

# Table of Contents

关于Titan	1.1
版本履历	1.2
1.0.0	1.2.1
开发者指南	1.3
鉴权方式	1.4
API 文档	1.5
API 错误码	1.5.1
titan.android	1.5.2
TitanContext	1.5.2.1
DeviceNotSupportedException	1.5.2.2
MethodNotSupportedException	1.5.2.3
NotPermissionException	1.5.2.4
RemoteExecuteException	1.5.2.5
ServiceDeathException	1.5.2.6
ServiceNotFoundException	1.5.2.7
titan.android.dataacq	1.5.3
DataAcqService	1.5.3.1
ComponentStatus	1.5.3.2
titan.android.dataacq.hardware	1.5.4
AudioJack	1.5.4.1
Battery	1.5.4.2
Bluetooth	1.5.4.3
Camera	1.5.4.4
CPU	1.5.4.5
EMMC	1.5.4.6
Ethernet	1.5.4.7
Flashlight	1.5.4.8
GMChip	1.5.4.9
ICCard	1.5.4.10
Key	1.5.4.11
LCD	1.5.4.12
Magcard	1.5.4.13
Modem	1.5.4.14
Pinpad	1.5.4.15
Printer	1.5.4.16

RFCard	1.5.4.17
SafeModel	1.5.4.18
Sensors	1.5.4.19
TouchPanel	1.5.4.20
USB	1.5.4.21
Wifi	1.5.4.22
titan.android.dataacq.software	1.5.5
Security	1.5.5.1
SystemUsage	1.5.5.2

# 系统统一接口

## 项目背景

AND平台的智能POS按不同客户需求分别定制不同的系统API提供给客户调用。这部分API以不同形式存在，如广播，JAR包，AIDL等不同方式提供给客户。此外，提供出去的这些系统API忽略了应用权限检查，使不同身份的应用都可以调用系统接口，因此存在极大的安全隐患。智能POS系统统一接口服务是为了提供一套统一的，规范化的，以及基于应用身份检查的系统API接口。智能POS系统统一接口服务整合当前已存在的定制化API，并将之按功能分类，统一分配权限，开发者使用同一套API，而无须关心设备及版本差异。

系统统一接口服务框架采用Android接口定义语言（AIDL）作为服务进程与客户进程交互的设计语言。接口服务按接口类型划分为几个独立模块，各个模块提供同一类系统功能的接口。如数据采集模块，应用包模块，设置模块，位置服务模块等。

## 适用平台

系统统一接口服务框架适用于AND平台，如APOS A8， C10等运行于Android5.0及以上联迪智能POS终端。

版本

1.0.1

发布日期

2018-11-30

---

版本

1.0.0

发布日期

2018-10-30

## 版本号

1.0.0

## 发布日期

2018-10-30

## 主要更新

无

## OS版本

### • APOS

4.3.36

### • C10

1.3.6

## 依赖模块版本

#### • APOS平台

titan_conf	1.0.0
titan_seccheck	1.0.0
titan_sysfunc	1.0.0
facreset	1.2.15a

#### • C10平台

titan_conf	1.0.0
titan_seccheck	1.0.0
titan_sysfunc	1.0.0
facreset	1.2.15a
easytouch	1.3.7

