修改记录

|  |  |  |  |
| --- | --- | --- | --- |
| 修改日期 | 修改内容 | 修改人 | 备注 |
| 2015/1/15 | 添加StartStream、StopStream协议说明 | 潘亮 | 深红 |
|  |  |  |  |

SDL HMI协议介绍

# 简介

## 简介

HMI协议是SDL与HMI进行连接、通信的协议。采用Socket(TCP/IP)通信方式，采用Json作为协议格式，协议类型包括4种：HMI请求、SDL请求、HMI通知、SDL通知。

## 整体框架

Mobile协议

HMI协议

## 通讯通道

HMI与SDL建立连接，按照模块需要创建如下几个Socket连接：

* BasicCommunication
* VehicleInfo
* VR
* Buttons
* Navigation
* TTS
* UI

具体的创建流程，请参考[HMI初始化](#_HMI初始化)。

## 通讯设置(smartDeviceLink.ini)

ServerAddress = 127.0.0.1 // SDL服务端IP地址，一般设置默认值。

ServerPort = 12346 // SDL的HMI服务端端口号。

DefaultTimeout = 10000 // 每个RPC的默认超时时间。

# 协议内容

## 协议基本结构

### 请求(Request)的结构

#### 例子

{

"id" : 9,

"jsonrpc" : "2.0",

"method" : "VR.AddCommand",

"params" : {

"appID" : 0,

"cmdID" : 2000000001,

"vrCommands" : [ "Help" ]

}

}

#### 说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| jsonrpc | String | 版本 | 是 |  |
| id | Int32 | 请求编号 | 是 | 递增数 |
| method | String | 方法名称 | 是 |  |
| params | Map | 参数枚举 | 否 | 见具体协议 |

#### 参数描述(params)

<function name="AddCommand" messagetype="request">

<description>Request from SDL to add a command(string with associated id) to VR.</description>

<param name="cmdID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="true">

<description>ID of a command (further to be used in OnCommand notification).</description>

</param>

<param name="vrCommands" type="String" mandatory="true" maxlength="99" minsize="1" maxsize="100" array="true">

<description>List of strings to be used as VR commands.</description>

</param>

<param name="appID" type="Integer" mandatory="false">

<description>ID of application that requested this RPC.</description>

</param>

</function>

### 回复(Response)的结构

#### 正常

##### 例子

{

"id":1,

"jsonrpc":"2.0",

"result":{

"available":true,

"code":0,

"method":"VR.IsReady"

}

}

##### 说明

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| jsonrpc | | String | 版本 | 是 |  |
| id | | Int32 | 请求编号 | 是 |  |
| result | | Map / Int32 / String | 结果枚举/结果数值/结果字符串 | 是 | 其它子参数见具体协议。除以MB开头的协议外，其它协议均为Map类型 |
|  | method | String | 方法名称 | 是 |  |
|  | code | Result | 结果 | 是 |  |

##### 参数描述(result)

<function name="IsReady" messagetype="response">

<param name="available" type="Boolean" mandatory="true">

<description>Must be true if VR is present and ready to communicate with SDL.</description>

</param>

</function>

#### 异常

##### 例子

{

"id":1,

"jsonrpc":"2.0",

"error":{

"available":false,

"code":0,

"method":"VR.IsReady"

"message":"is ready"

}

}

##### 说明

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 名称 | | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| jsonrpc | | String | 版本 | 是 |  |
| id | | Int32 | 请求编号 | 是 |  |
| error | | Map | 结果枚举/结果数值/结果字符串 | 是 | 其它子参数见具体协议 |
|  | method | String | 方法名称 | 是 |  |
|  | code | Result | 结果 | 是 |  |
|  | message | String | 结果说明 | 否 |  |

##### 参数描述(error)

<function name="IsReady" messagetype="response">

<param name="available" type="Boolean" mandatory="true">

<description>Must be true if VR is present and ready to communicate with SDL.</description>

</param>

</function>

### 通知(Notification)的结构

#### 例子

{

"jsonrpc" : "2.0",

"method" : "BasicCommunication.OnAppRegistered",

"params" : {

"application" : {

"appID" : 65537,

"appName" : "Sync Proxy Tester",

"deviceName" : "10.27.0.143",

"hmiDisplayLanguageDesired" : "EN-US",

"icon" : "",

"isMediaApplication" : true,

"ngnMediaScreenAppName" : "Sync Proxy Tester"

},

"ttsName" : [

{

"text" : "Sync Proxy Tester",

"type" : "TEXT"

}

],

"vrSynonyms" : [ "Sync Proxy Tester" ]

}

}

#### 说明

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| jsonrpc | String | 版本 | 是 |  |
| method | String | 方法名称 | 是 |  |
| params | Map | 参数枚举 | 否 | 见具体协议 |

#### 参数描述(params)

<function name="OnAppRegistered" messagetype="notification">

<description>Issued by SDL to notify HMI about new application registered.</description>

<param name="application" type="Common.HMIApplication" mandatory="true">

<description>The information about application registered. See HMIApplication. </description>

</param>

<param name="ttsName" type="Common.TTSChunk" minsize="1" maxsize="100" array="true" mandatory="false" >

<description> TTS string for VR recognition of the mobile application name, e.g. "Ford Drive Green".Meant to overcome any failing on speech engine in properly pronouncing / understanding app name.

Needs to be unique over all applications.

May not be empty.

May not start with a new line character.

May not interfere with any name or synonym of previously registered applications and any predefined blacklist of words (global commands)

</description>

</param>

<param name="vrSynonyms" type="String" maxlength="40" minsize="1" maxsize="100" array="true" mandatory="false">

<description>

Defines additional voice recognition commands.

May not interfere with any name or synonym of previously registered applications and any predefined blacklist of words (global commands)

</description>

</param>

</function>

## 协议列表

|  |  |
| --- | --- |
| 通讯通道 | 协议名称（红色表示新增的协议） |
| 全部通道 | [MB.registerComponent](#_MB.registerComponent) [MB.subscribeTo](#_MB.subscribeTo) |
| BasicCommunication | [UpdateDeviceList](#_BasicCommunication.UpdateDeviceList) [ActivateApp](#_BasicCommunication.ActivateApp) [OnReady](#_BasicCommunication.OnReady) [OnStartDeviceDiscovery](#_BasicCommunication.OnStartDeviceDis) [OnDeviceChosen](#_BasicCommunication.OnDeviceChosen)  [OnAppActivated](#_BasicCommunication.OnAppActivated) [OnAppDeactivated](#_BasicCommunication.OnAppDeactivated) [OnSearchAppList](#_BasicCommunication.OnSearchAppList) [OnAppRegistered](#_BasicCommunication.OnAppRegistered) [PlayTone](#_BasicCommunication.PlayTone) [OnAppUnregistered](#_BasicCommunication.OnAppUnregistere) [SDLLog](#_BasicCommunication.SDLLog) [OnFindAppList](#_BasicCommunication.OnFindAppList) |
| VehicleInfo | [IsReady](#_VehicleInfo.IsReady) [GetVehicleType](#_VehicleInfo.GetVehicleType) [ReadDID](#_VehicleInfo.ReadDID) [GetDTCs](#_VehicleInfo.GetDTCs) |
| VR | [IsReady](#_VR.IsReady) [GetLanguage](#_VR.GetLanguage) [GetSupportedLanguages](#_VR.GetSupportedLanguages) [AddCommand](#_VR.AddCommand) [DeleteCommand](#_VR.DeleteCommand) [ChangeRegistration](#_VR.ChangeRegistration) [StartRecord](#_VR.StartRecord) [CancelRecord](#_VR.CancelRecord) [VRStatus](#_VR.VRStatus) [VRCancel](#_VR.VRCancel) [VRCommandHelp](#_VR.VRCommandHelp) [VRExitApp](#_VR.VRExitApp) [VRSwitchApp](#_VR.VRSwitchApp) [VRCommandTTS](#_VR.VRCommandTTS) [VRResult](#_VR.VRResult) |
| Buttons | [GetCapabilities](#_Buttons.GetCapabilities) [OnButtonEvent](#_Buttons.OnButtonEvent) [OnButtonPress](#_Buttons.OnButtonPress) |
| Navigation | [IsReady](#_Navigation.IsReady) [ShowConstantTBT](#_Navigation.ShowConstantTBT) [AlertManeuver](#_Navigation.AlertManeuver)  [StartStream](#_Navigation.StartStream)  [StopStream](#_Navigation.StopStream) |
| TTS | [IsReady](#_TTS.IsReady) [GetLanguage](#_TTS.GetLanguage) [GetSupportedLanguages](#_TTS.GetSupportedLanguages) [Speak](#_TTS.Speak) [SetGlobalProperties](#_TTS.SetGlobalProperties) [ChangeRegistration](#_TTS.ChangeRegistration) |
| UI | [IsReady](#_UI.IsReady) [GetLanguage](#_UI.GetLanguage) [GetSupportedLanguages](#_UI.GetSupportedLanguages) [GetCapabilities](#_UI.GetCapabilities) [AddCommand](#_UI.AddCommand) [SetAppIcon](#_UI.SetAppIcon) [AddSubMenu](#_UI.AddSubMenu) [DeleteSubMenu](#_UI.DeleteSubMenu) [DeleteCommand](#_UI.DeleteCommand) [Alert](#_UI.Alert) [Show](#_UI.Show) [SetGlobalProperties](#_UI.SetGlobalProperties) [SetMediaClockTimer](#_UI.SetMediaClockTimer) [PerformInteraction](#_UI.PerformInteraction) [Slider](#_UI.Slider) [ScrollableMessage](#_UI.ScrollableMessage) [ChangeRegistration](#_UI.ChangeRegistration) [PerformAudioPassThru](#_UI.PerformAudioPassThru) [EndAudioPassThru](#_UI.EndAudioPassThru) [OnSystemContext](#_UI.OnSystemContext) [OnCommand](#_UI.OnCommand) [CreateInteractionChoiceSet](#_UI.CreateInteractionChoiceSet) [DeleteInteractionChoiceSet](#_UI.DeleteInteractionChoiceSet) [SubscribeButton](#_UI.SubscribeButton) [UnsubscribeButton](#_UI.UnsubscribeButton) |

## HMI初始化

### 注册连接

#### 描述

HMI为每个通道创建一个Socket连接，连接到SDL。连接一个通道后，调用MB.registerComponent请求注册一个通道名字，在回复中可以获得通道的编号，并且此编号为订阅方法的起始编号。

