# dbus package

* 介绍

dbus package包括一个dbus消息路由程序dbus-daemon，一个dbus库libdbus，和其他一些dbus工具程序比如dbus-monitor。

* 版本号

要求使用dbus 1.8.8版本

建议下载地址<http://dbus.freedesktop.org/releases/dbus/dbus-1.8.8.tar.gz>

* Configure 选项

--enable-shared \

--enable-static \

--disable-abstract-sockets \

--disable-ansi \

--disable-asserts \

--disable-console-owner-file \

--disable-doxygen-docs \

--disable-compiler\_coverage \

--disable-selinux \

--disable-tests \

--enable-verbose-mode \

--disable-xml-docs \

--with-xml="expat" \

--with-dbus-user=root \

--with-dbus-daemondir="/usr/sbin" \

--with-system-socket="/var/run/dbus/system\_bus\_socket" \

--with-system-pid-file="/var/run/dbus.pid" \

--without-x \

--libexecdir=/usr/lib/dbus-1

* Make选项

无

* 安装文件

Libdbus库文件安装到/usr/lib目录

配置文件（包括session.conf/system.conf以及session.d/和system.d两个目录）安装到/etc/dbus-1目录下

Dbus-daemon安装到/usr/sbin目录

Dbus-uuidgen安装到/usr/sbin目录

Dbus的启动脚本（负责启动dbus-daemon for system bus）

Dbus-send/dbus-monitor/dbus-clean-sockets安装到/usr/bin目录

* 默认配置

使用dbus自带的配置文件即可

* 启动

启动时执行/usr/bin/dbus-uuidgen –ensure生成/var/lib/dbus/machine-id文件

执行dbus-daemon –system启动system bus，启动时机只要保证在App Framework之前启动即可

* SDK

libdbus的头文件和.so/.a库，要求安装到SDK中，供电信和开发者使用。

# glib package

* 介绍

Glib 提供了 gdbus库，gdbus和libdbus类似，也是一个dbus协议的实现，可以用来实现dbus server/client。Gdbus相对libdbus 提供的API更好用，我们用gdbus替代libdbus作为dbus server/client的开发库。

Gdbus无法独立于glib工作，因此必须编译整个glib。

* 版本号

要求使用2.41.1版本。

建议下载地址<http://ftp.gnome.org/pub/GNOME/sources/glib/2.41/glib-2.41.1.tar.xz>

* Configure 选项

--enable-shared \

--enable-static \

--enable-debug=no \

--disable-selinux \

--disable-fam

* Make选项

无

* 安装文件

库文件（libgio/libgobject/libgthread/libglib/libgmodule）安装到/usr/lib/glib-2.0目录

Gdbus程序安装到/usr/bin目录

* 默认配置

无

* 启动

无

* SDK

Glib相关头文件安装到SDK中，供电信和开发者使用

# DBus服务通用规范

本章节描述智能网关中DBus服务定义的通用规范，无论是系统软件模块还是第三方应用，在定义DBus服务时均需遵守此规范。

## DBus标准接口

以下几个DBus接口属于标准接口，和具体业务无关。

* org.freedesktop.DBus.Peer

本接口是DBus规范定义的标准接口，详情参见DBus规范。

要求所有导出的Object均实现本接口。

* org.freedesktop.DBus.Introspectable

本接口是DBus规范定义的标准接口，详情参见DBus规范。

要求所有导出的对象均实现本接口。

* com.ctc.igd1.Properties

本接口是对org.freedesktop.DBus.Properties接口的扩展，接口包含如下方法：

Com.ctc.igd1.Properties.Get(in STRING interface\_name,

in STRING property\_name,

out VARIANT value);

com.ctc.igd1.Properties.Set (in STRING interface\_name,

in STRING property\_name,

in VARIANT value);

com.ctc.igd1.Properties.GetAll (in STRING interface\_name,

out DICT<STRING,VARIANT> props);

com.ctc.igd1.Properties.SetMulti (in STRING interface\_name,

out DICT<STRING,VARIANT> props);

