

# 百度语音合成 SDK Android 版 开发手册 V2.0



百度在线网络技术(北京)有限公司

(版权所有, 翻版必究)

## 目录

目录.....	2
1 概念解释.....	3
2 简介.....	3
2.1 兼容性.....	3
2.2 开发包说明.....	3
2.3 总体框图.....	3
3 集成指南.....	4
3.1 添加 BDTTSCClient 到工程.....	4
3.2 添加语音合成资源文件.....	4
3.3 权限声明.....	4
3.4 Proguard 配置.....	5
4 语音合成功能代码.....	5
4.1 Tts 初始化与授权等参数设置.....	5
4.2 参数设置.....	5
4.3 授权验证接口.....	6
4.4 合成并播放.....	6
4.5 流程控制.....	6
4.6 状态监听.....	6
4.7 只合成不播放.....	6
4.8 批量合成播放接口.....	7
4.9 获取合成数据.....	7
5 日志管理.....	7
FAQ.....	8

## 1 概念解释

语音合成是实现人机语音交互，建立一个有听和讲能力的交互系统所必需的关键技术。随着语音技术的发展，百度自主研发了语音合成系统（TTS），功能是接受用户发送的文本，生成语音发送给用户。

对本文中将提到的概念约定如下：

- **语音合成**：将文本合成为语音，即声音文件。
- **合成引擎**：将文本合成为语音的核心模块。
- **TTS**：Text To Speech，即“从文本到语音”。
- **BDTTSCient**：语音合成 SDK 简称，详见下条。
- **语音合成 SDK**：即本开发包，文中简称为 BDTTSCient。BDTTSCient 是一个封装了网络收发、音频播放功能的语音合成解决方案。借助 BDTTSCient 可以快速地在应用程序中集成语音合成功能。

## 2 简介

百度语音合成客户端 Android 版 SDK（以下简称 BDTTSCient）是一种面向 Android 移动设备的语音合成解决方案，以 JAR 包 + 动态链接库的形式发布。目前版本已支持 SDK 内部直接播放合成语音和从 SDK 获取语音数据，并支持男女声、语速、音调、音量、音频码率设置，后续版本将支持合成进度提示以及更多参数设置（—最新信息请参见语音合成 WEB 文档—）。

### 2.1 兼容性

系统：支持 Android 2.3（API Level 9）及以上，需要开发者通过 minSdkVersion 来保证支持系统的检测。

机型：手机和平板皆可。

构架：支持 arm64-v8a armeabi armeabi-v7a mips x86 x86\_64。

### 2.2 开发包说明

表 1 开发包说明表

文件(夹)名	说明
doc/Baidu_TTS_SDK_Android_Manual.pdf	本文档
doc/javaDoc	Javadoc
libs	语音合成 SDK Lib 库，包括各平台的 so 库及 Jar 包
BaiduTtsSample	开发示例(Eclipse project)
data	语音合成资源文件
license	授权文件

### 2.3 总体框图



图 1 语音合成系统架构图

## 3 集成指南

### 3.1 添加 BDTTSCClient 到工程

将开发包中的 `libs` 目录整体拷贝到工程目录，`libs` 目录包括了各平台的 `SO` 库，开发者视应用需要可以进行删减。`galaxy_lite.jar` 是百度 Android 公共基础库，如果项目中还集成了其它百度 SDK，如 `Push SDK`，在打包过程中出现类似如下的错误信息：

```
[2013-10-22 11:02:57 - Dex Loader] Unable to execute dex: Multiple dex files define Lcom/baidu/android/common/logging/Configuration;
[2013-10-22 11:02:57 - VoiceRecognitionDemo] Conversion to Dalvik format failed: Unable to execute dex: Multiple dex files define Lcom/baidu/android/common/logging/Configuration;
```

请将此 Jar 包移除。

如果 Eclipse ADT 版本插件低于 17，需要手工添加依赖库，添加方法为：`Project => Properties => Java Build Path => Libraries => Add JAR...`。

### 3.2 添加语音合成资源文件

将开发包中的 `data` 目录下的 `dat` 文件放到某一可读路径下，一遍设置资源文件参数时使用。

### 3.3 权限声明

BDTTSCClient 需要一些权限需要在 `AndroidManifest.xml` 文件，增加如下权限：

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

```
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.WRITE_SETTINGS" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
```

