”时将认证通过。

```
function execute(data) {
  data = JSON.parse(data)
  body = data.body
  if (body["headers"]["x-auth"] === 'user1:xxxx') {
    result = {
      "status": "allow",
      "context": {
        "user_name": "user1"
      }
    }
  }
  else {
    result = {
      "status": "deny",
      "context": {
        "error_code": "1001",
        "error_message": "incorrect token input"
      }
    }
  }
  return result
}
```

6. 完成函数编辑后，在页面右上角单击“保存”，保存函数后端配置。

7. 在页面右上角单击“部署”，在确认弹窗中单击“确定”，部署函数后端。

创建前端自定义认证

1. 在实例控制台左侧的导航栏选择“服务集成 APIC > API策略”，在“自定义认证”页签下单击“创建自定义认证”。
2. 在创建自定义认证弹窗中配置前端自定义认证信息。

图 7-2 自定义认证配置

表 7-2 前端自定义认证配置

参数	说明
认证名称	填写自定义认证的名称，自定义。
集成应用	选择2创建的集成应用。
类型	创建前端自定义认证时，选择“前端”。
函数地址	选择创建用于前端认证的函数后端中创建的函数后端。
缓存时间(秒)	保持默认设置。
身份来源	保持默认设置。
是否发送body	保持默认设置。
用户数据	保持默认设置。

3. 完成后单击“确定”。

7.3 使用自定义认证

创建自定义认证的 API

在ROMA Connect中创建一个使用自定义认证的API。

1. 在实例控制台的左侧导航栏选择“服务集成 APIC > API列表”，在页面右上角单击“创建API”。
2. 在创建API页面配置API的前端请求信息。

图 7-3 前端请求信息

前端定义

* API名称: API_test
支持汉字、英文、数字、中划线、下划线、点、斜杠、中英文格式下的小括号和冒号、中文格式下的顿号，且只能以英文、汉字和数字开头，3-255个字符。

* 集成应用: app-test C 新建集成应用

* 所属分组: Group-test C 新建分组

* URL: 请求方法: GET, 请求协议: HTTPS, 子域名: 774d8b..., 路径: /test

匹配模式: 绝对匹配 (选中), 前缀匹配

调用请求Path固定为创建时填写的API请求Path。

标签: 请选择或输入标签名

描述: 输入对API的描述

0/255

表 7-3 前端请求信息

参数	说明
API名称	填写API的名称，自定义。
集成应用	选择 创建用于前端认证的函数后端 中创建的集成应用。
所属分组	选择前端API所属的API分组。若没有可用的API分组，可单击右侧的“创建分组”，创建一个API分组。
URL	配置API的URL。 <ul style="list-style-type: none">● 请求方法：选择API的请求方法，此处选择“GET”。● 请求协议：选择API使用的请求协议，此处选择使用“HTTPS”。● 路径：填写API的请求路径，格式如：/getUserInfo/{userId}，此处以“/test”为例。
匹配模式	选择API请求路径的匹配模式，此处选择“绝对匹配”。
标签	保持默认设置。
描述	保持默认设置。

3. 配置API的安全配置信息。

图 7-4 安全配置信息

安全配置

类型 ? 公开 私有

安全认证 APP认证 华为IAM认证 自定义认证 无认证

使用自定义认证来控制API的访问。安全级别:

自定义认证 [新建自定义认证](#)

支持跨域CORS 开启支持跨域CORS

表 7-4 安全配置信息

参数	说明
类型	选择是否允许API上架到云市场，“公开”为表示允许。本配置样例中不涉及上架操作，此处保持默认设置即可。
安全认证	安全认证方式选择“自定义认证”。
自定义认证	选择 创建前端自定义认证 中创建的自定义认证。
支持跨域CORS	保持默认设置，不开启。

- 配置API的请求参数。为简单起见，此处不定义请求参数，直接单击“下一步”。
- 选择“后端服务类型”，为简单起见，本配置样例中选择“Mock”类型后端服务，固定返回指定响应结果。
- 配置API的后端配置信息。