#### 序列图



#### MB.registerComponent

##### 描述

HMI请求，注册HMI连接，根据componentName建立不同的socket通道。

##### 请求(Request)

<function name="MB.registerComponent" messagetype="request">

<param name=" componentName " type="String" maxlength="50" mandatory="true" />

</function>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| componentName | String | 连接名称 | 是 | 包括： Buttons  TTS  VR  BasicCommunication  UI  VehicleInfo  Navigation |

##### 回复(Response)

<function name="MB.registerComponent" messagetype="response">

<result name=" result " type="Integer" minvalue="0" maxvalue="10000" mandatory="true" />

</function>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| result | Int32 | 连接编号 | 是 |  |

### 订阅方法

#### 描述

HMI的每个通道分别调用MB.subscribeTo请求，向SDL订阅自己的通知（HMI想要接收通知，必须先订阅通知）。订阅成功后，收到SDL的回复，包括订阅通知的编号。

#### 序列图



#### MB.subscribeTo

##### 描述

HMI请求，向SDL订阅方法，订阅后的方法，SDL才会处理。

##### 请求(Request)

<function name="MB.subscribeTo" messagetype="request">

<param name="propertyName" type="String" maxlength="100" mandatory="true" />

</function>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| propertyName | String | 方法名称 | 是 | 包括： BasicCommunication.OnAppRegistered  BasicCommunication.OnAppUnregistered  BasicCommunication.PlayTone  BasicCommunication.OnSDLClose  BasicCommunication.SDLLog  VR.VRCancel  VR.VRStatus  VR.VRCommandHelp  VR.VRCommandTTS  VR.VRExitApp  VR.VRSwitchApp  VR.VRResult |

##### 回复(Response)

<function name="MB.subscribeTo" messagetype="response">

<result name=" result " type="String" maxlength="100" mandatory="true" />

</function>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| result | String | 方法编号 | 是 |  |

### 准备初始化

#### 描述

当收到全部通道的MB.subscribeTo回复后，HMI发送BasicCommunication.OnReady请求，通知SDL，HMI已经连接完成。

#### 序列图



#### BasicCommunication.OnReady

##### 描述

HMI通知，通知SDL，HMI已准备好。

##### 通知(Notification)

<function name="OnReady" messagetype="notification">

<description>HMI must notify SDL about its readiness to start communication. In fact, this has to be the first message between SDL and HMI.</description>

</function>

### VR初始化

#### 序列图



#### VR.IsReady

##### 描述

SDL请求，VR是否准备好。

##### 请求(Request)

<function name="IsReady" messagetype="request">

<description>Method is invoked at system startup. Response provides information about presence of VR module and its readiness to cooperate with SDL.</description>

</function>

##### 回复(Response)

<function name="IsReady" messagetype="response">

<param name="available" type="Boolean" mandatory="true">

<description>Must be true if VR is present and ready to communicate with SDL.</description>

</param>

</function>

#### VR.GetLanguage

##### 描述

SDL请求，获得当前VR的语言。

##### 请求(Request)

<function name="GetLanguage" messagetype="request">

<description>Request from SmartDeviceLink to HMI to get currently active VR language</description>

</function>

##### 回复(Response)

<function name="GetLanguage" messagetype="response">

<param name="language" type="Common.Language" mandatory="true"/>

</function>

#### VR.GetSupportedLanguages

##### 描述

SDL请求，获得当前VR支持的语言。

##### 请求(Request)

<function name="GetSupportedLanguages" messagetype="request">

<description>Method is invoked at system start-up. Response must provide the information about VR supported languages.</description>

</function>

##### 回复(Response)

<function name="GetSupportedLanguages" messagetype="response">

<param name="languages" type="Common.Language" mandatory="true" array="true" minsize="1" maxsize="100">

<description>List of languages supported in VR.</description>

</param>

</function>

#### VR.AddCommand

##### 描述

SDL请求，添加一个VR命令。

##### 请求(Request)

<function name="AddCommand" messagetype="request">

<description>Request from SDL to add a command(string with associated id) to VR.</description>

<param name="cmdID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="true">

<description>ID of a command (further to be used in OnCommand notification).</description>

</param>

<param name="vrCommands" type="String" mandatory="true" maxlength="99" minsize="1" maxsize="100" array="true">

<description>List of strings to be used as VR commands.</description>

</param>

<param name="appID" type="Integer" mandatory="false">

<description>ID of application that requested this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="AddCommand" messagetype="response">

</function>

### TTS初始化

#### 序列图



#### TTS.IsReady

##### 描述

SDL请求，TTS是否准备好。

##### 请求(Request)

<function name="IsReady" messagetype="request">

<description>Method is invoked at system start-up. Response must provide the information about presence of TTS module and its readiness to cooperate with SDL.</description>

</function>

##### 回复(Response)

<function name="IsReady" messagetype="response">

<param name="available" type="Boolean" mandatory="true">

<description>Must be true if TTS is present and ready to communicate with SDL.</description>

</param>

</function>

#### TTS.GetLanguage

##### 描述

SDL请求，获得当前TTS的语言。

##### 请求(Request)

<function name="GetLanguage" messagetype="request">

<description>Request from SmartDeviceLink to HMI to get currently active TTS language</description>

</function>

##### 回复(Response)

<function name="GetLanguage" messagetype="response">

<param name="language" type="Common.Language" mandatory="true"/>

</function>

#### TTS.GetSupportedLanguages

##### 描述

SDL请求，获得当前TTS支持的语言。

##### 请求(Request)

<function name="GetSupportedLanguages" messagetype="request">

<description>Method is invoked at system start-up by SDL. Response must provide the information about TTS supported languages.</description>

</function>

##### 回复(Response)

<function name="GetSupportedLanguages" messagetype="response">

<param name="languages" type="Common.Language" mandatory="true" array="true" minsize="1" maxsize="100">

<description>List of languages supported in TTS.</description>

</param>

</function>

### UI初始化

#### 序列图



#### UI.IsReady

##### 描述

SDL请求，UI是否准备好。

##### 请求(Request)

<function name="IsReady" messagetype="request">

<description>Method is invoked at system startup. Response provides information about presence of UI module and its readiness to cooperate with SDL.</description>

</function>

##### 回复(Response)

<function name="IsReady" messagetype="response">

<param name="available" type="Boolean" mandatory="true">

<description>Must be true if UI is present and ready to communicate with SDL.</description>

</param>

</function>

#### UI.GetLanguage

##### 描述

SDL请求，获得当前UI的语言。

##### 请求(Request)

<function name="GetLanguage" messagetype="request">

<description>Request from SmartDeviceLink to HMI to get currently active UI language</description>

</function>

##### 回复(Response)

<function name="GetLanguage" messagetype="response">

<param name="language" type="Common.Language" mandatory="true"/>

</function>

#### UI.GetSupportedLanguages

##### 描述

SDL请求，获得当前UI支持的语言。

##### 请求(Request)

<function name="GetSupportedLanguages" messagetype="request">

<description>Method should be invoked at system startup. Response provides information about UI supported languages.</description>

</function>

##### 回复(Response)

<function name="GetSupportedLanguages" messagetype="response">

<param name="languages" type="Common.Language" mandatory="true" array="true" minsize="1" maxsize="100">

<description>List of languages supported in UI.</description>

</param>

</function>

#### UI.GetCapabilities

##### 描述

SDL请求，获得HMI的UI的功能。

##### 请求(Request)

<function name="GetCapabilities" messagetype="request">

<description>Method is invoked at system startup by SDL to request information about UI capabilities of HMI.</description>

</function>

##### 回复(Response)

<function name="GetCapabilities" messagetype="response">

<param name="displayCapabilities" type="Common.DisplayCapabilities" mandatory="true">

<description>Information about the capabilities of the display: its type, text field supported, etc. See DisplayCapabilities. </description>

</param>

<param name="audioPassThruCapabilities" type="Common.AudioPassThruCapabilities" mandatory="true"/>

<param name="hmiZoneCapabilities" type="Common.HmiZoneCapabilities" mandatory="true"/>

<param name="softButtonCapabilities" type="Common.SoftButtonCapabilities" mandatory="false">

<description>Must be returned if the platform supports on-screen SoftButtons.</description>

</param>

</function>

### Navigation初始化

#### 序列图



#### Navigation.IsReady

##### 描述

SDL请求，Navigation是否准备好。

##### 请求(Request)

<function name="IsReady" messagetype="request">

<description>Method is invoked at system startup. Response must provide the information about presence of UI Navigation module and its readiness to cooperate with SDL.</description>

</function>

##### 回复(Response)

<function name="IsReady" messagetype="response">

<param name="available" type="Boolean" mandatory="true">

<description>Must be true if Navigation is present and ready to communicate with SDL.</description>

</param>

</function>

### VehicleInfo初始化

#### 序列图



#### VehicleInfo.IsReady

##### 描述

SDL请求，VehicleInfo是否准备好。

##### 请求(Request)

<function name="IsReady" messagetype="request">

<description>Method is invoked at system startup. Response should provide information about presence of any of vehicle information modules (ECU, GPS, etc) and their readiness to cooperate with SDL.</description>

</function>

##### 回复(Response)

<function name="IsReady" messagetype="response">

<param name="available" type="Boolean" mandatory="true">

<description>Must be true if vehicle data modules are present and ready to communicate with SDL.</description>

</param>

</function>

#### VehicleInfo.GetVehicleType

##### 描述

SDL请求，获得Vehicle的类型。

##### 请求(Request)

<function name="GetVehicleType" messagetype="request">

<description>Request from SmartDeviceLinkCore to HMI to get info about the vehicle (type, model, etc.).</description>

</function>

##### 回复(Response)

<function name="GetVehicleType" messagetype="response">

<param name="vehicleType" type="Common.VehicleType" mandatory="true"/>

</function>

### Buttons初始化

#### 序列图



#### Buttons.GetCapabilities

##### 描述

SDL请求，获得HMI的Buttons的功能。

##### 请求(Request)

<function name="GetCapabilities" messagetype="request">

<description>Method is invoked at system start-up. SDL requests the information about all supported hardware buttons and their capabilities</description>

</function>

##### 回复(Response)

<function name="GetCapabilities" messagetype="response">

<param name="capabilities" type="Common.ButtonCapabilities" array="true" minsize="1" maxsize="100" mandatory="true">

<description>Response must provide the names of available buttons and their capabilities.See ButtonCapabilities</description>

</param>

<param name="presetBankCapabilities" type="Common.PresetBankCapabilities" mandatory="false">