下载地址

## 开发者指南

## 鉴权方式





# Errors

since 1.0.0

## 概览

系统统一接口返回值常量定义类。调用任何一个API，都可以通过TitanContext.getLastError()接口获取对应的错误码。

## 常量

名称	取值	说明
NONE	0x00	成功
NOTPERMISSION	0x10001	应用无权限
METHOD_NOT_SUPPORTED	0x10002	系统不支持, 一般是由于系统软件功能不存在或版本过低
DEVICE_NOT_SUPPORTED	0x10003	设备不支持, 一般是由于终端不存在某一硬件模块, 如电池, NFC芯片等硬件模块
ERRPARAM	0x11001	参数错误
APP_NOT_RUNNING	0x21001	App未运行
APP_NOT_INSTALLED	0x21002	App未安装
APK_NOT_FOUND	0x21003	Apk文件未找到
APP_UNSIGNED	0x21004	App未签名
BOOT_ANIMATION_ZIP_NOT_EXIST	0x22001	开机动画zip包不存在
COPY_BOOT_ANIMATION_ZIP_FAIL	0x22002	拷贝开机动画zip包到指定路径失败
CHMOD_BOOT_ANIMATION_ZIP_FIAL	0x22003	修改开机动画文件权限失败
CURRENT_IME_IS_NULL	0x22004	未设置当前输入法
IME_NOT_INSTALLED	0x22005	输入法未安装
ERR_SIM_STATE	0x23001	SIM卡状态错误
SET_PREFERRED_NETWORK_TYPE_FAIL	0x23002	设置网络首选项失败
UNKNOWN_PREFERRED_NETWORK_TYPE	0x23003	未知类型的网络首选项



# TitanContext

## Singleton Pattern

since 1.0.0

## 概览

系统统一接口上下文，应用程序使用系统统一接口服务的全局接口。通过TitanContext可以访问：

- [DataAcqService](#) 数据采集
- [EasyTouchService](#) 悬浮球
- [EthernetService](#) 以太网
- [ResetFactoryService](#) 恢复出厂设置
- [SystemFunctionService](#) 系统功能

等系统功能。

## 接口说明

### 构造方法

单例模式，不支持应用直接实例化，而是通过getContext(Context)接口获取实例。

### getTitanContext

```
public static TitanContext getTitanContext(Context context)
```

接口功能

获取TitanContext实例。

方法参数

- **context**: 应用上下文。

返回值

TitanContext唯一实例。

### getService

```
public T getService(@NonNull Class serviceType) throws ServiceNotFoundException
```

接口功能

获取统一接口远程服务在应用端的代理对象,通过服务代理对象访问系统功能。

方法参数

- **serviceType**: 服务代理对象类型，不可为null。枚举类型有：
  - DataAcqService.class
  - EasyTouchService.class
  - EthernetService.class
  - ResetFactoryService.class
  - SystemFunctionService.class

返回值

服务代理对象，由入参**serviceType**指定代理对象类型。

异常

- **ServiceNotFoundException**: **serviceType**所指类型服务不存在时抛出此异常。

## 编程示例:

```
TitanContext titanContext = TitanContext.getTitanContext(context);
try {
    SystemFunctionService service = titanContext.getService(SystemFunctionService.class);
    ApplicationModule application = service.getFunction(ApplicationModule.class);
    AppSignInInfo appSignInInfo = application.getAppSignInInfo("com.example.demo");
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# DeviceNotSupportedException

extends titan.android.RemoteExecuteException

since 1.0.0

## 概览

应用调用统一API时，设备不支持此功能时，抛出此异常，属于硬件范畴。如ACER设备不支持电源数据采集，当应用采集电源数据时，抛出此异常。

## 错误码

0x10003

# MethodNotSupportedException

extends titan.android.RemoteExecuteException

since 1.0.0

## 概览

应用调用统一API时，统一接口服务不支持此功能时，抛出此异常, 属于软件系统范畴。

## 错误码

0x10002

# NotPermissionException

extends titan.android.RemoteExecuteException

since 1.0.0

## 概览

应用无权限异常.

## 错误码

0x10001



# RemoteExecuteException

extends `java.lang.Exception`

since 1.0.0

## 概览

系统统一接口功能执行异常。

应用使用统一API接口时，当出现以下几种情况时会抛出RemoteExecuteException异常或其子类异常：

- 参数错误时抛出 RemoteExecuteException
- 应用无权限时抛出 NotPermissionException
- 设备不支持,如设备不具备执行此接口的条件时抛出 DeviceNotSupportedException
- 功能不支持,如系统服务未升级到满足该接口的版本或与远程服务交互异常时抛出 MethodNotSupportedException

# ServiceDeathException

extends `java.lang.Exception`

since 1.0.0

## 概览

应用调用统一接口时，接口服务未运行时抛出此异常.服务未运行一般有以下情况:

- 服务未安装
- 服务已安装但未启动
- 服务运行过程中出现崩溃，崩溃后未重启

# ServiceNotFoundException

extends `java.lang.Exception`

since 1.0.0

## 概览

当应用调用 `TitanContext.getService(Class)` 传入的 `class` 类型不被识别时抛出此异常。



# DataAcqService

extends titan.android.core.TitanService

since 1.0.0

## 概览

系统统一接口数据采集服务. 用于获取设备的统计量和状态量. 不能直接实例化DataAcqService, 而是通过TitanContext.getService(Class)接口, 传入参数 DataAcqService.class 获取采集服务代理对象.

需要采集某一模块的统计量或状态量时, 通过getComponent(Class) 接口获取对应模块的实例, 入参为模块的类型, 如获取音频孔模块, 传入AudioJack.class, 如果远程采集服务未运行, 则抛出 ServiceDeathException异常.

## 接口说明

### 构造方法

不支持应用直接实例化, 而是通过TitanContext.getService(DataAcqService.class)接口获取实例。

### getComponent