### 3.4 Proguard 配置

请不要混淆 sdk 的 jar 包。

## 4 语音合成功能代码

### 4.1 Tts 初始化与授权等参数设置

```
// 获取 tts 实例
speechSynthesizer = SpeechSynthesizer.getInstance();
// 设置 app 上下文（必需参数）
speechSynthesizer.setContext(Context);
// 设置 tts 监听器
speechSynthesizer.setSpeechSynthesizerListener(SpeechSynthesizerListener);
// 文本模型文件路径（离线引擎使用）
speechSynthesizer.setParam(SpeechSynthesizer.PARAM_TTS_TEXT_MODEL_FILE,
TEXT_MODEL_FILE_FULL_PATH_NAME);
// 声学模型文件路径（离线引擎使用）
speechSynthesizer.setParam(SpeechSynthesizer.PARAM_TTS_SPEECH_MODEL_FILE,
SPEECH_MODEL_FILE_FULL_PATH_NAME);
// 本地授权文件路径,如未设置将使用默认路径.设置临时授权文件路径,
LICENCE_FILE_NAME 请替换成临时授权文件的实际路径,仅在使用临时 license 文件时需要进行
设置,如果在[应用管理]中开通了离线授权,不需要设置该参数,建议将该行代码删除（离线引擎）
speechSynthesizer.setParam(SpeechSynthesizer.PARAM_TTS_LICENCE_FILE,
LICENSE_FILE_FULL_PATH_NAME);
// 请替换为语音开发者平台上注册应用得到的 App ID (离线授权)
speechSynthesizer.setAppId("your_app_id");
// 请替换为语音开发者平台注册应用得到的 apikey 和 secretkey (在线授权)
speechSynthesizer.setApiKey("your_api_key", "your_secret_key");
// 授权检测接口
AuthInfo authInfo = speechSynthesizer.auth(TtsMode);
// 引擎初始化接口
speechSynthesizer.initTts(TtsMode);
```

### 4.2 参数设置

在调用 speak 方法开始合成并朗读之前,可以对参数进行配置,代码如下（完整参数列表详见

JavaDoc)。

```
speechSynthesizer.setParam(SpeechSynthesizer.PARAM_VOLUME, "5");  
speechSynthesizer.setParam(SpeechSynthesizer.PARAM_SPEED, "5");  
speechSynthesizer.setParam(SpeechSynthesizer.PARAM_PITCH, "5");
```

参数默认值列表如下：

参数名	默认值	备注
PARAM_VOLUME	"5"	中级音量
PARAM_SPEED	"5"	中速
PARAM_PITCH	"5"	中调

## 4.3 授权验证接口

```
AuthInfo authInfo = speechSynthesizer.auth(TtsMode ttsMode)
```

## 4.4 合成并播放

```
speechSynthesizer.speak("百度一下");
```

该接口比较耗时，采用排队策略，调用后将自动加入合成队列，并按调用顺序进行合成和播放

```
speechSynthesizer.speak(String text, String utteranceId)
```

该接口功能和 `speak(String text)` 功能相同，参数 `utteranceId` 用来指定此次文本合成的标识，`utteranceId` 可以用数字区分 ("1","2"...)，或者全局唯一标识 `uuid`，总之用户可以选择自己可以用来区分的值即可。所有回调接口都会返回该标识 `utteranceId`，以方便用户做区分（不能用文本或者文本的 md5 区分是因为可能有相同文本存在）。

## 4.5 流程控制

通过 `pause()`, `resume()`, `stop()`, `release()` 对流程进行控制

## 4.6 状态监听

为了更好地实现用于界面，BDTTSCient 提供了 `SpeechSynthesizerListener` 监听接口用于对合成器的状态进行通知，监听器的回调方法参见 JavaDoc。

完整的开发示例请参见开发包所附示例 eclipse project——BaiduTtsSample。

## 4.7 只合成不播放

BDTTSCient 还支持仅获取合成数据，开发者可以在获取数据后先本地缓存，在合适的时候再进行播放，以避免较差网络情况下播放出现卡顿。参数设置同 4.1.2。

```
speechSynthesizer.synthesize("百度一下");
```

该接口比较耗时，采用排队策略，调用后将自动加入合成队列，并按调用顺序进行合成。

```
speechSynthesizer.synthesize(String text, String utteranceId)
```

该接口功能和 `synthesize(String text)` 功能相同，参数 `utteranceId` 用来指定此次文本合成的标识，`utteranceId` 可以用数字区分 ("1","2"...)，或者全局唯一标识 `uuid`，总之用户可以选择自己可以用来

区分的值即可。所有回调接口都会返回该标识 `utteranceId`，以方便用户做区分（不能用文本或者文本的 md5 区分是因为可能有相同文本存在）。

## 4.8 批量合成播放接口

```
int batchSpeak(java.util.List<SpeechSynthesizeBag> speechSynthesizeBags)
```

该接口可以批量传入多个文本并进行排队合成并播放（如果没有设置 `utteranceId`，则使用 `list` 的索引值作为 `utteranceId`）。

## 4.9 获取合成数据

通过 `SpeechSynthesizerListener` 的 `onSynthesizeDataArrived` 方法获取数据，代码如下：

```
public void onSynthesizeDataArrived(String utteranceId, byte[] audioData, int progress);
```

合成数据分多次返回。

## 5 日志管理

为了方便调试，BDTTS+提供了对日志的管理功能。

```
LoggerProxy.printable(boolean isPrint);
```

该接口控制日志的打印功能，如果设置为 `false`，将关闭所有日志；如果设置为 `true`，则开启日志打印功能。默认日志级别为 `debug` 级别

## FAQ

1. 如何联系我们？

开发过程中有任何 bug 或反馈意见，请发送邮件至 [voice\\_feedback@baidu.com](mailto:voice_feedback@baidu.com)

2. Unable to execute dex: Multiple dex files define Lcom/baidu/android/common/logging/Log

应用同时集成了百度其它 SDK，造成公共库冲突，请删除语音 SDK 中的 galaxy\_lite.jar