图 7-5 后端配置信息

后端基础定义

Mock自定义返回码

Mock返回结果

10/2,048

Mock一般用于开发调试验证。在项目初始阶段，后端服务没有搭建好API联调环境，可以使用Mock模式，将预期结果固定返回给API调用方，方便调用方进行项目开发。

后端认证 使用自定义安全认证

常量参数名	参数值
-------	-----

[+ 添加header参数](#)

表 7-5 后端配置信息

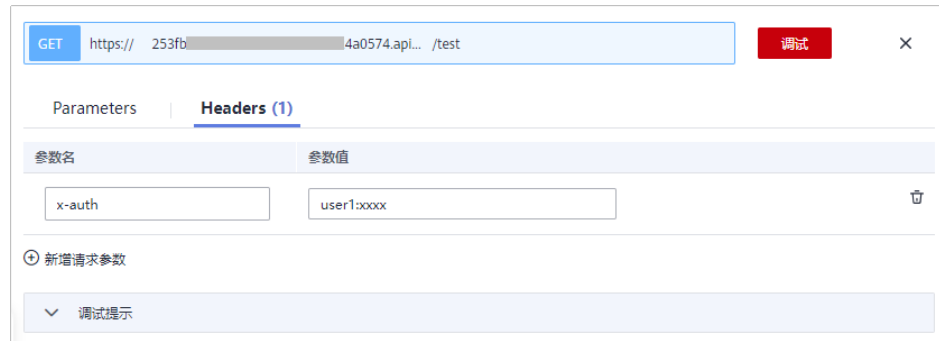
参数	配置说明
Mock自定义返回码	选择API响应的HTTP状态码，此处选择“200”。
Mock返回结果	填写API的响应结果，此处填写为“ROMA TEST!”，即用户在调用该API时，API固定返回“ROMA TEST!”作为响应结果。
后端认证	保持默认设置，不开启。
添加header参数	保持默认设置。

7. 配置返回结果的响应示例，为简单起见，此处保持默认设置。
8. 完成后单击“完成”。
API创建成功后，自动跳转到API运行页面，可查看API的详细信息。

调试 API

1. 在API运行页面左侧选择**创建自定义认证的API**中创建的API，然后在页面右上角单击“调试”。
2. 在调试界面中选择“Headers”参数页签，然后单击“新增请求参数”。
3. 在新增的请求参数中，“参数名”填写“x-auth”，“参数值”填写“user1:xxxx”。

图 7-6 API 调试



4. 单击URL右侧的“调试”，页面下方显示调用API发送的请求信息和接收的响应结果信息。

此时，响应结果中显示调用API成功的响应信息“ROMA TEST!”。

```
HTTP/1.1 200 OK
Transfer-Encoding: chunked
Connection: keep-alive
Content-Type: application/json
...
```

```
ROMA TEST!
```

5. 删除3中增加的请求参数“x-auth”，然后再次单击“调试”。

此时，响应结果中显示调用API失败的响应信息，报错信息为鉴权信息错误。

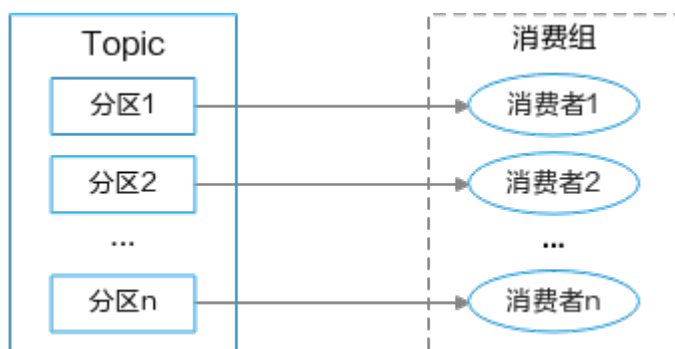
```
HTTP/1.1 401 Unauthorized
Transfer-Encoding: chunked
Connection: keep-alive
Content-Type: application/json
...
```

```
{"error_msg": "Incorrect authentication information: frontend authorizer", "error_code": "APIC.0305", "request_id": "56b2c6ae4a49f44b88670201eade9e05"}
```