Get/Set/GetAll接口和DBus规范定义一致。

新增的方法是SetMulti，用于一次设置多个Property，减少交互次数，提高服务端的处理效率。

本接口的信号为com.ctc.igd1.Properties.PropertiesChanged，和Dbus规范一致。

要求所有导出的对象，即使改对象实现的接口中没有任何属性，都必须实现本接口。

* com.ctc.igd1.ObjectManager

本接口是对org.freedesktop.DBus.ObjectManager的扩展，包含方法如下：

com.ctc.igd1.ObjectManager.GetManagedObjects(out DICT<OBJPATH,DICT<STRING,DICT<STRING,VARIANT>>> objpath\_interfaces\_and\_properties);

com.ctc.igd1.ObjectManager.AddObject(in DICT<STRING,VARIANT>,

out OBJPATH objpath);

com.ctc.igd1.ObjectManager.DeleteObject(in OBJPATH objpath);

新增的方法是AddObject和DeleteObject，用于增加和删除对象。

本接口的信号为com.ctc.igd1.ObjectManager.InterfaceAdded和com.ctc.igd1.ObjectManager.InterfaceRemoved，和DBus规范一致。

本接口非必须接口，如需实现，在对象的定义中会指定。

## DBus服务命名

* 系统软件提供的DBus服务，命名为com.ctc.igd1。
* Appframework提供DBus服务，命名为com.ctc.appframework1。
* App提供的DBus服务，按照DBus服务命名规则命名，保证不冲突即可。

## 系统DBus服务规范

本章节描述系统DBus服务的通用规范。包括服务命名、对象和接口的命名、组织通用规则，不描述具体的对象和接口规范，具体规范在后续章节描述。

* 基本原则是将TR098、TR104以及ITMS接口规范中的数据模型，映射为DBus 对象和接口。
* 单实例节点，映射为一个DBus 接口，参数映射为接口属性。
* 多实例节点中的父节点，不需要映射为DBus接口。
* 单实例节点的实例，映射一个DBus对象，该对象实现该节点的DBus接口，同时实现org.freedesktop.DBus.Peer、org.freedesktop.DBus.Introspectable、com.ctc.igd.Properties通用接口
* 多实例节点中的父节点，映射一个DBus 对象，该对象实现com.ctc.igd1.ObjectManager接口，对下属实例进行管理。下属每个实例按照单实例规则，都要进行映射到一个DBus对象。下属实例的Object Path格式为/com/ctc/igd1/…/<Parent Path>/<index>，其中index为实例索引。

以下以TR098 Baseline Profile为例，说明如何将数据模型映射到DBus服务。

接口一栏没有列出com.ctc.igd1.Properties、org.freedesktop.DBus.Introspectable、org.freedesktop.DBus.Peer接口，这三个接口默认都要实现的。