```
public T getComponent(@NonNull Class clazz) throws ServiceDeathException, ServiceNotFoundException
```

接口功能

获取采集模块.

方法参数

- **clazz**: 服务代理对象类型, 不可为null。枚举类型有:
  - AudioJack.class
  - Battery.class
  - Bluetooth.class
  - Camera.class
  - CPU.class
  - 其他类型...

返回值

采集模块实例, 实例类型即为入参clazz指定的类型。

异常

- ServiceNotFoundException: clazz所指模块类型不存在时抛出此异常。
- ServiceDeathException: 远程服务未运行时抛出

## 编程示例:

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    AudioJack audioJack = dataAcq.getComponent(AudioJack.class);
    // 获取音频头插入次数统计量
    long count = audioJack.getUsage(AudioJack.PLUG_COUNT);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
}
```

# ComponentStatus

since 1.0.0

## 概览

采集模块状态量包装类。

## 枚举类

Level

状态码等级

- S：正常
- E：故障
- W：警告

## 接口说明

### getEncoder

```
public String getEncoder()
```

接口功能

获取编码定义方。

返回值

编码定义方。

### getLevel

```
public Level getLevel()
```

接口功能

获取状态码级别。

返回值

状态码级别。

### getDescription

```
public String getDescription()
```

接口功能

获取状态码简单描述。

返回值

状态码简单描述。

## | getTagNo

```
public int getTagNo()
```

接口功能

获取采集因子tagNo。

返回值

采集因子tagNo。

## | getFactorNo

```
public int getFactorNo()
```

接口功能

获取采集因子factorNo。

返回值

采集因子factorNo。

## | getStatusCode

```
public String getStatusCode()
```

接口功能

获取状态码，16进制表示的状态编码，如0x41。

返回值

状态码。





# AudioJack

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

音频孔数据采集. 通过DataAcqService.getComponent(AudioJack.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
PLUG_COUNT	Long	获取音频头插入次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(AudioJack.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - PLUG\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    AudioJack audioJack = dataAcq.getComponent(AudioJack.class);
    //获取音频头插入次数
    long count = audioJack.getUsage(AudioJack.PLUG_COUNT);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# Battery

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

电池数据采集. 通过DataAcqService.getComponent(Battery.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
POWER_LEVEL	String	获取电池电量
BATTERY_VOLTAGE	String	获取电池电压
STANDBY_BATTERY_VOLTAGE	String	获取备份电池电压
CHARGE_DURATION	Long	获取充电累计时间(分钟)
POWER_USAGE	String	获取电量使用情况(mAh)
CHARGE_COUNT	Long	获取充电次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Battery.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - POWER\_LEVEL
  - BATTERY\_VOLTAGE
  - STANDBY\_BATTERY\_VOLTAGE
  - CHARGE\_DURATION
  - POWER\_USAGE

- CHARGE\_COUNT

返回值

统计量。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getStatus

`public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取电池状态量信息。

返回值

电池状态量信息。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Battery battery = dataAcq.getComponent(Battery.class);
    //获取充电累计时间(分钟)
    long count = battery.getUsage(Battery.CHARGE_DURATION);
    //获取电池电压
    String voltage = battery.getUsage(Battery.BATTERY_VOLTAGE);
    //获取电池状态
    ComponentStatus status = battery.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
}
```