<description>Must be returned if the platform supports custom on-screen Presets</description>

</param>

</function>

## SDL请求

### BasicCommunication

#### BasicCommunication.UpdateDeviceList

##### 描述

SDL请求，更新HMI的设备列表。

##### 请求(Request)

<function name="UpdateDeviceList" messagetype="request">

<description>Request from SmartDeviceLink to HMI to change device list.</description>

<param name="deviceList" type="Common.DeviceInfo" mandatory="true" array="true" minsize="0" maxsize="100">

<description>The array of names/IDs of connected devices</description>

</param>

</function>

##### 回复(Response)

<function name="UpdateDeviceList" messagetype="response">

</function>

#### BasicCommunication.ActivateApp

##### 描述

SDL请求，激活App，即将App的HMI状态改为FULL。

##### 请求(Request)

<function name="ActivateApp" messagetype="request">

<description>Request from SDL to HMI to bring specified application to front on UI e.g make it HMI status 'FULL'.</description>

<param name="appID" type="Integer" mandatory="true">

<description>ID of deactivated application.</description>

</param>

</function>

##### 回复(Response)

<function name="ActivateApp" messagetype="response">

</function>

### VR

#### VR.AddCommand

##### 描述

见[VR.AddCommand](#_VR.AddCommand)。

#### VR.DeleteCommand

##### 描述

SDL请求，删除一个VR命令。

##### 请求(Request)

<function name="DeleteCommand" messagetype="request">

<description>Request from SDL to delete a command from VR.</description>

<param name="cmdID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="true">

<description>Id of a command (list of strings), previously sent by AddCommand.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that requested this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="DeleteCommand" messagetype="response">

</function>

#### VR.ChangeRegistration

##### 描述

SDL请求，改变VR的语言。

##### 请求(Request)

<function name="ChangeRegistration" messagetype="request">

<description>Request from SmartDeviceLink to HMI to change language of VR.</description>

<param name="language" type="Common.Language" mandatory="true">

<description>The language application wants to switch to.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="ChangeRegistration" messagetype="response">

</function>

### TTS

#### TTS.Speak

##### 描述

SDL请求，说文本。

##### 请求(Request)

<function name="Speak" messagetype="request">

<description>RPC from SDL to TTS for speaking the text.</description>

<param name="ttsChunks" type="Common.TTSChunk" mandatory="true" array="true" minsize="1" maxsize="100">

<description>List of strings to be spoken.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that requested this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="Speak" messagetype="response">

<description>Provides information about success of operation.</description>

</function>

#### TTS.SetGlobalProperties

##### 描述

SDL请求，设置App的全局TTS属性。

##### 请求(Request)

<function name="SetGlobalProperties" messagetype="request">

<description>Sets some properties for the application initiated request.</description>

<param name="helpPrompt" type="Common.TTSChunk" minsize="1" maxsize="100" array="true" mandatory="false" >

<description>

The help prompt.

An array of text chunks of type TTSChunk. See TTSChunk.

The array must have at least one item.

</description>

</param>

<param name="timeoutPrompt" type="Common.TTSChunk" minsize="1" maxsize="100" array="true" mandatory="false" >

<description>

Help text for a wait timeout.

An array of text chunks of type TTSChunk. See TTSChunk.

The array must have at least one item.

</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="SetGlobalProperties" messagetype="response">

</function>

#### TTS.ChangeRegistration

##### 描述

SDL请求，改变TTS的语言。

##### 请求(Request)

<function name="ChangeRegistration" messagetype="request">

<description>Request from SmartDeviceLink to HMI to change language of TTS.</description>

<param name="language" type="Common.Language" mandatory="true">

<description>The language application wants to switch to.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="ChangeRegistration" messagetype="response">

</function>

### UI

#### UI.AddCommand

##### 描述

SDL请求，添加命令到菜单。

##### 请求(Request)

<function name="AddCommand" messagetype="request">

<description>Request from SDL to add a command to the application menu.</description>

<param name="cmdID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="true">

<description>ID of the command to be added.</description>

</param>

<param name="menuParams" type="Common.MenuParams" mandatory="false">

<description>Optional sub value containing parameters of the command (position, name, etc.). See MenuParams. If omitted the command should be added to the end of the list of commands.</description>

</param>

<param name="cmdIcon" type="Common.Image" mandatory="false">

<description>Image to be displayed for representing the command. See Image.</description>

<description>If omitted, no (or the default if applicable) icon should be displayed.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="AddCommand" messagetype="response">

</function>

#### UI.SetAppIcon

##### 描述

SDL请求，设置App的图标。

##### 请求(Request)

<function name="SetAppIcon" messagetype="request">

<description>Used to set existing local file on SYNC as the app's icon.</description>

<param name="syncFileName" type="Common.Image" mandatory="true">

<description>Either the path to the dynamic image stored on HY or the static binary image itself. See Image</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="SetAppIcon" messagetype="response">

</function>

#### UI.AddSubMenu

##### 描述

SDL请求，添加子菜单。

##### 请求(Request)

<function name="AddSubMenu" messagetype="request">

<description>Request from SDL to add a sub menu to the in-application menu.</description>

<param name="menuID" type="Integer" minvalue="1" maxvalue="2000000000" mandatory="true">

<description>ID of the sub menu to be added. Unique for the application.</description>

</param>

<param name="menuParams" type="Common.MenuParams" mandatory="true">

<description>Position and name of menu to be added. 'parent' field is omitted for this RPC. </description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that requested this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="AddSubMenu" messagetype="response">

</function>

#### UI.DeleteSubMenu

##### 描述

SDL请求，删除子菜单。

##### 请求(Request)

<function name="DeleteSubMenu" messagetype="request">

<description>Request from SDL to delete a submenu from the in-application menu.</description>

<param name="menuID" type="Integer" minvalue="1" maxvalue="2000000000" mandatory="true" >

<description>The "menuID" of the sub-menu to be deleted. (See addSubMenu.menuID)</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="DeleteSubMenu" messagetype="response">

</function>

#### UI.DeleteCommand

##### 描述

SDL请求，从菜单中删除命令。

##### 请求(Request)

<function name="DeleteCommand" messagetype="request">

<description>Request from SDL to delete a command from the in-application menu with the specified command id.</description>

<param name="cmdID" type="Integer" mandatory="true" minvalue="0" maxvalue="2000000000">

<description>cmdId previously sent via AddCommand request - id of the command to be deleted.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="DeleteCommand" messagetype="response">

</function>

#### UI.Alert

##### 描述

SDL请求，在HMI屏幕上显示一个报警消息。

##### 请求(Request)

<function name="Alert" messagetype="request">

<description>Request from SDL to show an alert message on the display.</description>

<param name="alertStrings" type="Common.TextFieldStruct" mandatory="true" array="true" minsize="0" maxsize="3">

<description>Array of lines of alert text fields. See TextFieldStruct. Uses alertText1, alertText2, alertText3.</description>

</param>

<param name="duration" type="Integer" mandatory="true" minvalue="3000" maxvalue="10000" defvalue="5000">

<description>Timeout in milliseconds.</description>

</param>

<param name="softButtons" type="Common.SoftButton" mandatory="false" minsize="0" maxsize="4" array="true">

<description>App defined SoftButtons</description>

</param>

<param name="progressIndicator" type="Boolean" mandatory="false">

<description>If supported on the given platform, the alert GUI will include some sort of animation indicating that loading of a feature is progressing. e.g. a spinning wheel or hourglass, etc.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application requested this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="Alert" messagetype="response">

<param name="tryAgainTime" type="Integer" mandatory="false" minvalue="0" maxvalue="2000000000">

<description>Amount of time (in milliseconds) that SDL must wait before resending an alert. Must be provided if another system event or overlay currently has a higher priority than this alert.</description>

</param>

</function>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| code | Result | 返回状态 | 是 | ABORTED：取消交互 TIMED\_OUT：交互超时 |

#### UI.Show

##### 描述

SDL请求，将App的相关信息显示在HMI屏幕上。

##### 请求(Request)

<function name="Show" messagetype="request">

<param name="showStrings" type="Common.TextFieldStruct" mandatory="true" array="true" minsize="0" maxsize="7">

<description>Array of lines of show text fields. See TextFieldStruct. Uses mainField1, mainField2, mainField3, mainField4. If some field is not set, the corresponding text should stay unchanged. If field's text is empty "", the field must be cleared.

mainField1: The text that should be displayed in a single or upper display line.

mainField2: The text that should be displayed on the second display line.

mainField3: The text that should be displayed on the second "page" first display line.

mainField4: The text that should be displayed on the second "page" second display line.

statusBar: statusBar.

mediaClock: Text value for MediaClock field.

mediaTrack: The text that should be displayed in the track field. This field is only valid for media applications on NGN type ACMs.

</description>

</param>

<param name="alignment" type="Common.TextAlignment" mandatory="false">

<description>Specifies how mainField1 and mainField2 texts should be aligned on the display.</description>

<description>If omitted, texts must be centered</description>

</param>

<param name="graphic" type="Common.Image" mandatory="false">

<description>Path to optional dynamic image or the static binary image itself. See Image. If omitted, the displayed graphic should not change.</description>

</param>

<param name="secondaryGraphic" type="Common.Image" mandatory="false">

<description>

Image struct determining whether static or dynamic secondary image to display in app.

If omitted on supported displays, the displayed secondary graphic shall not change.

