

中科院声学所mrcp

概要设计说明

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| V1.0 | 段志亮 |  | 2021-11-24 | 创建文件 |
|  |  |  |  |  |

目录

[1.概述 3](#_Toc88817371)

[**1.1简介** 3](#_Toc88817372)

[1.2 读者对象 3](#_Toc88817373)

[1.3术语定义 4](#_Toc88817374)

[2.总体设计 4](#_Toc88817375)

[2.1 总体需求 4](#_Toc88817376)

[2.2 运行环境 5](#_Toc88817377)

[2.2.1 软件环境 5](#_Toc88817378)

[2.2.2 硬件环境 5](#_Toc88817379)

[2.3系统总体结构设计 5](#_Toc88817380)

[2.3.1 系统结构说明 5](#_Toc88817381)

[2.3.2系统总体时序流程 6](#_Toc88817382)

[3.mrcp服务模块设计 8](#_Toc88817383)

[3.1 对象管理模块 8](#_Toc88817384)

[3.2 网络通信模块 10](#_Toc88817385)

[3.3 流程控制模块 10](#_Toc88817386)

[4.配置及测试设计 13](#_Toc88817387)

[4.1 配置文件设计 13](#_Toc88817388)

[4.2 测试功能点 14](#_Toc88817389)

[5.可靠性设计 16](#_Toc88817390)

[5.1 健壮性设计 16](#_Toc88817391)

[5.2 故障恢复 17](#_Toc88817392)

[6.扩展性设计 17](#_Toc88817393)

[7.安全性设计 18](#_Toc88817394)

[8.易用性设计 19](#_Toc88817395)

# 1.概述

1

**1.1简介**

UniMRCP 是一个开源的、跨平台的 MRCP 协议实现，由 C / C ++语言 编写，包括了 MRCP 客户端和服务端两个部分，mrcp服务端接收客户端请求，然后和ASR进行交互，本文档重点对mrcp服务端进行介绍。

## 1.2 读者对象

本文档的读者对象为mrcp服务的开发人员、测试人员、系统维护人员及接入mrcp服务的第三方业务人员，通过本文档能够从总体上了解mrcp服务中识别模块的架构形式及数据流向。

本说明给出mrcp服务的设计说明，包括最终实现的系统必须满足的功能、性能、接口、附属测试工具程序及设计约束等。

目的在于：

* 为开发人员提供依据；
* 为代码修改、维护提供条件；

## 1.3术语定义

语音识别（Automatic Speech Recognition），简称（ASR）。

# 2.总体设计

2

## 2.1 总体需求

功能需求：接收客户端语音流，发送到ASR识别服务端，并给客户端返回语音流的识别结果；

性能需求：支持处理并发任务；

## 2.2 运行环境

### 2.2.1 软件环境

|  |  |  |
| --- | --- | --- |
| 分类 | 名称 | 版本 |
| 操作系统 | Centos | 7.0以上 |
| 编译环境 | gcc/g++ | 4.8.x |

### 2.2.2 硬件环境

|  |  |  |
| --- | --- | --- |
| 服务器 | 最低配置 | 推荐配置 |
| mrcp服务器 | CPU：2  内存：10G  磁盘：20G | CPU：4  内存：15G  磁盘：50G |

## 2.3系统总体结构设计

### 2.3.1 系统结构说明

mrcp识别服务系统主要分为：分布式转发服务和mrcp服务两个子系统。

分布式转发服务：对外提供调用ip和端口，转发客户端请求到不同的mrcp服务端；

mrcp识别服务：接收客户端会话语音流，发送到识别引擎，并返回识别结果。

系统总体流程图如下所示：



### 2.3.2系统总体时序流程

客户端、mrcp服务和asr服务总体时序流程图如下所示：



# 3.mrcp服务模块设计

3

mrcp识别服务包含对象管理模块、网络通信模块和流程控制模块。

模块间关系如下：



## 3.1 对象管理模块

线程管理模块负责在服务启动时，创建一定数量的http客户端对象；有新的会话接入时，从创建好的http客户端对象中分配一个对象；当会话结束时，回收分配的对象。

处理流程如下：



**相关函数介绍：**

**bool** **HttpClientManager::AllocAllHttpClients**(**int** num)

函数功能：该函数主要创建一定数量的http客户端对象。

**HttpClient \*HttpClientManager::AllocHttpClient**()

函数功能：该函数为新接入的会话分配对象。

**bool** **HttpClientManager::FreeHttpClient**(HttpClient \*httpclient)

函数功能：该函数回收结束会话对象。

## 3.2 网络通信模块

封装http请求方法，网络请求提供调用接口；

相关函数介绍：

**int** **GetJsonFromPostMethod**(std::string URL,std::string jsonObj,std::string &str)

函数功能：post方法请求http服务端，获取响应信息。

## 3.3 流程控制模块

流程控制模块主要功能为处理会话信令，根据会话请求设置参数、分配会话资源、检测语音流和请求ASR服务端进行语音识别，返回识别信息等。

会话控制流程图如下：



相关函数介绍：

**static** apt\_bool\_t **RecogMrcpRecognize**(mrcp\_engine\_channel\_t \*channel, mrcp\_message\_t \*request, mrcp\_message\_t \*response)