```
} catch (RemoteExecuteException e) {  
    e.printStackTrace();  
}
```

# Bluetooth

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

蓝牙数据采集. 通过DataAcqService.getComponent(Bluetooth.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
BLUETOOTH_MODEL	String	获取蓝牙型号
BLUETOOTH_MAC	String	获取蓝牙MAC地址
BLUETOOTH_OPEN_COUNT	Long	获取蓝牙打开次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Bluetooth.class)接口获取实例。

### getUsage

```
public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException
```

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - BLUETOOTH\_MODEL
  - BLUETOOTH\_MAC
  - BLUETOOTH\_OPEN\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。

- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getStatus

public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取蓝牙运行状态。

返回值

蓝牙运行状态量信息。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Bluetooth bluetooth = dataAcq.getComponent(Bluetooth.class);
    //获取蓝牙型号
    String model = bluetooth.getUsage(Bluetooth.BLUETOOTH_MODEL);
    //获取蓝牙运行状态
    ComponentStatus status = bluetooth.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```



# Camera

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

摄像头数据采集. 通过DataAcqService.getComponent(Camera.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
FRONT_CAMERA_COUNT	Long	获取前置摄像头打开次数
BACK_CAMERA_COUNT	Long	获取后置摄像头打开次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Camera.class)接口获取实例。

### getUsage

```
public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException
```

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - FRONT\_CAMERA\_COUNT
  - BACK\_CAMERA\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限

- 数据解析失败
- 采集因子不被支持
- 系统不支持

## getFrontCameraStatus

`public ComponentStatus getFrontCameraStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取前置摄像头状态。

返回值

前置摄像头状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getBackCameraStatus

`public ComponentStatus getBackCameraStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取后置摄像头状态。

返回值

后置摄像头状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Camera camera = dataAcq.getComponent(Camera.class);
    //获取前置摄像头打开次数
    long count = camera.getUsage(Camera.FRONT_CAMERA_COUNT);
}
```

```
// 获取前置摄像头状态
ComponentStatus status = camera.getFrontCameraStatus();
//状态码等级
Level level = status.getLevel();
//状态码
String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# CPU

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

CPU数据采集. 通过DataAcqService.getComponent(CPU.class)获取实例。

## 采集常量

名称	<b>getUsage</b> 返回类型	说明
CPU_MODEL	String	获取CPU型号

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(CPU.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- **request**: 采集请求，不可为null。传入[采集常量](#):
  - CPU\_MODEL

返回值

统计量。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    CPU cpu = dataAcq.getComponent(CPU.class);
    //获取CPU型号
    String model = cpu.getUsage(CPU.CPU_MODEL);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# EMMC

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

EMMC模块采集. 通过DataAcqService.getComponent(EMMC.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
EMMC_MANUFACTURER	String	获取EMMC厂商
EMMC_MODEL	String	获取EMMC型号
EMMC_LIFE	String	获取EMMC使用寿命

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(EMMC.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - EMMC\_MANUFACTURER
  - EMMC\_MODEL
  - EMMC\_LIFE

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。

- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    EMMC emmc = dataAcq.getComponent(EMMC.class);
    //获取EMMC厂商
    String manu = camera.getUsage(EMMC.EMMC_MANUFACTURER);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# Flashlight

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

闪光灯数据采集. 通过DataAcqService.getComponent(Flashlight.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
FLASH_OPEN_COUNT	Long	获取闪光灯打开次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Flashlight.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - FLASH\_OPEN\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持



## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(Flashlight.class);
    Flashlight flashlight = dataAcq.getComponent(Flashlight.class);
    //获取闪光灯打开次数
    Long count = flashlight.getUsage(Flashlight.FLASH_OPEN_COUNT);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# GMChip

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

国密芯片数据采集. 通过DataAcqService.getComponent(GMChip.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
ENCRYPT_IC_MODEL	String	获取国密芯片型号

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(GMChip.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - ENCRYPT\_IC\_MODEL

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    GMChip chip = dataAcq.getComponent(GMChip.class);
    //获取国密芯片型号
    String model = chip.getUsage(GMChip.ENCRYPT_IC_MODEL);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# ICCard

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

接触式卡模块数据采集. 通过DataAcqService.getComponent(ICCard.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
POWERUP_SUCCESS_COUNT	Long	获取IC卡上电成功次数
POWERUP_FAIL_COUNT	Long	获取IC卡上电失败次数
APDU_FAIL_COUNT	Long	获取IC卡APDU失败次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(ICCard.class)接口获取实例。

### getUsage