|  |  |  |
| --- | --- | --- |
| TR098节点名 | Object Path | DBus接口 |
| InternetGatewayDevice. | /com/ctc/igd1 | Com.ctc.igd1 |
| InternetGatewayDevice.DeviceInfo | /com/ctc/igd1/DeviceInfo | Com.ctc.igd1.DeviceInfo |
| InternetGatewayDevice.ManagementServer | /com/ctc/igd1/ManagementServer | Com.ctc.igd1.ManagementServer |
| InternetGatewayDevice.Layer3Forwarding. | /com/ctc/igd1/Layer3Forwardning | Com.ctc.igd1.Layer3Forwarding |
| InternetGatewayDevice.Layer3Forwarding.Forwarding. | /com/ctc/igd1/Layer3Forwarding/Forwarding | com.ctc.igd1.ObjectManager |
| InternetGatewayDevce.Layer3Forwarding.Forwarding.{i}. | /com/ctc/igd1/Layer3Forwarding/Forwarding/{i} | Com.ctc.igd1.Layer3ForwardingEntry |
| InternetGatewayDevice.LANConfigSecurity | /com/ctc/igd1/LANConfigSecurity | Com.ctc.igd1.LANConfigSecurity |
| InternetGatewayDevice.LANDevice. | /com/ctc/igd1/LANDevice | Com.ctc.igd1.ObjectManager |
| InternetGatewayDevice.LANDevice.{i}. | /com/ctc/igd1/LANDevice/{i} | Com.ctc.igd1.LANDevice |
| InternetGatewayDevice.LANDevice.{i}.LANHostConfigManagement. | /com/ctc/igd1/LANDevice/{i}/LANHostConfigManagement | Com.ctc.igd1.LANHostConfigManagement |
| InternetGatewayDevice.LANDevice.{i}.LANHostConfigManagement.IPInterface. | /com/ctc/igd1/LANDevice/{i}/LANHostConfigManagement/IPInterface | Com.ctc.igd1.ObjectManager |
| InternetGatewayDevice.LANDevice.{i}.LANHostConfigManagement.IPInterface.{i} | /com/ctc/igd1/LANDevice/{i}/LANHostConfigManagement/IPInterface/{i} | Com.ctc.igd1.LANConfigIPInterface |
| InternetGatewayDevice.LANDevice.{i}.Hosts | /com/ctc/igd1/LANDevice/{i}/Hosts | Com.ctc.igd1.ObjectManager |
| InternetGatewayDevice.LANDevice.{i}.Hosts.Host.{i} | /com/ctc/igd1/LANDevice/{i}/Hosts/{i} | Com.ctc.igd1.LANHost |

### 数据类型

本章节描述如何将TR069协议中的数据类型映射到DBus协议中。

|  |  |  |
| --- | --- | --- |
| TR069数据类型 | DBus数据类型 | 备注 |
| String(N) | STRING |  |
| Int | INT32 |  |
| Unsigned int | UINT32 |  |
| Boolean | BOOLEAN |  |
| dateTime | STRING | ISO 8601格式，如0001-01-01T00:00:00Z |
| Base64(N) | STRING |  |

Dbus不支持限定长度的STRING类型，本规范在定义DBus 接口时，通过注释的形式指定长度限制，厂家在实现时，需要检查参数的长度限制，超过长度限制时，返回错误org.freedesktop.DBus.Error.InvalidArgs。

### 错误码

本章节描述如何将TR069协议错误码映射到DBus错误码，DBus的标准错误码参见<http://dbus.freedesktop.org/doc/api/html/group__DBusProtocol.html>

|  |  |  |
| --- | --- | --- |
| CPE 错误码 | DBus 错误码 | DBus Error Msg |
| 9000 Method not supported | org.freedesktop.DBus.Error.NotSupported | “Method Not supported” |
| 9001 Request denied (no reason specified) | org.freedesktop.DBus.Error.AccessDenied | “Request denied” |
| 9002 Internal error | org.freedesktop.DBus.Error.Failed | “Internal error” |
| 9003 Invalid arguments | org.freedesktop.DBus.Error.InvalidArgs | “Invalid arguments” |
| 9004 Resources exceeded | org.freedesktop.DBus.Error.LimitsExceeded | “Resource exceeded” |
| 9005 Invalid parameter name | org.freedesktop.DBus.Error.InvalidArgs | “Invalid parameter name” |
| 9006 Invalid parameter type | org.freedesktop.DBus.Error.InvalidArgs | “Invalid parameter type” |
| 9007 Invalid parameter value | org.freedesktop.DBus.Error.InvalidArgs | “Invalid parameter value” |
| 9008 Attempt to set a non-writable parameter | org.freedesktop.DBus.Error.AccessDenied | “Attempt to set a non-writable parameter” |
| 9009 Notification request rejected | 无 |  |
| 9010 Download failure | 无 |  |
| 9011 Upload failure | 无 |  |
| 9012 File transfer server authentication failure | 无 |  |
| 9013 Unsupported protocol for file transfer | 无 |  |
| 9800 – 9899 | 无 |  |