</description>

</param>

<param name="softButtons" type="Common.SoftButton" minsize="0" maxsize="8" array="true" mandatory="false">

<description>App defined SoftButtons.</description>

<description>If omitted, the currently displayed SoftButton values should not change.</description>

</param>

<param name="customPresets" type="String" maxlength="500" minsize="0" maxsize="8" array="true" mandatory="false">

<description>App labeled on-screen presets (i.e. GEN3 media presets or dynamic search suggestions).</description>

<description>If omitted on supported displays, the presets will be shown as not defined.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>Id of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="Show" messagetype="response">

</function>

#### UI.SetGlobalProperties

##### 描述

SDL请求，设置App的全局UI属性。

##### 请求(Request)

<function name="SetGlobalProperties" messagetype="request">

<description>Request from SDL to set some properties for VR help.</description>

<param name="vrHelpTitle" type="String" maxlength="500" mandatory="false">

<description>VR Help Title text.</description>

<description>If omitted on supported displays, the default HU system help title should be used.</description>

</param>

<param name="vrHelp" type="Common.VrHelpItem" minsize="1" maxsize="100" array="true" mandatory="false">

<description>VR Help Items. If omitted on supported displays, the default HU system generated help items should be used.</description>

</param>

<param name="menuTitle" maxlength="500" type="String" mandatory="false">

<description>Optional text to label an app menu button (for certain touchscreen platforms).</description>

</param>

<param name="menuIcon" type="Common.Image" mandatory="false">

<description>>Optional icon to draw on an app menu button (for certain touchscreen platforms).</description>

</param>

<param name="keyboardProperties" type="Common.KeyboardProperties" mandatory="false">

<description>On-screen keybaord configuration (if available).</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="SetGlobalProperties" messagetype="response">

</function>

#### UI.SetMediaClockTimer

##### 描述

SDL请求，设置初始媒体时钟值和自动更新方法。

##### 请求(Request)

<function name="SetMediaClockTimer" messagetype="request">

<description>Sets the initial media clock value and automatic update method.</description>

<param name="startTime" type="Common.TimeFormat" mandatory="false">

<description>startTime should be ignored for "PAUSE", "RESUME", and "CLEAR"</description>

</param>

<param name="endTime" type="Common.TimeFormat" mandatory="false">

<description>

See TimeFormat.

endTime can be provided for "COUNTUP" and "COUNTDOWN"; to be used to calculate any visual progress bar (if not provided, this feature is ignored)

If endTime is greater then startTime for COUNTDOWN or less than startTime for COUNTUP, then the request will return an INVALID\_DATA.

endTime will be ignored for "PAUSE", "RESUME", and "CLEAR"

</description>

</param>

<param name="updateMode" type="Common.ClockUpdateMode" mandatory="true">

<description>The update method of the media clock.</description>

<description>In case of pause, resume, or clear, the start time value is ignored and shall be left out. For resume, the time continues with the same value as it was when paused.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that requested this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="SetMediaClockTimer" messagetype="response">

</function>

#### UI.PerformInteraction

##### 描述

SDL请求，触发一个交互。

##### 请求(Request)

<function name="PerformInteraction" messagetype="request">

<description>Request from SDL for triggering an interaction (e.g. "Permit GPS?" - Yes, no, Always Allow).</description>

<param name="initialText" type="Common.TextFieldStruct" mandatory="true">

<description>Uses initialInteractionText. See TextFieldStruct.</description>

</param>

<param name="choiceSet" type="Common.Choice" minsize="1" maxsize="100" array="true" mandatory="false">

<description>The list of choices to be used for the interaction with the user</description>

</param>

<param name="vrHelpTitle" type="String" maxlength="500" mandatory="false">

<description>VR Help Title text.</description>

<description>If omitted on supported displays, the default HU system help title should be used.</description>

</param>

<param name="vrHelp" type="Common.VrHelpItem" minsize="1" maxsize="100" array="true" mandatory="false">

<description>VR Help Items. If omitted on supported displays, the default HU system generated help items should be used.</description>

</param>

<param name="timeout" type="Integer" minvalue="5000" maxvalue="100000" defvalue="10000" mandatory="true">

<description>Timeout in milliseconds.</description>

</param>

<param name="interactionLayout" type="Common.LayoutMode" mandatory="false">

<description>See LayoutMode.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="PerformInteraction" messagetype="response">

<param name="choiceID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="false">

<description>ID of the choice that was selected in response to PerformInteraction.</description>

</param>

<param name="manualTextEntry" type="String" maxlength="500" mandatory="false">

<description>

Manually entered text selection, e.g. through keyboard

Can be returned in lieu of choiceID, depending on trigger source

</description>

</param>

</function>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 类型 | 描述 | 必需(是/否 ) | 注意事项 |
| code | Result | 返回状态 | 是 | ABORTED：取消交互 TIMED\_OUT：交互超时 |

#### UI.Slider

##### 描述

SDL请求，创建一个全屏或弹出的滑动条。

##### 请求(Request)

<function name="Slider" messagetype="request">

<description>Creates a full screen or pop-up overlay (depending on platform) with a single user controlled slider.</description>

<param name="numTicks" type="Integer" minvalue="2" maxvalue="26" mandatory="true">

<description>Number of selectable items on a horizontal axis</description>

</param>

<param name="position" type="Integer" minvalue="1" maxvalue="26" mandatory="true">

<description>Initial position of slider control (cannot exceed numTicks)</description>

</param>

<param name="sliderHeader" type="String" maxlength="500" mandatory="true">

<description>Text header to be displayed.</description>

</param>

<param name="sliderFooter" type="String" maxlength="500" minsize="1" maxsize="26" array="true" mandatory="false">

<description>Text footer to be displayed (meant to display min/max threshold descriptors).</description>

<description>For a static text footer, only one footer string shall be provided in the array.</description>

<description>For a dynamic text footer, the number of footer text string in the array must match the numTicks value.</description>

<description>For a dynamic text footer, text array string should correlate with potential slider position index.</description>

<description>If omitted on supported displays, no footer text shall be displayed.</description>

</param>

<param name="timeout" type="Integer" minvalue="1000" maxvalue="65535" mandatory="true">

<description>Timeout. The slider should be displayed until the defined amount of time has elapsed. </description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="Slider" messagetype="response">

<param name="sliderPosition" type="Integer" minvalue="1" maxvalue="26" mandatory="false">

<description>Current slider position. Must be returned when the user has clicked the ‘Save’ or ‘Canceled’ button or by the timeout </description>

</param>

</function>

#### UI.ScrollableMessage

##### 描述

SDL请求，创建一个全屏可滑动的格式化文本区域。

##### 请求(Request)

<function name="ScrollableMessage" messagetype="request">

<description>Creates a full screen overlay containing a large block of formatted text that can be scrolled with up to 8 SoftButtons defined</description>

<param name="messageText" type="Common.TextFieldStruct" mandatory="true">

<description>Body of text that can include newlines and tabs. Uses scrollableMessageBody.</description>

</param>

<param name="timeout" type="Integer" minvalue="0" maxvalue="65535" defvalue="30000" mandatory="true">

<description>Timeout in milliseconds. The message should be displayed until the time defined is up.</description>

</param>

<param name="softButtons" type="Common.SoftButton" minsize="0" maxsize="8" array="true" mandatory="false">

<description>App defined SoftButtons.</description>

<description>If omitted on supported displays, only the system defined "Close" SoftButton should be displayed.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="ScrollableMessage" messagetype="response">

</function>

#### UI.ChangeRegistration

##### 描述

SDL请求，改变UI的语言。

##### 请求(Request)

<function name="ChangeRegistration" messagetype="request">

<description>Request from SmartDeviceLink to HMI to change language for app.</description>

<param name="language" type="Common.Language" mandatory="true">

<description>The language application wants to switch to.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="ChangeRegistration" messagetype="response">

</function>

#### UI.PerformAudioPassThru

##### 描述

SDL请求，执行录音。

##### 请求(Request)

<function name="PerformAudioPassThru" messagetype="request">

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

<param name="audioPassThruDisplayTexts" type="Common.TextFieldStruct" mandatory="true" minsize="0" maxsize="2" array="true">

<description>Uses

audioPassThruDisplayText1: First line of text displayed during audio capture.

audioPassThruDisplayText2: Second line of text displayed during audio capture.</description>

</param>

<param name="maxDuration" type="Integer" minvalue="1" maxvalue="1000000" mandatory="true">

<description>The maximum duration of audio recording in milliseconds. If not provided, the recording should be performed until EndAudioPassThru arrives.</description>

</param>

<param name="samplingRate" type="Common.SamplingRate" mandatory="false">

<description> This value shall be allowed at 8 khz or 16 or 22 or 44 khz.</description>

</param>

<param name="bitsPerSample" type="Common.BitsPerSample" mandatory="false">

<description>Specifies the quality the audio is recorded. Currently 8 bit or 16 bit.</description>

</param>

<param name="audioType" type="Common.AudioType" mandatory="false">

<description>Specifies the type of audio data being requested.</description>

</param>

<param name="muteAudio" type="Boolean" mandatory="false">

<description>

Defines if the current audio source should be muted during the APT session. If not, the audio source will play without interruption.

If omitted, the value is set to true.

</description>

</param>

</function>

##### 回复(Response)

<function name="PerformAudioPassThru" messagetype="response">

</function>

#### UI.EndAudioPassThru

##### 描述

SDL请求，结束录音。

##### 请求(Request)

<function name="EndAudioPassThru" messagetype="request">

<description>Request is sent by SDL to stop the audio capturing.</description>

</function>

##### 回复(Response)

<function name="EndAudioPassThru" messagetype="response">

</function>

### Navigation

#### Navigation.ShowConstantTBT

##### 描述

SDL请求，显示关于Navigation的信息。

##### 请求(Request)

<function name="ShowConstantTBT" messagetype="request">

<description>Request from SmartDeviceLinkCore to HMI to show info about navigation.</description>

<param name="navigationTexts" type="Common.TextFieldStruct" mandatory="true" array="true" minsize="0" maxsize="5">

<description>See TextFieldStruct. Uses:

navigationText1

navigationText2

ETA

totalDistance

timeToDestination.