```
public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException
```

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - POWERUP\_SUCCESS\_COUNT
  - POWERUP\_FAIL\_COUNT
  - APDU\_FAIL\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。

- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getICCard1Status

public ComponentStatus getICCard1Status() throws ServiceDeathException, RemoteExecuteException

接口功能

获取IC卡设备状态。

返回值

IC卡设备状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getICCard2Status

public ComponentStatus getICCard2Status() throws ServiceDeathException, RemoteExecuteException

接口功能

获取IC卡2设备状态。

返回值

IC卡2设备状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getSAM1CardStatus

public ComponentStatus getSAM1CardStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取SAM1设备状态。

返回值

SAM1设备状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getSAM2CardStatus

`public ComponentStatus getSAM2CardStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取SAM2设备状态。

返回值

SAM2设备状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getSAM3CardStatus

`public ComponentStatus getSAM3CardStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取SAM3设备状态。

返回值

SAM3设备状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持

- 系统不支持

## getSAM4CardStatus

public ComponentStatus getSAM4CardStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取SAM4设备状态。

返回值

SAM4设备状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    ICCard iccard = dataAcq.getComponent(ICCard.class);
    //获取IC卡上电成功次数
    long count = iccard.getUsage(ICCard.POWERUP_SUCCESS_COUNT);
    //获取ICCard设备1状态
    ComponentStatus status = iccard.getICCard1Status();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# Key

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

电源键数据采集. 通过DataAcqService.getComponent(Key.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
POWER_KEY_CLICK_COUNT	Long	获取电源键按键次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Key.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - POWER\_KEY\_CLICK\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持



## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Key key = dataAcq.getComponent(Key.class);
    // 获取电源键按键次数
    long count = key.getUsage(Key.POWER_KEY_CLICK_COUNT);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# LCD

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

显示屏数据采集. 通过DataAcqService.getComponent(LCD.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
SCREENON_DURATION	Long	获取屏幕背光点亮时间(分钟)
SCREEN_CLICK_COUNT	Long	获取屏幕点击次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(LCD.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - SCREENON\_DURATION
  - SCREEN\_CLICK\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时

- 数据解析失败
- 采集因子不被支持
- 系统不支持

## getStatus

public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取显示屏接口通信状态。

返回值

显示屏接口通信状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    LCD lcd = dataAcq.getComponent(LCD.class);
    // 获取屏幕点击次数
    long count = lcd.getUsage(LCD.SCREEN_CLICK_COUNT);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# Magcard

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

磁条卡模块数据采集. 通过DataAcqService.getComponent(Magcard.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
SWIPE_COUNT	Long	获取刷卡总次数
TRK1_SUCCESS_COUNT	Long	获取1磁道成功次数
TRK1_FAIL_COUNT	Long	获取1磁道失败次数
TRK2_FAIL_COUNT	Long	获取2磁道失败次数
TRK3_SUCCESS_COUNT	Long	获取3磁道成功次数
TRK3_FAIL_COUNT	Long	获取3磁道失败次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Battery.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据.

方法参数

- request: 采集请求，不可为null。传入[采集常量](#):
  - SWIPE\_COUNT
  - TRK1\_SUCCESS\_COUNT
  - TRK1\_FAIL\_COUNT
  - TRK2\_FAIL\_COUNT
  - TRK3\_SUCCESS\_COUNT

- TRK3\_FAIL\_COUNT

返回值

统计量。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getStatus

`public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取磁卡模块状态。

返回值

电池状态量信息。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Magcard magcard = dataAcq.getComponent(Magcard.class);
    //获取刷卡总次数
    long count = magcard.getUsage(Magcard.SWIPE_COUNT);
    //获取磁卡模块状态
    ComponentStatus status = magcard.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

}

# Modem

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

Modem数据采集. 通过DataAcqService.getComponent(Modem.class)获取实例。

## 采集常量

名称	<b>getUsage</b> 返回类型	说明
DIAL_COUNT	Long	获取MODEM拨号次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Modem.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- **request**: 采集请求，不可为null。传入[采集常量](#):
  - DIAL\_COUNT

返回值

统计量。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Modem modem = dataAcq.getComponent(Modem.class);
    //获取MODEM拨号次数
    long count = modem.getUsage(Modem.DIAL_COUNT);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```



# Pinpad

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

Pinpad模块数据采集. 通过DataAcqService.getComponent(Pinpad.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
OFFLINE_COUNT	Long	获取脱机PIN输入次数
ONLINE_COUNT	Long	获取联机PIN输入次数
NOPIN_COUNT	Long	获取非PIN输入次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Battery.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - OFFLINE\_COUNT
  - ONLINE\_COUNT
  - NOPIN\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。

- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getStatus

public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取密码键盘状态。

返回值

密码键盘状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Pinpad pinpad = dataAcq.getComponent(Pinpad.class);
    //获取联机PIN输入次数
    long count = pinpad.getUsage(Pinpad.ONLINE_COUNT);
    //获取密码键盘状态
    ComponentStatus status = pinpad.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# Printer

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

打印设备数据采集. 通过DataAcqService.getComponent(Printer.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
HEAT_TEMP_COUNT	Long	获取打印高温次数
LOW_TEMP_COUNT	Long	获取打印低温次数
FEED_COUNT	Long	获取走纸电机步进量
PRINTER_MODEL	String	获取打印机芯型号
PRINTER_PAPER_LENGTH	Long	获取打印走纸长度(毫米)

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Printer.class)接口获取实例。

### getUsage