其他错误码？（厂家反馈）

## App DBus服务规范

### App标准DBus服务

本章节描述App必须实现的DBus服务，这些服务被APPFramework用来管理App。包括运行时管理和配置管理。

#### 服务命名

com.<company-name>.<app-name><version>

注意这里的version由App自行管理，不必和本规范中的版本号保持一致。

#### 运行时管理服务

##### DBus接口

###### com.ctc.appframework1.AppAgent

属性

<无>

方法

* stop()

停止App

* reload()

重载 App配置

信号

无

##### DBus对象

|  |  |  |
| --- | --- | --- |
| Object Path | Interfaces | 备注 |
| /com/<company-name>/<app-name> | com.ctc.appframework1.AppAgent |  |

#### 配置管理服务

App使用UCI配置模型管理配置，App同时需要将UCI配置映射为DBus服务。

##### DBus接口

将UCI中每个Section Type映射为一个DBus接口，接口名字为：

com.<company-name>.<app-name>.Config.<section-type>

并将该Section Type中的参数映射为接口属性。

##### DBus对象

导出的DBus对象根目录为：/com/<company-name>/<app-name>/Config

按照如下规则将各个UCI Section的参数映射到DBus

|  |  |  |
| --- | --- | --- |
| Object Path | Interfaces | 备注 |
| /<section-type> | org.freedesktop.DBus.ObjectManager | 存放类型相同的所有sections |
| /<section-type>/<section-name> | org.freedesktop.DBus.Properties  com.<company-name>.<app-name>.Config.<section-type> | 存放有名section下所有的配置项 |
| /<section-type>/{i} | org.freedesktop.DBus.Properties  com.<company-name>.<app-name>.Config.<section-type> | 存放匿名section下所有的配置项，{i}为section index |

注意：上表中Object Path为相对路径

## 兼容性

* 在定义DBus服务时， 服务名、接口名、Object Path中，要求都嵌入了同样的版本号信息，比如DBus接口名com.ctc.igd1，其中1即为版本号。
* 当接口定义升级，并且无法向后兼容时，版本号+1。
* 一般情况下，接口定义升级尽量保持向后兼容，即不允许删除现有DBus接口、不允许修改现有DBus接口签名，只允许增加新的DBus接口。
* App在Manifest文件中声明依赖的服务名，比如com.ctc.igd1。当这个服务名和网关提供的服务名不匹配时，平台应拒绝网关下载App，App Framework应拒绝安装App。

当网关升级后，重新校验依赖关系，如果服务名不匹配，应删除App。

* App提供的DBus服务，也应在服务名、接口名、Object Path中嵌入版本号信息，并按照以上规则维护接口。

# TR098服务

本章描述TR098数据模型对应的DBus对象和DBus接口。

## DBus接口

### com.ctc.igd1

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| DeviceSummary | STRING(1024) |  | read | 无 |  |

#### 方法

<略>

#### 信号

<略>

### com.ctc.igd1.DeviceInfo

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| Manufacturer | STRING(64) |  | read | 无 |  |
| ManufacturerOUI | STRING(6) |  | read | 无 |  |
| ModelName | STRING(64) |  | read | 无 |  |
| Description | STRING(256) |  | read | 无 |  |
| ProductClass | STRING(64) |  | read | 无 |  |
| SerialNumber | STRING(64) |  | read | 无 |  |
| HardwareVersion | STRING(64) |  | read | 无 |  |
| SoftwareVersion | STRING(64) |  | read | 无 |  |
| ModemFirmwareVersion | STRING(64) |  | read | 无 |  |
| EnabledOptions | STRING(1024) |  | read | 无 |  |
| AdditionalHardwareVersion | STRING(64) |  | read | 无 |  |
| AdditionalSoftwareVersion | STRING(64) |  | read | 无 |  |
| SpecVersion | STRING(16) |  | read | 无 |  |
| ProvisioningCode | STRING(64) |  | read | 无 |  |
| UpTime | UINT32 |  | read | 无 |  |
| FirstUseDate | STRING(64) |  | read | 无 |  |