函数功能：该函数接收客户端识别请求，初始化资源；

**static** apt\_bool\_t **RecogMrcpTimersStart**(mrcp\_engine\_channel\_t \*channel, mrcp\_message\_t \*request, mrcp\_message\_t \*response)

函数功能：该函数接收客户端语音输入通知请求；

**static** apt\_bool\_t **RecogStartOfInput**(AsrRecogChannel \*recog\_channel)

函数功能：该函数通知客户端检测到有效语音；

**static** apt\_bool\_t **RecogVoiceStream**(mpf\_audio\_stream\_t \*stream, **const** mpf\_frame\_t \*frame)

函数功能：该函数检测语音流，请求asr识别服务，获取识别结果；

# 4.配置及测试设计

4

## 4.1 配置文件设计

设置配置文件，方便使用不同控制逻辑，支持不同的功能，mrcp服务配置文件实现如下目的：

* 通过参数控制支持特殊功能
* 检测异常数据
* 日志输出
* 服务地址配置

配置文件主要参数如下表：

|  |  |  |
| --- | --- | --- |
| 参数 | 取值 | 说明 |
| save\_voice\_data | true、false | false代表不保存语音；true代表保存接收到的语音 |
| max\_connection | 0-100 | 支持并发量 |
| enable\_bargin | true、false | 是否支持打断功能 |
| network\_timeout | 范围>=4000 | 接收或发送数据的网络超时时间，单位为ms |
| result\_encoding | 字符串 | 返回结果编码形式 |
| applicationId | 字符串 | 鉴权信息 |
| accessId | 字符串 | 鉴权信息 |
| accessSecret | 字符串 | 鉴权信息 |

## 4.2 测试功能点

根据设计需求，对服务必须满足的功能点进行验证测试。

|  |  |
| --- | --- |
| 测试功能 | 期望结果 |
| 服务启动验证 | 正确配置服务参数，启动服务，CPU、内存满足情况下，服务能够正常启动。 |
| 保存语音 | 配置保存语音参数，保存语音正常。 |
| 关闭保存语音 | 配置不保存语音参数，不保存语音。 |
| 关闭保存识别结果 | 配置不保存识别结果参数，不保存识别结果 |
| 支持并发验证 | 配置不同的并发数，进行并发请求 |
| 打断功能验证 | 配置打断参数，验证打断功能 |

# 5.可靠性设计

5

## 5.1 健壮性设计

服务运行过程中，客户端发送过来的异常数据及异常会话可能导致输出结果错误、服务崩溃等问题，为避免此类问题软件设计中有如下措施：

* 检查输入数据的数据类型,防止异常数据引起输出结果错误。
* 模块调用时检查参数的合法性,对非法参数请求返回错误信息提示。
* 简化软件的功能复杂性,与服务无关功能禁止加在服务上。
* 网络异常超时，返回对应错误信息。

如下是部分错误码信息：

|  |  |
| --- | --- |
| 错误码 | 说明 |
| 2 | 静音超时 |
| 6 | 识别结束，未获取到识别结果 |
| 11 | 超并发，分配对象失败 |

## 5.2 故障恢复

服务运行中异常情况导致的服务重启，mrcp服务需要不在人为干预的情况下自动重启。

# 6.扩展性设计

6

mrcp服务节点扩展，只需要在kamailio检测的服务列表中添加mrcp节点，然后执行加载命令即可，无需重启kamailo服务。

# 7.安全性设计

7

从客户端发送的语音数据，在请求ASR识别服务之前进行base64加密处理，同时从服务端返回的响应数据也进行base64加密，在传输的过程中一定程度上保证数据的安全性。

# 8.易用性设计

8

代码编译：在代码工程指定目录下执行编译脚本即可生成可执行文件；将可bin目录和plugin目录替换到部署系统的指定目录下即可实现系统更新；

系统部署：提供完整的文档介绍，用户可以根据文档说明完成系统部署启动；

系统调用：系统测试客户端和服务端在相同目录下，用户可以方便的使用客户端测试验证系统的正确性。