```
public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException
```

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - HEAT\_TEMP\_COUNT
  - LOW\_TEMP\_COUNT
  - FEED\_COUNT
  - PRINTER\_MODEL
  - PRINTER\_PAPER\_LENGTH

返回值

统计量。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getStatus

`public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取打印设备状态。

返回值

打印设备状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Printer printer = dataAcq.getComponent(Printer.class);
    //获取打印高温次数
    long count = printer.getUsage(Printer.HEAT_TEMP_COUNT);
    //获取打印设备状态
    ComponentStatus status = printer.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```



# RFCard

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

非接触式射频模块数据采集. 通过DataAcqService.getComponent(RFCard.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
ACTIVATE_SUCCESS_COUNT	Long	获取射频卡激活成功量
ACTIVATE_FAIL_COUNT	Long	获取射频卡激活失败量
APDU_FAIL_COUNT	Long	获取射频卡数据交换失败量
RFCARD_MODEL	String	获取射频卡芯片型号

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(RFCard.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - ACTIVATE\_SUCCESS\_COUNT
  - ACTIVATE\_FAIL\_COUNT
  - APDU\_FAIL\_COUNT
  - RFCARD\_MODEL

返回值

统计量。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getStatus

`public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取射频卡设备状态。

返回值

射频卡设备状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    RFCard rfcard = dataAcq.getComponent(RFCard.class);
    //获取射频卡激活成功量
    long count = rfcard.getUsage(RFCard.ACTIVATE_SUCCESS_COUNT);
    //获取射频卡设备状态
    ComponentStatus status = rfcard.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```





# SafeModel

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

安全模块数据采集. 通过DataAcqService.getComponent(SafeModel.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
SAFE_MODEL	String	获取安全模块型号

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(SafeModel.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - SAFE\_MODEL

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    SafeModel safeModel = dataAcq.getComponent(SafeModel.class);
    //获取安全模块型号
    String model = safeModel.getUsage(SafeModel.SAFE_MODEL);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# Sensors

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

传感器数据采集. 通过DataAcqService.getComponent(Sensors.class)获取实例。

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Sensors.class)接口获取实例。

### getLightSensorStatus

public ComponentStatus getLightSensorStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取光传感器通信状态。

返回值

光传感器通信状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

### getAccSensorStatus

public ComponentStatus getAccSensorStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取加速传感器通信状态。

返回值

加速传感器通信状态。

异常

- `ServiceDeathException` 远程服务未运行时抛出。
- `RemoteExecuteException` 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Sensors sensors = dataAcq.getComponent(Sensors.class);
    //获取打印设备状态
    ComponentStatus status = sensors.getLightSensorStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# TouchPanel

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

触屏数据采集. 通过DataAcqService.getComponent(TouchPanel.class)获取实例。

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(TouchPanel.class)接口获取实例。

### getStatus

public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取触屏通信状态。

返回值

触屏通信状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    TouchPanel touchPanel = dataAcq.getComponent(TouchPanel.class);
    //获取触屏通信状态
    ComponentStatus status = touchPanel.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
}
```

```
} catch (ServiceNotFoundException e) {  
    e.printStackTrace();  
}  
} catch (ServiceDeathException e) {  
    e.printStackTrace();  
}  
} catch (RemoteExecuteException e) {  
    e.printStackTrace();  
}  
}
```