#### 方法

<略>

#### 信号

<略>

### com.ctc.igd1.Time

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| NTPServer1 | STRING(64) |  | readwrite | 无 |  |
| NTPServer2 | STRING(64) |  | readwrite | 无 |  |
| NTPServer3 | STRING(64) |  | readwrite | 无 |  |
| NTPServer4 | STRING(64) |  | readwrite | 无 |  |
| NTPServer5 | STRING(64) |  | readwrite | 无 |  |
| CurrentLocalTime | STRING(64) |  | read | 无 |  |
| LocalTimeZone | STRING(6) |  | readwrite | 无 |  |
| LocalTimeZoneName | STRING(64) |  | readwrite | 无 |  |
| DaylightSavingsUsed | BOOLEAN |  | readwrite | 无 |  |
| DaylightSavingsStart | STRING(64) |  | readwrite | 无 |  |
| DaylightSavingsEnd | STRING(64) |  | readwrite | 无 |  |

#### 方法

<略>

#### 信号

<略>

### com.ctc.igd1.L3Forwarding

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| DefaultConnectionService | OBJPATH |  | readwrite | 无 | 对应的WAN L3连接 对象 |

#### 方法

#### 信号

### com.ctc.igd1.L3ForwardingEntry

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| Enable | BOOLEAN |  | readwrite | 无 |  |
| Status | STRING |  | read |  |  |
| Type | STRING |  | readwrite |  |  |
| DestIPAddress | STRING |  | readwrite |  |  |
| DestSubnetMask | STRING |  | readwrite |  |  |
| SourceIPAddress | STRING |  | readwrite |  |  |
| SourceSubnetMask | STRING |  | readwrite |  |  |
| ForwardingPolicy | INT32 | [-1..] | readwrite |  |  |
| GatewayIPAddress | STRING |  | readwrite |  |  |
| Interface | OBJPATH |  | readwrite |  |  |
| ForwardingMetric | INT32 | [-1..] | readwrite |  |  |
| MTU | UINT32 | [1:1540] | readwrite |  |  |

#### 方法

<无>

#### 信号

<无>

### com.ctc.igd1.WLANConfiguration

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| Enable | BOOLEAN |  | readwrite | 无 |  |
| Status |  |  |  |  |  |
| BSSID |  |  |  |  |  |
| MaxBitRate |  |  |  |  |  |
| Channel |  |  |  |  |  |
| SSID |  |  |  |  |  |
| BeaconType |  |  |  |  |  |
| MACAddressControlEnabled |  |  |  |  |  |
| Standard |  |  |  |  |  |
|  |  |  |  |  |  |

#### 方法

#### 信号

## DBus对象

# USB设备管理服务

为了支持USB Dongle以及USB存储业务，厂家需按照本章节规范，封装DBus服务，提供给开发者，实现USB设备的控制和访问。

## DBus接口

|  |  |
| --- | --- |
| Interface Name | 备注 |
| Com.ctc.igd1. usb.Filesystem |  |
| Com.ctc.igd1. usb.Serial |  |
| Com.ctc.igd1.usb.CDC |  |
| Com.ctc.igd1.usb.HID |  |
| Com.ctc.igd1.usb.Device |  |

### Com.ctc.igd1.usb.Device

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| DeviceType | UINT16 |  | read | 无 | CtUsbDevType |
| DeviceId | INT32 |  | read | 无 |  |
| DeviceName | STRING(128) |  | read | 无 |  |

#### 方法

<无>

#### 信号

<无>

### Com.ctc.igd1.usb.FileSystem

#### 属性

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名 | 类型 | 取值范围 | 权限 | 变更通知 | 备注 |
| MountPoint | STRING |  | read | 无 |  |

#### 方法

* Mount(IN a{sv} options, OUT s mount\_path);

挂载文件系统

* Unmount{IN a{sv} options};

卸载文件系统

#### 信号

<无>

## DBus对象

|  |  |  |
| --- | --- | --- |
| Object Path | 接口 | 备注 |
| /com/ctc/igd1/usb |  |  |
| /com/ctc/igd1/usb/devices |  |  |
| /com/ctc/igd1/usb/devices/{i} |  |  |