

# 04 Log\_Service详细设计（含Log上传）

- 1 文档介绍 Introduction
  - 1.1 范围说明 Scope
  - 1.2 参考文档 References
  - 1.3 读者对象 Target Audience
  - 1.4. 术语和缩写 Definitions and Abbreviations
- 2 架构设计 Architecture Design
- 3 静态设计 Static Design
  - 3.1 详细模块/组件框图 Detailed Module/Component Diagram （颗粒度L3~L4级）
  - 3.2 模块详细设计 Module Detailed Design （L4级别）
- 4 动态设计 Dynamic Design
  - 4.1 业务时序图
  - 4.2 业务流程图
- 5. 数据库/数据结构设计 （可选）
  - 5.1、数据库的设计
- 6. 对外接口设计
  - 6.1 log\_service\_init
  - 6.2 log\_service\_run
  - 6.3 log\_service\_stop
  - 6.4 log\_service\_deinit
  - 6.5 log\_trigger\_upload
- 7. 单元测试
  - 7.1. Mock工具

Version / 版本	Date / 日期	Responsible / 编写人	Reviewer / 审核人	Review Date / 审核日期	Change / 修改内容

## 1 文档介绍 Introduction

### 1.1 范围说明 Scope

Log Service主要负责Log采集、打包和传输。

IVI的Log分为SOC Log以及Tbox Log。其中主节点部署在SOC中，因此需要将Tbox的Log同步到SOC中。

### 1.2 参考文档 References

Ducment Name / 文档名称	Date / 日期	Version / 版本	Location / 位置

### 1.3 读者对象 Target Audience

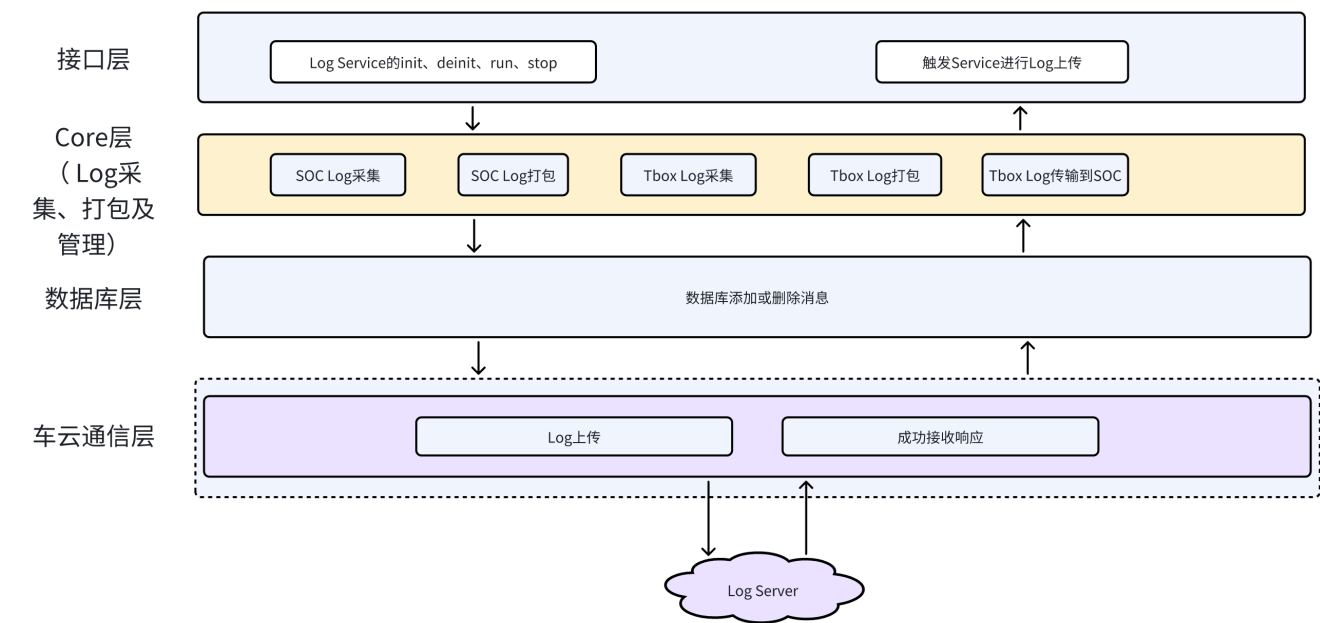
系统工程师，开发人员，测试人员，技术支持人员，及其他相关人员。

### 1.4. 术语和缩写 Definitions and Abbreviations

本章节对本文中使用的术语定义和缩写进行解释。


Definitions or Abbreviations / 术语或缩写	Explanation / 解释
--------------------------------------	------------------