</description>

</param>

<param name="turnIcon" type="Common.Image" mandatory="false">

</param>

<param name="nextTurnIcon" type="Common.Image" mandatory="false">

</param>

<param name="distanceToManeuver" type="Float" minvalue="0" maxvalue="1000000000" mandatory="true">

<description>Fraction of distance till next maneuver (from previous maneuver).</description>

<description>May be used to calculate progress bar.</description>

</param>

<param name="distanceToManeuverScale" type="Float" minvalue="0" maxvalue="1000000000" mandatory="true">

<description>Fraction of distance till next maneuver (starting from when AlertManeuver is triggered).</description>

<description>May be used to calculate progress bar.</description>

</param>

<param name="maneuverComplete" type="Boolean" mandatory="false">

<description>If and when a maneuver has completed while an AlertManeuver is active, SDL will send this value set to TRUE in order to clear the AlertManeuver overlay.</description>

<description>If omitted the value should be assumed as FALSE.</description>

</param>

<param name="softButtons" type="Common.SoftButton" minsize="0" maxsize="3" array="true" mandatory="false">

<description>Three dynamic SoftButtons available</description>

<description>If omitted on supported displays, the currently displayed SoftButton values will not change.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="ShowConstantTBT" messagetype="response">

</function>

#### Navigation.AlertManeuver

##### 描述

SDL请求，显示导航操作。

##### 请求(Request)

<function name="AlertManeuver" messagetype="request">

<description>Request from SmartDeviceLinkCore to HMI to announce navigation maneuver</description>

<param name="softButtons" type="Common.SoftButton" minsize="0" maxsize="3" array="true" mandatory="false">

<description>If omitted, only the system defined "Close" SoftButton should be displayed.</description>

</param>

</function>

##### 回复(Response)

<function name="AlertManeuver" messagetype="response">

</function>

#### Navigation.StartStream

##### 描述

SDL请求，通知HMI开始播放VideoStream数据。

##### 请求(Request)

<function name="StartStream" messagetype="request">

<description>Notification from SmartDeviceLinkCore to HMI to start playing video streaming.</description>

<param name="url" type="String" minlength="21" maxlength="500" mandatory="true">

<description>URL that HMI start playing.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="StartStream" messagetype="response">

</function>

#### Navigation.StopStream

##### 描述

SDL请求，通知HMI停止播放VideoStream数据。

##### 请求(Request)

<function name="StopStream" messagetype="request">

<description>Notification from SmartDeviceLinkCore to HMI to start playing video streaming.</description>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="StopStream" messagetype="response">

</function>

### VehicleInfo

#### VehicleInfo.ReadDID

##### 描述

SDL请求，读取车辆诊断数据。

##### 请求(Request)

<function name="ReadDID" messagetype="request">

<description>Request from SDL for vehicle data reading.</description>

<param name="ecuName" type="Integer" minvalue="0" maxvalue="65535" mandatory="true">

<description>Name of ECU.</description>

</param>

<param name="didLocation" type="Integer" minvalue="0" maxvalue="65535" minsize="1" maxsize="1000" array="true" mandatory="true">

<description>Get raw data from vehicle data DID location(s).</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application related to this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="ReadDID" messagetype="response">

<param name="didResult" type="Common.DIDResult" minsize="0" maxsize="1000" array="true" mandatory="false">

<description>Array of requested DID results (with data if available).</description>

</param>

</function>

#### VehicleInfo.GetDTCs

##### 描述

SDL请求，获得车辆模块诊断故障代码。

##### 请求(Request)

<function name="GetDTCs" messagetype="request">

<description>Vehicle module diagnostic trouble code request.</description>

<param name="ecuName" type="Integer" minvalue="0" maxvalue="65535" mandatory="true">

<description>Name of ECU.</description>

</param>

<param name="dtcMask" type="Integer" minvalue="0" maxvalue="255" mandatory="false">

<description>DTC Mask Byte to be sent in diagnostic request to module .</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that requested this RPC.</description>

</param>

</function>

##### 回复(Response)

<function name="GetDTCs" messagetype="response">

<param name="ecuHeader" type="Integer" minvalue="0" maxvalue="65535" mandatory="true">

<description>2 byte ECU Header for DTC response (as defined in VHR\_Layout\_Specification\_DTCs.pdf)</description>

</param>

<param name="dtc" type="String" mandatory="false" minsize="1" maxsize="15" maxlength="10" array="true">

<description>

Array of all reported DTCs on module. Each DTC is represented with 4 bytes:

3 bytes for data

1 byte for status

</description>

</param>

</function>

## HMI通知

### BasicCommunication

#### BasicCommunication.OnStartDeviceDiscovery

##### 描述

HMI通知，搜索设备，用以获得连接设备的列表。

##### 通知(Notification)

<function name="OnStartDeviceDiscovery" messagetype="notification">

<description>Initiated by HMI user for getting the list of connected devices.</description>

</function>

#### BasicCommunication.OnDeviceChosen

##### 描述

HMI通知，在设备列表中选择设备，用以获得注册的App列表。

##### 通知(Notification)

<function name="OnDeviceChosen" messagetype="notification">

<description>Notification must be initiated by HMI on user selecting device in the list of devices.</description>

<param name="deviceInfo" type="Common.DeviceInfo" mandatory="true">

<description>The name and ID of the device chosen</description>

</param>

</function>

#### BasicCommunication.OnAppActivated

##### 描述

HMI通知，在注册的App列表中选择App，即激活App。

##### 通知(Notification)

<function name="OnAppActivated" messagetype="notification">

<description>Must be sent by HU system when the user clicks on app in the list of registered apps or on soft button with 'STEAL\_FOCUS' action.</description>

<param name="appID" type="Integer" mandatory="true">

<description>ID of selected application.</description>

</param>

</function>

#### BasicCommunication.OnAppDeactivated

##### 描述

HMI通知，从App中切出，即切换到另一个App、或者切换到App列表界面。

##### 通知(Notification)

<function name="OnAppDeactivated" messagetype="notification">

<description>Must be sent by HU system when the user switches to any functionality which is not other mobile application.</description>

<param name="appID" type="Integer" mandatory="true">

<description>ID of deactivated application.</description>

</param>

<param name="reason" type="Common.DeactivateReason" mandatory="true">

<description>Specifies the functionality the user has switched to.</description>

</param>

</function>

#### BasicCommunication.OnSearchAppList

##### 描述

HMI通知，搜索已经注册的全部App，用以获得全部注册的App列表。

##### 通知(Notification)

<function name="OnSearchAppList" messagetype="notification">

<description>SearchAppList</description>

</function>

### VR

#### VR.StartRecord

##### 描述

HMI通知，启动语音识别进行录音。

##### 通知(Notification)

<function name="StartRecord" messagetype="notification">

<description>Method is sent from HMI to SDL when start record.</description>

</function>

#### VR.CancelRecord

##### 描述

HMI通知，停止语音识别关闭录音。

##### 通知(Notification)

<function name="CancelRecord" messagetype="notification">

<description>Method is sent from HMI to SDL when cancel record.</description>

</function>

### UI

#### UI.OnSystemContext

##### 描述

HMI通知，改变App的上下文时发送给SDL，即切到主菜单、切换到VR、弹出界面等。

##### 通知(Notification)

<function name="OnSystemContext" messagetype="notification">

<description>Notification must be initiated by HMI when the user changes the context of application: goes to menu (in-application menu or system menu); switches to VR; pop-up appears on screen etc.</description>

<param name="systemContext" type="Common.SystemContext" mandatory="true">

<description>The context the application is brought into.</description>

</param>

</function>

#### UI.OnCommand

##### 描述

HMI通知，在菜单中选择一个命令时触发。

##### 通知(Notification)

<function name="OnCommand" messagetype="notification">

<description>Notification must be initiated by HMI on user choosing menu item.</description>

<param name="cmdID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="true">

<description>Command ID, which is related to a specific menu entry (previously sent with AddCommand).</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that is related to this RPC.</description>

</param>

</function>

### Buttons

#### Buttons.OnButtonEvent

##### 描述

HMI通知，按钮事件，即按钮按下、或松开时分别触发。

##### 通知(Notification)

<function name="OnButtonEvent" messagetype="notification">

<description>HU system must notify about every UP/DOWN event for buttons</description>

<param name="name" type="Common.ButtonName" mandatory="true"/>

<param name="mode" type="Common.ButtonEventMode" mandatory="true">

<description>Indicates whether this is an UP or DOWN event.</description>

</param>

<param name="customButtonID" type="Integer" mandatory="false" minvalue="0" maxvalue="65536">

<description>Must be provided if ButtonName is CUSTOM\_BUTTON, this references the integer ID passed by a custom button. (e.g. softButtonName)</description>

</param>

</function>

#### Buttons.OnButtonPress

##### 描述

HMI通知，按钮事件，即长按、或短按按钮时触发。

##### 通知(Notification)

<function name="OnButtonPress" messagetype="notification">

<param name="name" type="Common.ButtonName" mandatory="true"/>

<param name="mode" type="Common.ButtonPressMode" mandatory="true">

<description>Indicates whether this is a LONG or SHORT button press event.</description>

</param>

<param name="customButtonID" type="Integer" mandatory="false" minvalue="0" maxvalue="65536">

<description>Must be returned if ButtonName is CUSTOM\_BUTTON, this references the string passed by a custom button. (e.g. softButtonName)</description>

</param>

</function>

## SDL通知

### BasicCommunication

#### BasicCommunication.OnAppRegistered

##### 描述

SDL通知，通知HMI，有一个新的App注册。

##### 通知(Notification)

<function name="OnAppRegistered" messagetype="notification">

<description>Issued by SDL to notify HMI about new application registered.</description>

<param name="application" type="Common.HMIApplication" mandatory="true">

<description>The information about application registered. See HMIApplication. </description>

</param>

<param name="ttsName" type="Common.TTSChunk" minsize="1" maxsize="100" array="true" mandatory="false" >

<description> TTS string for VR recognition of the mobile application name, e.g. "Ford Drive Green".Meant to overcome any failing on speech engine in properly pronouncing / understanding app name.

Needs to be unique over all applications.

May not be empty.

May not start with a new line character.

May not interfere with any name or synonym of previously registered applications and any predefined blacklist of words (global commands)

</description>

</param>

<param name="vrSynonyms" type="String" maxlength="40" minsize="1" maxsize="100" array="true" mandatory="false">

<description>

Defines additional voice recognition commands.