# USB

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

USB数据采集. 通过DataAcqService.getComponent(USB.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
USB_PLUG_COUNT	Long	获取USB连接次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(USB.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - USB\_PLUG\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    USB usb = dataAcq.getComponent(USB.class);
    //获取USB连接次数
    long count = usb.getUsage(USB.USB_PLUG_COUNT);
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```



# Wifi

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

Wifi数据采集. 通过DataAcqService.getComponent(Wifi.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
WIFI_MODEL	String	获取WIFI型号
WIFI_MAC	String	获取WIFI MAC地址

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Wifi.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - WIFI\_MODEL
  - WIFI\_MAC

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时

- 数据解析失败
- 采集因子不被支持
- 系统不支持

## getStatus

public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取WIFI驱动加载状态。

返回值

WIFI驱动加载状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Wifi wifi = dataAcq.getComponent(Wifi.class);
    //获取WIFI MAC地址
    String mac = wifi.getUsage(Wifi.WIFI_MAC);
    //获取WIFI驱动加载状态
    ComponentStatus status = wifi.getStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```



# Security

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

安全模块数据采集. 通过DataAcqService.getComponent(Security.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
CLEARATTACK_COUNT	Long	获取清除攻击次数

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(Security.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- request: 采集请求，不可为null。传入采集常量：
  - CLEARATTACK\_COUNT

返回值

统计量。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getSecAttackDetectStatus

public ComponentStatus getSecAttackDetectStatus() throws ServiceDeathException, RemoteExecuteException