## 2 架构设计 Architecture Design



## 3 静态设计 Static Design

### 3.1 详细模块/组件框图 Detailed Module/Component Diagram (颗粒度L3~L4级)

详细设计模块总体框图L3-L4级，可作为《》中“模块/组件框图”细化（注：如果没有细化的必要，则可以直接使用概要设计中的模块/组件框图），从总体上表达软件内部模块结构以及层次关系。

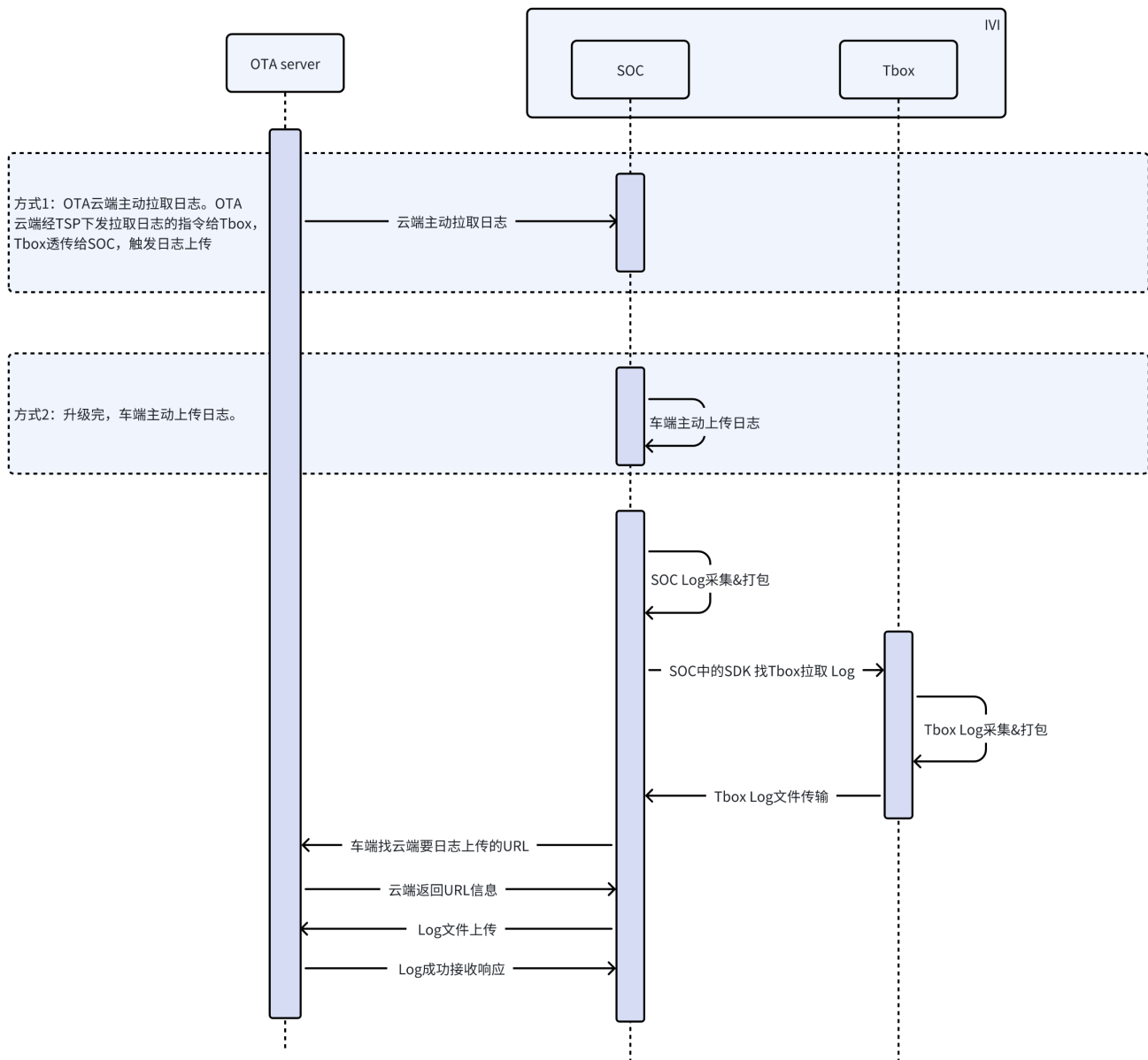
### 3.2 模块详细设计 Module Detailed Design (L4级别)