8 MQS 消息堆积最佳实践

方案概述

Kafka将Topic划分为多个分区，消息被分布式存储在分区中。同一个消费组内，一个消费者可同时消费多个分区，但一个分区在同一时刻只能被一个消费者消费。



在消息处理过程中，如果客户端的消费速度跟不上服务端的发送速度，未处理的消息会越来越多，这部分消息就被称为堆积消息。消息没有被及时消费就会产生消息堆积，从而会造成消息消费延迟。

消息堆积原因

导致消息堆积的常见原因如下：

- 生产者短时间内生产大量消息到Topic，消费者无法及时消费。
- 消费者的消费能力不足（消费者并发低、消息处理时间长），导致消费效率低于生产效率。
- 消费者异常（如消费者故障、消费者网络异常等）导致无法消费消息。
- Topic分区设置不合理，或新增分区无消费者消费。
- Topic频繁重平衡导致消费效率降低。

实施步骤

从消息堆积的原因看，消息堆积问题可以从消费者端、生产者端和服务端三个方面进行处理。

- 消费者端

- 根据实际业务需求，合理增加消费者个数（消费并发度），建议消费者数量和分区数保持一致。
- 提高消费者的消费速度，通过优化消费者处理逻辑（减少复杂计算、第三方接口调用和读库操作），减少消费时间。
- 增加消费者每次拉取消息的数量：拉取数据/处理时间 \geq 生产速度。

- **生产者端**

生产消息时，给消息Key加随机后缀，使消息均衡分布到不同分区上。

 **说明**

在实际业务场景中，为消息Key加随机后缀，会导致消息全局不保序，需根据实际业务判断是否适合给消息Key加随机后缀。

- **服务端**

- 合理设置Topic的分区数，在不影响业务处理效率的情况下，调大Topic的分区数量。
- 当服务端出现消息堆积时，对生产者进行熔断，或将生产者的消息转发到其他Topic。

9 定时同步 MySQL 中的数据到 Oracle

[方案概述](#)

[资源规划](#)

[创建MySQL连接器](#)

[创建Oracle连接器](#)

[选择模板创建组合应用](#)

[数据同步验证](#)

9.1 方案概述

为了确保数据的实时性和一致性，可以周期性地将MySQL数据库中的数据同步至Oracle数据库中。

这一过程会按照预设的时间间隔精确执行，通过模板“定时同步MySQL中的数据到Oracle”创建组合应用，系统会自动抓取MySQL数据库中的最新数据记录，并将其无缝对接到Oracle数据库中，实现两个数据库之间的高效、准确的数据更新与同步。

该方案不仅能够减少人工操作带来的错误和遗漏，Oracle数据库中的数据都能反映出MySQL数据库的最新状态，满足业务系统的数据需求和数据分析的一致性要求。

9.2 资源规划

完成本实践所需的资源：

表 9-1 资源规划

资源	资源说明	数量
MySQL	2核 4GB	1
Oracle	2核 4GB	1
新版ROMA Connect实例	配额设置为组合应用，2RCU	1

9.3 创建 MySQL 连接器

1. 登录[新版ROMA Connect控制台](#)。
2. 在左侧导航栏选择“连接器”，在连接器页面单击“新建连接”。
3. 选择“MySQL”连接器。
4. 在弹窗中配置连接器信息，完成后单击“测试连接”。

参数	说明
连接名称	填写连接器实例名称MySQL_test。
区域	选择区域。
项目	选择项目。
实例	选择实例，用于后续验证连通性。
连接与安全	选择数据库的连接模式。当前支持“默认”模式，由系统根据用户配置自动拼接数据源连接字符串。
主机IP地址	填写数据库的连接IP地址。
端口	填写数据库的连接端口号。
数据库名	填写要连接的数据库名。
用户名	连接数据库的用户名。
密码	连接数据库的用户密码。
描述	填写连接器的描述信息。