接口功能

获取SEC攻击检测状态。

返回值

SEC攻击检测状态。

异常

- ServiceDeathException 远程服务未运行时抛出。
- RemoteExecuteException 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
    Security security = dataAcq.getComponent(Security.class);
    //获取清除攻击次数
    long count = security.getUsage(Security.CLEARATTACK_COUNT);
    //获取SEC攻击检测
    ComponentStatus status = security.getSecAttackDetectStatus();
    //状态码等级
    Level level = status.getLevel();
    //状态码
    String statusCode = status.getStatusCode();
} catch (ServiceNotFoundException e) {
    e.printStackTrace();
} catch (ServiceDeathException e) {
    e.printStackTrace();
} catch (RemoteExecuteException e) {
    e.printStackTrace();
}
```

# SystemUsage

extends titan.android.core.dadaacq.StatsComponent

since 1.0.0

## 概览

系统信息数据采集. 通过DataAcqService.getComponent(SystemUsage.class)获取实例。

## 采集常量

名称	getUsage 返回类型	说明
POWERON_COUNT	Long	获取开机次数
SHUTDOWN_COUNT	Long	获取关机次数
LIFECYCLE_TOTAL	Long	获取累计运行时间(分钟)
PRODUCT_NAME	String	获取品牌名称
MODEL_NAME	String	获取型号
DEVICE_NAME	String	获取设备名称
SDCARD_TOTAL	Long	获取内置SD卡容量(bytes)
SDCARD_AVAILABLE	Long	获取内置SD卡可用大小(bytes)
NET_TYPE	String	获取网络类型
SDK_VERSION	String	获取SDK版本
ANDROID_VERSION	String	获取Android版本
UPTIME_INFO	Long	获取系统运行时间(分钟)
BASEBAND_VERSION	String	获取基带版本
SYSTEM_VERSION	String	获取系统版本
SMARTOS_VERSION	String	获取SmartOS版本
MEMORY_TOTAL	Long	获取系统总内存大小(bytes)
MEMORY_AVAILABLE	Long	获取系统可用内存大小(bytes)
DEVICE_MANUFACTURER	String	获取厂商名称
SERIAL_NO	String	获取序列号
IMEI	String	获取IMEI号
MEID	String	获取MEID号
SAFE_PATCH_VERSION	String	获取安全补丁版本

KERNEL_VERSION	String	获取内核版本号
ROM_TOTAL	Long	获取机身存储容量(bytes)
ROM_AVAILABLE	Long	获取机身存储可用大小(bytes)
TFCARD_TOTAL	Long	获取TF卡容量(bytes)
TFCARD_AVAILABLE	Long	获取TF卡可用大小(bytes)
IP_ADDRESS	String	获取IP地址
BOOT_VERSION	String	获取Boot版本
SDCARD_STATUS	String	获取内置SD卡状态
TFCARD_STATUS	String	获取TF卡状态

## 接口说明

### 构造方法

不支持应用直接实例化，而是通过DataAcqService.getComponent(SystemUsage.class)接口获取实例。

### getUsage

public T getUsage(DataAcqRequest request) throws ServiceDeathException, RemoteExecuteException

接口功能

获取采集数据。

方法参数

- **request:** 采集请求，不可为null。传入采集常量：
  - POWERON\_COUNT
  - SHUTDOWN\_COUNT
  - LIFECYCLE\_TOTAL
  - PRODUCT\_NAME
  - MODEL\_NAME
  - DEVICE\_NAME
  - SDCARD\_TOTAL
  - SDCARD\_AVAILABLE
  - NET\_TYPE
  - SDK\_VERSION
  - ANDROID\_VERSION
  - UPTIME\_INFO
  - BASEBAND\_VERSION
  - SYSTEM\_VERSION
  - SMARTOS\_VERSION
  - MEMORY\_TOTAL
  - MEMORY\_AVAILABLE
  - DEVICE\_MANUFACTURER
  - SERIAL\_NO

- IMEI
- MEID
- SAFE\_PATCH\_VERSION
- KERNEL\_VERSION
- ROM\_TOTAL
- ROM\_AVAILABLE
- TFCARD\_TOTAL
- TFCARD\_AVAILABLE
- IP\_ADDRESS
- BOOT\_VERSION
- SDCARD\_STATUS
- TFCARD\_STATUS

返回值

统计量。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限时
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## getStatus

`public ComponentStatus getStatus() throws ServiceDeathException, RemoteExecuteException`

接口功能

获取打印设备状态。

返回值

打印设备状态。

异常

- **ServiceDeathException** 远程服务未运行时抛出。
- **RemoteExecuteException** 以下错误时抛出：
  - 应用无权限
  - 数据解析失败
  - 采集因子不被支持
  - 系统不支持

## 编程示例：

```
TitanContext titan = TitanContext.getTitanContext(context);
try {
    DataAcqService dataAcq = titan.getService(DataAcqService.class);
```



```
        SystemUsage systemUsage = dataAcq.getComponent(SystemUsage.class);
        //获取开机次数
        long count = systemUsage.getUsage(SystemUsage.POWERON_COUNT);
    } catch (ServiceNotFoundException e) {
        e.printStackTrace();
    } catch (ServiceDeathException e) {
        e.printStackTrace();
    } catch (RemoteExecuteException e) {
        e.printStackTrace();
    }
}
```