概要设计模块链接：

## 4 动态设计 Dynamic Design

### 4.1 业务时序图

Log的触发方式分为云端主动拉取和车端主动上传。



## 4.2 业务流程图



## 5. 数据库/数据结构设计（可选）

全局的（不属于某个模块的）设计数据库表结构或者数据结构，数据库请用ER图表示

Log搜集功能最终会将Log上传到云端，当车云通信的时候，可能会出现断电的情况，因此需要将车云通信的数据做落盘存储，当发送成功或发送失败达到指定的次数，可将待发送的消息清除。

### 5.1、数据库的设计

创建的数据表是ota\_log\_data，表的字段及类型如下所示

字段	uniqueId	vsid	taskType	state	filePathName	uploadTaskId	uploadUrl	uploadUrlEndtime	uploadStatus	upload
类型	INTEGER PRIMARY KEY AUTOINCREMENT UNIQUE	VARCHAR (64)	INTEGER	INTEGER	VARCHAR (192)	VARCHAR (64)	VARCHAR (384)	INTEGER	INTEGER	INTEGER
含义	自增的PRIMARY KEY。此值也是logUID，用来唯一记录1条Log上传的Taskid。	预留，此值是整车软件版本Id，可从获取。	0：表示云端任务 1：OTA升级任务	OTALOG_STATE_IDLE = 0, OTALOG_STATE_PRE = 10, // delete temp dir OTALOG_STATE_COLLECT = 20, // is collecting OTALOG_STATE_ALLZIP = 30, // compress temp dir to .zip, and delete temp dir OTALOG_STATE_URL = 40, // get upload url OTALOG_STATE_UPLOAD = 50, // is uploading OTALOG_STATE_COMPLETED = 60, //is upload completed OTALOG_STATE_FAILED = 70, // upload failed	压缩后的Log文件，即待上传的Log文件。	保留。日志上传的任务Id，通过车云协议传输。	日志上传时的服务器URL地址，通过车云协议传输。	uploadUrl对应的失效时刻，通过车云协议传输。	日志上传的状态。Log Service从文件上传模块处获取。  UPLOAD_IDLE_STATUS = 0, //idle UPLOAD_RUNNING_STATUS, //is uploading UPLOAD_SUCCESS_STATUS, //upload status UPLOAD_FAIL_STATUS, //upload fail UPLOAD_STOP_STATUS, //stop upload	预留，1

## 6. 对外接口设计

### 6.1 log\_service\_init

函数	int log_service_init(INIT_PHASE phase);
同步/异步	同步

可重入	不可重入	
参数 (in)	初始化时传入的参数	<pre>typedef enum {     INIT_PHASE_INSIDE=0,     INIT_PHASE_RESTORE,     INIT_PHASE_OUTSIDE,     INIT_PHASE_COUNT, }</pre>
参数 (inout)	无	
返回值	int	0: 成功. 其他: 失败.
描述	日志服务初始化	

6.2 log\_service\_run

函数	int log_service_run(void);	
同步/异步	同步	
可重入	不可重入	
参数 (in)	无	
参数 (inout)	无	
返回值	int	0: 成功. 其他: 失败.
描述	日志服务运行	

6.3 log\_service\_stop

函数	int log_service_stop(void);	
同步/异步	同步	
可重入	不可重入	
参数 (in)	无	
参数 (inout)	无	
返回值	int	0: 成功. 其他: 失败.
描述	配置服务停止	

6.4 log\_service\_deinit

函数	int log_service_deinit(void);	
同步/异步	同步	
可重入	不可重入	
参数 (in)	无	
参数 (inout)	无	
返回值	int	0: 成功. 其他: 失败.

描述	日志服务去初始化
----	----------

## 6.5 log\_trigger\_upload

函数	int log_trigger_upload(enum OTALOG_TASK_TYPE type);	
同步/异步	异步	
可重入	不可重入	
参数 (in)	OTALOG_TASK_TY E	enum OTALOG_TASK_TYPE { OTALOG_TASK_BY_CLOUD = 0, //云端触发 OTALOG_TASK_BY_UPGRADE, //升级触发 };
参数 (inout)	无	
返回值	int	0: 成功. 其他: 失败.
描述	触发日志上传，可能是车端自行触发，也可能是云端触发	

## 7. 单元测试

### 7.1. Mock工具

在某些情况下，不容易产生某些数据或状态。为了简化测试，采用Mock工具进行单元测试，当前在仓库中有单元测试的代码  
Gerrit代码地址是



测试步骤如下：

Step1: Cmake编译测试程序

```
cd log_service/unittest
mkdir build
cd build
cmake ../
make
```

Step2: 初始化测试环境

```
cd unittest
source env.sh
```

Step3: 开始测试

```
cd log_service/unittest
./log_service_test
```