May not interfere with any name or synonym of previously registered applications and any predefined blacklist of words (global commands)

</description>

</param>

</function>

#### BasicCommunication.PlayTone

##### 描述

SDL通知，通知HMI，有声音被播放。

##### 通知(Notification)

<function name="PlayTone" messagetype="notification">

<description>Sent by SDL to HMI to notify that the tone should be played.</description>

</function>

#### BasicCommunication.OnAppUnregistered

##### 描述

SDL通知，通知HMI，有一个App注销。

##### 通知(Notification)

<function name="OnAppUnregistered" messagetype="notification">

<description>Issued by SDL to notify HMI about application unregistered. Application then to be removed from application list; all data connected with application has to be cleared up.</description>

<param name="appID" type="Integer" mandatory="true">

<description>ID of the application unregistered</description>

</param>

</function>

#### BasicCommunication.SDLLog

##### 描述

SDL通知，向HMI发送SDL与设备App的通信协议文本，即RPC的json字符串。

##### 通知(Notification)

<function name="SDLLog" messagetype="notification">

<description>Method is sent from SDL to HMI when the mobile RPC invoked.</description>

<param name="app\_id" type="Integer" mandatory="true" />

<param name="correlation\_id" type="Integer" mandatory="true" />

<param name="function" type="String" maxlength="80" mandatory="true" />

<param name="type" type="String" maxlength="20" mandatory="true" />

<param name="data" type="String" maxlength="10000" mandatory="true" />

</function>

#### BasicCommunication.OnFindAppList

##### 描述

SDL通知，向HMI发送搜索到的App列表，与[BasicCommunication.OnSearchAppList](#_BasicCommunication.OnSearchAppList)对应。

##### 通知(Notification)

<function name="OnFindAppList" messagetype="notification">

<description>SearchAppList</description>

<param name="applications" type="Common.HMIApplication" mandatory="true" array="true" minsize="1" maxsize="100"/>

</function>

### VR

#### VR.VRStatus

##### 描述

SDL通知，当VR状态改变时通知HMI。

##### 通知(Notification)

<function name="VRStatus" messagetype="notification">

<description>Method is sent from SDL to HMI when VR's status is changed.</description>

<param name="status" type="Common.VRStatus" mandatory="true">

<description>Types of status of VR.</description>

</param>

</function>

#### VR.VRCancel

##### 描述

SDL通知，当VR识别为取消时通知HMI。

##### 通知(Notification)

<function name="VRCancel" messagetype="notification">

<description>Method is sent from SDL to HMI when VR is cancel.</description>

</function>

#### VR.VRCommandHelp

##### 描述

SDL通知，当VR识别为帮助时通知HMI。

##### 通知(Notification)

<function name="VRCommandHelp" messagetype="notification">

<description>Method is sent from SDL to HMI when VR is help.</description>

<param name="vrContent" type="String" maxlength="100" mandatory="true">

<description>VR content.</description>

</param>

</function>

#### VR.VRExitApp

##### 描述

SDL通知，当VR识别为退出时通知HMI。

##### 通知(Notification)

<function name="VRExitApp" messagetype="notification">

<description>Method is sent from SDL to HMI when VR is exit app.</description>

</function>

#### VR.VRSwitchApp

##### 描述

SDL通知，当VR识别为切换App时通知HMI。

##### 通知(Notification)

<function name="VRSwitchApp" messagetype="notification">

<description>Method is sent from SDL to HMI when VR is selecting other app.</description>

<param name="appID" type="Integer" mandatory="true">

<description>ID of application that concerns this RPC.</description>

</param>

<param name="appName" type="String" maxlength="100" mandatory="false">

<description>The mobile application name, e.g. "Ford Drive Green".</description>

</param>

<param name="appVRName" type="String" maxlength="100" mandatory="true">

<description>app vr content.</description>

</param>

</function>

#### VR.VRCommandTTS

##### 描述

SDL通知，返回VR帮助的TTS到HMI。

##### 通知(Notification)

<function name="VRCommandTTS" messagetype="notification">

<description>Method is sent from SDL to HMI when VR is help.</description>

<param name="vrCommands" type="String" mandatory="true" maxlength="99" minsize="1" maxsize="1000" array="true">

<description>List of strings to be used as VR commands.</description>

</param>

</function>

#### VR.VRResult

##### 描述

SDL通知，返回VR命令到HMI。

##### 通知(Notification)

<function name="VRResult" messagetype="notification">

<description>Method is sent from SDL to HMI when VR command is sent.</description>

<param name="cmdID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="true">

<description>ID of the command to be sent.</description>

</param>

<param name="vrName" type="String" maxlength="100" mandatory="true">

<description>vr content.</description>

</param>

</function>

### UI

#### UI.CreateInteractionChoiceSet

##### 描述

SDL通知，将创建交互的信息发送给HMI。

##### 通知(Notification)

<function name="CreateInteractionChoiceSet" messagetype="notification">

<description>CreateInteractionChoiceSet</description>

<param name="interactionChoiceSetID" type="Integer" minvalue="0" maxvalue="2000000000">

<description>Unique ID used for this interaction choice set.</description>

</param>

<param name="choiceSet" type="Common.Choice" minsize="1" maxsize="100" array="true" />

</function>

#### UI.DeleteInteractionChoiceSet

##### 描述

SDL通知，将删除交互的信息发送给HMI。

##### 通知(Notification)

<function name="DeleteInteractionChoiceSet" messagetype="notification">

<description>DeleteInteractionChoiceSet</description>

<param name="interactionChoiceSetID" type="Integer" minvalue="0" maxvalue="2000000000">

<description>ID of the interaction choice set to delete.</description>

</param>

</function>

#### UI.SubscribeButton

##### 描述

SDL通知，将订阅按钮的信息发送给HMI。

##### 通知(Notification)

<function name="SubscribeButton" messagetype="notification">

<description>SubscribeButton</description>

<param name="buttonName" type="Common.ButtonName">

<description>Name of the button to subscribe.</description>

</param>

</function>

#### UI.UnsubscribeButton

##### 描述

SDL通知，将取消订阅按钮的信息发送给HMI。

##### 通知(Notification)

<function name="UnsubscribeButton" messagetype="notification">

<description>UnsubscribeButton</description>

<param name="buttonName" type="Common.ButtonName">

<description>Name of the button to unsubscribe.</description>

</param>

</function>

## 枚举

<enum name="Result">

<element name="SUCCESS" value="0"/>

<element name="UNSUPPORTED\_REQUEST" value="1"/>

<element name="UNSUPPORTED\_RESOURCE" value="2"/>

<element name="DISALLOWED" value="3"/>

<element name="REJECTED" value="4"/>

<element name="ABORTED" value="5"/>

<element name="IGNORED" value="6"/>

<element name="RETRY" value="7"/>

<element name="IN\_USE" value="8"/>

<element name="DATA\_NOT\_AVAILABLE" value="9"/>

<element name="TIMED\_OUT" value="10"/>

<element name="INVALID\_DATA" value="11"/>

<element name="CHAR\_LIMIT\_EXCEEDED" value="12"/>

<element name="INVALID\_ID" value="13"/>

<element name="DUPLICATE\_NAME" value="14"/>

<element name="APPLICATION\_NOT\_REGISTERED" value="15"/>

<element name="WRONG\_LANGUAGE" value="16"/>

<element name="OUT\_OF\_MEMORY" value="17"/>

<element name="TOO\_MANY\_PENDING\_REQUESTS" value="18"/>

<element name="NO\_APPS\_REGISTERED" value="19"/>

<element name="NO\_DEVICES\_CONNECTED" value="20"/>

<element name="WARNINGS" value="21"/>

<element name="GENERIC\_ERROR" value="22"/>

<element name="USER\_DISALLOWED" value="23"/>

</enum>

<enum name="ButtonName">

<description>Defines the hard (physical) and soft (touchscreen) buttons available from SYNC</description>

<element name="OK"/>

<element name="SEEKLEFT"/>

<element name="SEEKRIGHT"/>

<element name="TUNEUP"/>

<element name="TUNEDOWN"/>

<element name="PRESET\_0"/>

<element name="PRESET\_1"/>

<element name="PRESET\_2"/>

<element name="PRESET\_3"/>

<element name="PRESET\_4"/>

<element name="PRESET\_5"/>

<element name="PRESET\_6"/>

<element name="PRESET\_7"/>

<element name="PRESET\_8"/>

<element name="PRESET\_9"/>

<element name="CUSTOM\_BUTTON"/>

<element name="SEARCH"/>

</enum>

<enum name="ButtonEventMode">

<element name="BUTTONUP">

<description>A button has been released up</description>

</element>

<element name="BUTTONDOWN">

<description>A button has been pressed down</description>

</element>

</enum>

<enum name="ButtonPressMode">

<element name="LONG">

<description>A button was released, after it was pressed for a long time. Actual timing is defined by head unit and may vary</description>

</element>

<element name="SHORT">

<description>A button was released, after it was pressed for a short time. Actual timing is defined by head unit and may vary</description>

</element>

</enum>

<enum name="Language">

<element name="EN-US" internal\_name="EN\_US">

<description>English - US</description>

</element>

<element name="ES-MX" internal\_name="ES\_MX">

<description>Spanish - Mexico</description>

</element>

<element name="FR-CA" internal\_name="FR\_CA">

<description>French - Canada</description>

</element>

<element name="DE-DE" internal\_name="DE\_DE">

<description>German - Germany</description>

</element>

<element name="ES-ES" internal\_name="ES\_ES">

<description>Spanish - Spain</description>

</element>

<element name="EN-GB" internal\_name="EN\_GB">

<description>English - GB</description>

</element>

<element name="RU-RU" internal\_name="RU\_RU">

<description>Russian - Russia</description>

</element>

<element name="TR-TR" internal\_name="TR\_TR">

<description>Turkish - Turkey</description>

</element>

<element name="PL-PL" internal\_name="PL\_PL">

<description>Polish - Poland</description>

</element>

<element name="FR-FR" internal\_name="FR\_FR">

<description>French - France</description>

</element>

<element name="IT-IT" internal\_name="IT\_IT">

<description>Italian - Italy</description>

</element>

<element name="SV-SE" internal\_name="SV\_SE">

<description>Swedish - Sweden</description>

</element>

<element name="PT-PT" internal\_name="PT\_PT">

<description>Portuguese - Portugal</description>

</element>

<element name="NL-NL" internal\_name="NL\_NL">

<description>Dutch (Standard) - Netherlands</description>

</element>

<element name="EN-AU" internal\_name="EN\_AU">

<description>English - Australia</description>

</element>

<element name="ZH-CN" internal\_name="ZH\_CN">

<description>Mandarin - China</description>

</element>

<element name="ZH-TW" internal\_name="ZH\_TW">

<description>Mandarin - Taiwan</description>

</element>

<element name="JA-JP" internal\_name="JA\_JP">

<description>Japanese - Japan</description>

</element>

<element name="AR-SA" internal\_name="AR\_SA">

<description>Arabic - Saudi Arabia</description>

</element>

<element name="KO-KR" internal\_name="KO\_KR">

<description>Korean - South Korea</description>

</element>

<element name="PT-BR" internal\_name="PT\_BR">

<description>Portuguese - Brazil</description>

</element>

<element name="CS-CZ" internal\_name="CS\_CZ">

<description>Czech - Czech Republic</description>

</element>

<element name="DA-DK" internal\_name="DA\_DK">

<description>Danish - Denmark</description>

</element>

<element name="NO-NO" internal\_name="NO\_NO">

<description>Norwegian - Norway</description>

</element>

</enum>

<enum name="SoftButtonType">

<description>Contains information about the SoftButton capabilities.</description>

<element name="TEXT" internal\_name="SBT\_TEXT"/>

<element name="IMAGE" internal\_name="SBT\_IMAGE"/>

<element name="BOTH" internal\_name="SBT\_BOTH"/>

</enum>

<enum name="SystemAction">

<description>Enumeration that describes system actions that can be triggered.</description>