5. 连接成功后，单击“确定”，保存连接器。

9.4 创建 Oracle 连接器

1. 登录[新版ROMA Connect控制台](#)。
2. 在左侧导航栏选择“连接器”，在连接器页面单击“新建连接”。
3. 选择“Oracle”连接器。
4. 在弹窗中配置连接器信息，完成后单击“测试连接”。

参数	说明
连接名称	填写连接器实例名称Oracle_test。
区域	选择区域。
项目	选择项目。
实例	选择实例，用于后续验证连通性。

参数	说明
连接与安全	选择数据库的连接模式。 本实践选择默认模式。 <ul style="list-style-type: none"> 默认：由系统根据用户配置自动拼接数据源连接字符串。 专业：由用户自己输入数据源连接字符串。
主机IP地址	填写数据库的连接IP地址。
端口	填写数据库的连接端口号。
数据库名	填写要连接的数据库名。
pdb数据库名	填写要连接的pdb数据库名。
编码格式	填写数据库的编码格式。
超时时间	填写连接数据库的超时时间。
用户名	连接数据库的用户名。
密码	连接数据库的用户密码。
描述	填写连接器的描述信息，用于识别不同的连接器。

5. 连接成功后，单击“确定”，保存连接器。

9.5 选择模板创建组合应用

1. 登录新版ROMA Connect控制台。
2. 在左侧导航栏选择“资产目录”，在右侧搜索框搜索“定时同步”，选择“定时同步MySQL中的数据到Oracle”模板，单击“使用模板”。

📖 说明

本实践中MySQL数据源表中的字段与Oracle中数据目标表中的字段类型保持一致。
将MySQL数据表中的aa、bb字段同步到Oracle数据表中的TEST_INTEGER、TEST_VARCHAR字段，需要提前在数据库中创建好对应的数据表和字段。

图 9-1 定时同步 MySQL 中的数据到 Oracle



3. 在组合应用编排页面，对定时器、MySQL和Oracle连接器进行参数配置。
 - a. 配置定时器参数。
 - i. 单位：分钟。
 - ii. 间隔：1。表示每隔1分钟执行一次。

- iii. 在这些秒：0。表示在每分钟的0秒开始执行。
- iv. 时区：GMT
- v. 任务模式：并行。
- vi. 立即执行一次：否。

图 9-2 定时器配置

定时器

节点ID: 415940

定时器 - 固定间隔触发 (指定时间)

✔ 选择连接器 > ✔ 选择动作 > ✔ 配置参数

单位

秒 分钟 小时 天 月

* 间隔

1

* 在这些秒

0

* 时区

GMT

任务模式

并行 串行

立即执行一次

否 是

- b. 配置MySQL参数。
 - i. 选择连接，选择之前已创建的MySQL连接器“MySql_test”。
 - ii. 选择动作，选择模板已选择好的“获取记录”。
 - iii. 配置参数。选择需要同步的表名，字段名选择“aa”和“bb”，其他参数保持默认。

图 9-3 MySQL 配置



- c. 配置循环处理器，保持默认配置即可。
- d. 配置Oracle参数。
 - i. 选择连接，选择之前已创建的Oracle连接器“Oracle_test”。
 - ii. 选择动作，选择模板已选择好的“添加记录”。
 - iii. 配置参数。输入目标端的表名，插入数据选择字段名TEST_INTEGER、TEST_VARCHAR，值配置为\${500885|payload.aa}、\${500885|

payload.bb}], 其中500885为循环处理器的节点ID值, 其他参数保持默认。

图 9-4 配置 Oracle



4. 完成后单击右上角的“保存”。
5. 在新建组合任务弹窗中填写组合应用的名称和描述, 然后单击“确定”, 完成组合应用的创建。
6. 单击右上角的“启动”。
7. 在弹窗中选择区域、项目, 并选择已创建好的ROMA Connect实例, 单击“确认”, 启动该组合应用。

9.6 数据同步验证

1. 连接Mysql数据源, 在源端数据表中插入一条数据。
2. 连接Oracle数据源, 根据定时器设置的同步时间间隔等待一分钟, 在目标端数据表中查看源端插入的数据是否已同步到Oracle。