<element name="DEFAULT\_ACTION">

<description>Default action occurs. Standard behavior (e.g. SoftButton clears overlay).</description>

</element>

<element name="STEAL\_FOCUS">

<description>The calling app's dialog or related event should clear and the app should be brought into HMI\_FULL.</description>

</element>

<element name="KEEP\_CONTEXT">

<description>Current system context is maintained. An overlay is persisted even though a SoftButton has been pressed and the notification sent.</description>

</element>

</enum>

<enum name="AppHMIType">

<description>Enumeration listing possible app types.</description>

<element name="DEFAULT" />

<element name="COMMUNICATION" />

<element name="MEDIA" />

<element name="MESSAGING" />

<element name="NAVIGATION" />

<element name="INFORMATION" />

<element name="SOCIAL" />

<element name="BACKGROUND\_PROCESS" />

<element name="TESTING" />

<element name="SYSTEM" />

</enum>

<enum name="LayoutMode">

<description>For touchscreen interactions, the mode of how the choices are presented.</description>

<element name="ICON\_ONLY" />

<description>This mode causes the interaction to display the previous set of choices as icons.</description>

<element name="ICON\_WITH\_SEARCH" />

<description>This mode causes the interaction to display the previous set of choices as icons along with a search field in the HMI.</description>

<element name="LIST\_ONLY" />

<description>This mode causes the interaction to display the previous set of choices as a list.</description>

<element name="LIST\_WITH\_SEARCH" />

<description>This mode causes the interaction to display the previous set of choices as a list along with a search field in the HMI.</description>

<element name="KEYBOARD" />

<description>This mode causes the interaction to immediately display a keyboard entry through the HMI.</description>

</enum>

<enum name="DeactivateReason">

<description>Specifies the functionality the User has switched to.</description>

<element name="AUDIO">

<description>Navigated to audio(radio, etc)</description>

</element>

<element name="PHONECALL">

<description>Navigated to make a call.</description>

</element>

<element name="NAVIGATIONMAP">

<description>Navigated to navigation screen.</description>

</element>

<element name="PHONEMENU">

<description>Navigated to phone menu.</description>

</element>

<element name="SYNCSETTINGS">

<description>Navigated to settings menu.</description>

</element>

<element name="GENERAL">

<description>Other screens navigation apart from other mobile app.</description>

</element>

</enum>

<enum name="ClockUpdateMode">

<description>Describes how the media clock timer should behave on the platform</description>

<element name="COUNTUP" />

<description>Starts the media clock timer counting upwards, as in time elapsed.</description>

<element name="COUNTDOWN" />

<description>Starts the media clock timer counting downwards, as in time remaining.</description>

<element name="PAUSE" />

<description>Pauses the media clock timer</description>

<element name="RESUME" />

<description>Resume the media clock timer</description>

<element name="CLEAR" />

<description>Clears the media clock timer (previously done through Show->mediaClock)</description>

</enum>

<enum name="SystemContext">

<description>Enumeration that describes possible contexts the application might be in on HU.</description>

<description>Communicated to whichever app is in HMI FULL, except Alert.</description>

<element name="MAIN" internal\_name="SYSCTXT\_MAIN">

<description>The app's persistent display (whether media/non-media/navigation) is fully visible onscreen.</description>

<description> There is currently no user interaction (user-initiated or app-initiated) with the head-unit</description>

</element>

<element name="VRSESSION" internal\_name="SYSCTXT\_VRSESSION">

<description>The system is currently in a VR session (with whatever dedicated VR screen being overlaid onscreen).</description>

</element>

<element name="MENU" internal\_name="SYSCTXT\_MENU">

<description>The system is currently displaying a system or in-App menu onscreen.</description>

</element>

<element name="HMI\_OBSCURED" internal\_name="SYSCTXT\_HMI\_OBSCURED">

<description>The app's display HMI is currently obscuring with either a system or other app's overlay (except of Alert element).</description>

</element>

<element name="ALERT" internal\_name="SYSCTXT\_ALERT">

<description>Broadcast only to whichever app has an alert currently being displayed.</description>

</element>

</enum>

<enum name="HmiZoneCapabilities">

<description>Contains information about the HMI zone capabilities.</description>

<description>For future use.</description>

<element name="FRONT" />

<element name="BACK" />

</enum>

<enum name="SpeechCapabilities">

<description>Contains information about the TTS capabilities.</description>

<element name="TEXT" internal\_name="SC\_TEXT" />

<element name="SAPI\_PHONEMES" />

<element name="LHPLUS\_PHONEMES" />

<element name="PRE\_RECORDED" />

<element name="SILENCE" />

</enum>

<enum name="VrCapabilities">

<description>Contains information about the VR capabilities.</description>

<element name="TEXT" internal\_name="VR\_TEXT" />

</enum>

<enum name="PrerecordedSpeech">

<description>Contains a list of prerecorded speech items present on the platform.</description>

<element name="HELP\_JINGLE" />

<element name="INITIAL\_JINGLE" />

<element name="LISTEN\_JINGLE" />

<element name="POSITIVE\_JINGLE" />

<element name="NEGATIVE\_JINGLE" />

</enum>

<enum name="TBTState">

<description>Enumeration that describes possible states of turn-by-turn client or SmartDeviceLink app.</description>

<element name="ROUTE\_UPDATE\_REQUEST" />

<element name="ROUTE\_ACCEPTED" />

<element name="ROUTE\_REFUSED" />

<element name="ROUTE\_CANCELLED" />

<element name="ETA\_REQUEST" />

<element name="NEXT\_TURN\_REQUEST" />

<element name="ROUTE\_STATUS\_REQUEST" />

<element name="ROUTE\_SUMMARY\_REQUEST" />

<element name="TRIP\_STATUS\_REQUEST" />

<element name="ROUTE\_UPDATE\_REQUEST\_TIMEOUT" />

</enum>

<enum name="ApplicationsCloseReason">

<description>Describes the reasons for exiting all of applications.</description>

<element name="IGNITION\_OFF" />

<element name="MASTER\_RESET" />

<element name="FACTORY\_DEFAULTS" />

</enum>

<enum name="DisplayType">

<description>The possible types of HU display.</description>

<element name="CID">

<description> A 2-line x 20 character "dot matrix" display </description>

</element>

<element name="TYPE2">

<description> 1 line older radio head unit.</description>

</element>

<element name="TYPE5">

<description> Old radio head unit.</description>

</element>

<element name="NGN">

<description> Next Generation Navigation display.</description>

</element>

<element name="GEN2\_8\_DMA">

<description> GEN-2, 8 inch display.</description>

</element>

<element name="GEN2\_6\_DMA">

<description> GEN-2, 6 inch display.</description>

</element>

<element name="MFD3">

<description> 3 inch GEN1.1 display </description>

</element>

<element name="MFD4">

<description> 4 inch GEN1.1 display </description>

</element>

<element name="MFD5">

<description> 5 inch GEN1.1 display </description>

</element>

<element name="GEN3\_8-INCH" internal\_name="GEN3\_8\_INCH" />

</enum>

<enum name="ImageType">

<description>Contains information about the type of image.</description>

<element name="STATIC" />

<element name="DYNAMIC" />

</enum>

<enum name="TextFieldName">

<element name="mainField1">

<description>The first line of first set of main fields of the persistent display; applies to "Show"</description>

</element>

<element name="mainField2">

<description>The second line of first set of main fields of the persistent display; applies to "Show"</description>

</element>

<element name="mainField3">

<description>The first line of second set of main fields of persistent display; applies to "Show"</description>

</element>

<element name="mainField4">

<description>The second line of second set of main fields of the persistent display; applies to "Show"</description>

</element>

<element name="statusBar">

<description>The status bar on NGN; applies to "Show"</description>

</element>

<element name="mediaClock">

<description>Text value for MediaClock field; applies to "Show"</description>

</element>

<element name="mediaTrack">

<description>The track field of NGN and GEN1.1 MFD displays. This field is only available for media applications; applies to "Show"</description>

</element>

<element name="alertText1">

<description>The first line of the alert text field; applies to "Alert"</description>

</element>

<element name="alertText2">

<description>The second line of the alert text field; applies to "Alert"</description>

</element>

<element name="alertText3">

<description>The third line of the alert text field; applies to "Alert"</description>

</element>

<element name="scrollableMessageBody">

<description>Long form body of text that can include newlines and tabs; applies to "ScrollableMessage"</description>

</element>

<element name="initialInteractionText">

<description> First line suggestion for a user response (in the case of VR enabled interaction</description>

</element>

<element name="navigationText1">

<description> First line of navigation text</description>

</element>

<element name="navigationText2">

<description> Second line of navigation text</description>

</element>

<element name="ETA">

<description> Estimated Time of Arrival time for navigation</description>

</element>

<element name="totalDistance">

<description> Total distance to destination for navigation</description>

</element>

<element name="navigationText">

<description>Navigation text for UpdateTurnList.</description>

</element>

<element name="audioPassThruDisplayText1">

<description> First line of text for audio pass thru</description>

</element>

<element name="audioPassThruDisplayText2">

<description> Second line of text for audio pass thru</description>

</element>

<element name="sliderHeader">

<description> Header text for slider</description>

</element>

<element name="sliderFooter">

<description> Footer text for slider</description>

</element>

<element name="notificationText">

<description>Text of notification to be displayed on screen.</description>

</element>

<element name="menuName">

<description> Primary text for Choice</description>

</element>

<element name="secondaryText">

<description> Secondary text for Choice</description>

</element>

<element name="tertiaryText">

<description> Tertiary text for Choice</description>

</element>

<element name="timeToDestination"/>

<!-- TO DO to be removed -->

<element name="turnText"/>

</enum>

<enum name="ImageFieldName">

<element name="softButtonImage">

<description>The image field for SoftButton</description>

</element>

<element name="choiceImage">

<description>The first image field for Choice</description>

</element>

<element name="choiceSecondaryImage">

<description>The secondary image field for Choice</description>

</element>

<element name="vrHelpItem">

<description>The image field for vrHelpItem</description>

</element>

<element name="turnIcon">

<description>The image field for Turn</description>

</element>

<element name="menuIcon">

<description>The image field for the menu icon in SetGlobalProperties</description>

</element>

<element name="cmdIcon">

<description>The image field for AddCommand</description>

</element>

<element name="appIcon">

<description>The image field for the app icon (set by setAppIcon)</description>

</element>

<element name="graphic">

<description>The image field for Show</description>

</element>

<element name="showConstantTBTIcon">

<description>The primary image field for ShowConstantTBT</description>

</element>

<element name="showConstantTBTNextTurnIcon">

<description>The secondary image field for ShowConstantTBT</description>

</element>

</enum>

<enum name="TextAlignment">

<description>The list of possible alignments, left, right, or centered</description>

<element name="LEFT\_ALIGNED" />

<element name="RIGHT\_ALIGNED" />

<element name="CENTERED" />

</enum>

<enum name="DriverDistractionState">

<description>Enumeration that describes possible states of driver distraction.</description>

<element name="DD\_ON" />

<element name="DD\_OFF" />

</enum>

<enum name="MediaClockFormat">

<element name="CLOCK1">

<description>

minutesFieldWidth = 2;minutesFieldMax = 19;secondsFieldWidth = 2;secondsFieldMax = 99;maxHours = 19;maxMinutes = 59;maxSeconds = 59;

Is used for Type II, NGN and CID head units.

</description>

</element>

<element name="CLOCK2">

<description>

minutesFieldWidth = 3;minutesFieldMax = 199;secondsFieldWidth = 2;secondsFieldMax = 99;maxHours = 59;maxMinutes = 59;maxSeconds = 59;

Is used for Type V head units.

</description>

</element>

<element name="CLOCK3">

<description>

minutesFieldWidth = 2;minutesFieldMax = 59;secondsFieldWidth = 2;secondsFieldMax = 59;maxHours = 9;maxMinutes = 59;maxSeconds = 59;

Is used for GEN1.1 (i.e. MFD3/4/5) head units.

</description>

</element>

<element name="CLOCKTEXT1">

<description>

5 characters possible

Format: 1|sp c :|sp c c

1|sp : digit "1" or space

c : character out of following character set: sp|0-9|[letters

:|sp : colon or space

Is used for Type II head unit

</description>

</element>

<element name="CLOCKTEXT2">

<description>

5 chars possible

Format: 1|sp c :|sp c c

1|sp : digit "1" or space

c : character out of following character set: sp|0-9|[letters

:|sp : colon or space

Is used for CID and NGN head unit

</description>

</element>

<element name="CLOCKTEXT3">

<description>

6 chars possible

Format: 1|sp c c :|sp c c

1|sp : digit "1" or space

c : character out of following character set: sp|0-9|[letters

:|sp : colon or space

Is used for Type V head unit

</description>

</element>

<element name="CLOCKTEXT4">

<description>

6 chars possible

Format: c :|sp c c : c c

:|sp : colon or space

c : character out of following character set: sp|0-9|[letters].

Is used for GEN1.1 (i.e. MFD3/4/5) head units

</description>

</element>

</enum>

<!--IVI part-->

<enum name="ComponentVolumeStatus">

<description>The volume status of a vehicle component.</description>

<element name="UNKNOWN" internal\_name="CVS\_UNKNOWN">

<description> The data is unknown.</description>

</element>

<element name="NORMAL" internal\_name="CVS\_NORMAL">

<description> The volume is normal.</description>

</element>

<element name="LOW" internal\_name="CVS\_LOW">

<description> The volume is low.</description>

</element>

<element name="FAULT" internal\_name="CVS\_FAULT">

<description> The module/sensor is currently faulted. </description>

</element>

<element name="ALERT" internal\_name="CVS\_ALERT">

<description> The component`s volume is in critical level.</description>

</element>

<element name="NOT\_SUPPORTED" internal\_name="CVS\_NOT\_SUPPORTED">

<description> The data is not supported.</description>

</element>

</enum>

<enum name="PRNDL">

<description>The selected gear.</description>

<element name="PARK">

<description>Parking</description>

</element>

<element name="REVERSE">

<description>Reverse gear</description>

</element>

<element name="NEUTRAL">

<description>No gear</description>

</element>

<element name="DRIVE">

</element>

<element name="SPORT">

<description>Drive Sport mode</description>

</element>

<element name="LOWGEAR">

<description>1st gear hold</description>

</element>

<element name="FIRST">

</element>

<element name="SECOND">

</element>

<element name="THIRD">

</element>

<element name="FOURTH">

</element>

<element name="FIFTH">

</element>

<element name="SIXTH">

</element>

<element name="SEVENTH">

</element>

<element name="EIGHTH">

</element>

<element name="FAULT">

</element>

</enum>

<enum name="VehicleDataEventStatus">

<description>Reflects the status of a vehicle data event; e.g. a seat belt event status.</description>

<element name="NO\_EVENT" internal\_name="VDES\_NO\_EVENT">

<description> The system does not have the adequate information to send valid YES or NO states.</description>

</element>

<element name="NO" internal\_name="VDES\_NO">

<description> The requested event is in NO state.</description>

</element>

<element name="YES" internal\_name="VDES\_YES">

<description> The requested event is in YES state.</description>

</element>

<element name="NOT\_SUPPORTED" internal\_name="VDES\_NOT\_SUPPORTED">

<description> The requested data is not supported </description>

</element>

<element name="FAULT" internal\_name="VDES\_FAULT">

<description> The module/sensor is currently faulted.</description>

</element>

</enum>

<enum name="VehicleDataStatus">

<description>Reflects the status of a binary vehicle data item.</description>

<element name="NO\_DATA\_EXISTS" internal\_name="VDS\_NO\_DATA\_EXISTS">

</element>

<element name="OFF" internal\_name="VDS\_OFF">

</element>

<element name="ON" internal\_name="VDS\_ON">

</element>

</enum>

<enum name="IgnitionStableStatus">

<description>Reflects the ignition switch stability.</description>

<element name="IGNITION\_SWITCH\_NOT\_STABLE">

</element>

<element name="IGNITION\_SWITCH\_STABLE">

</element>

<element name="MISSING\_FROM\_TRANSMITTER">

<description> Either the data is not accessible or the sensor is broken.</description>

</element>

</enum>

<enum name="IgnitionStatus">

<description>Reflects the status of ignition.</description>

<element name="UNKNOWN" internal\_name="IS\_UNKNOWN">

<description> The information is not acceptable. </description>

</element>

<element name="OFF" internal\_name="IS\_OFF">

<description> The ignition is off.</description>

</element>

<element name="ACCESSORY" internal\_name="IS\_ACCESSORY">

<description> The accessories are active (power windows, audio, display, etc.).</description>

</element>

<element name="RUN" internal\_name="IS\_RUN">

<description> Ignition is active.</description>

</element>

<element name="START" internal\_name="IS\_START">

<description> Starter is switched.</description>

</element>

<element name="INVALID" internal\_name="IS\_INVALID">

<description> The data is provided, but there is some sort of fault or problem.</description>

</element>

</enum>

<enum name="DeviceLevelStatus">

<description>Reflects the reported component status of the connected device, if reported.</description>

<element name="ZERO\_LEVEL\_BARS">

</element>

<element name="ONE\_LEVEL\_BARS">

</element>

<element name="TWO\_LEVEL\_BARS">

</element>

<element name="THREE\_LEVEL\_BARS">

</element>

<element name="FOUR\_LEVEL\_BARS">

</element>

<element name="NOT\_PROVIDED">

</element>

</enum>

<enum name="PrimaryAudioSource">

<description>Reflects the current primary audio source (if selected).</description>

<element name="NO\_SOURCE\_SELECTED">

</element>

<element name="USB">

</element>

<element name="USB2">

</element>

<element name="BLUETOOTH\_STEREO\_BTST">

</element>

<element name="LINE\_IN">

</element>

<element name="IPOD">

</element>

<element name="MOBILE\_APP">

</element>

</enum>

<enum name="CompassDirection">

<description>The list of potential compass directions</description>

<element name="NORTH">

</element>

<element name="NORTHWEST">

</element>

<element name="WEST">

</element>

<element name="SOUTHWEST">

</element>

<element name="SOUTH">

</element>

<element name="SOUTHEAST">

</element>

<element name="EAST">

</element>

<element name="NORTHEAST">

</element>

</enum>

<enum name="Dimension">

<description>The supported dimensions of the GPS</description>

<element name="NO\_FIX" internal\_name="Dimension\_NO\_FIX">

<description>No GPS at all</description>

</element>

<element name="2D" internal\_name="Dimension\_2D">

<description>Longitude and lattitude</description>

</element>

<element name="3D" internal\_name="Dimension\_3D">

<description>Longitude and lattitude and altitude</description>

</element>

</enum>

<enum name="TouchType">

<element name="BEGIN"/>

<element name="MOVE"/>

<element name="END"/>

</enum>

<enum name="WarningLightStatus">

<description>Reflects the status of a cluster instrument warning light.</description>

<element name="OFF" internal\_name="WLS\_OFF">

</element>

<element name="ON" internal\_name="WLS\_ON">

</element>

<element name="FLASH" internal\_name="WLS\_FLASH">

</element>

<element name="NOT\_USED" internal\_name="WLS\_NOT\_USED">

</element>

</enum>

<enum name="VehicleDataResultCode">

<description>Enumeration that describes possible result codes of a vehicle data entry request.</description>

<element name="SUCCESS" internal\_name="VDRC\_SUCCESS"/>

<element name="TRUNCATED\_DATA" internal\_name="VDRC\_TRUNCATED\_DATA"/>

<element name="DISALLOWED" internal\_name="VDRC\_DISALLOWED"/>

<element name="USER\_DISALLOWED" internal\_name="VDRC\_USER\_DISALLOWED"/>

<element name="INVALID\_ID" internal\_name="VDRC\_INVALID\_ID"/>

<element name="VEHICLE\_DATA\_NOT\_AVAILABLE" internal\_name="VDRC\_DATA\_NOT\_AVAILABLE"/>

<element name="DATA\_ALREADY\_SUBSCRIBED" internal\_name="VDRC\_DATA\_ALREADY\_SUBSCRIBED"/>

<element name="DATA\_NOT\_SUBSCRIBED" internal\_name="VDRC\_DATA\_NOT\_SUBSCRIBED"/>

<element name="IGNORED" internal\_name="VDRC\_IGNORED"/>

</enum>

<enum name="VehicleDataType">

<description>Defines the data types that can be published and subscribed to.</description>

<element name="VEHICLEDATA\_GPS">

<description>Notifies GPSData may be subscribed</description>

</element>

<element name="VEHICLEDATA\_SPEED" />

<element name="VEHICLEDATA\_RPM" />

<element name="VEHICLEDATA\_FUELLEVEL" />

<element name="VEHICLEDATA\_FUELLEVEL\_STATE" />

<element name="VEHICLEDATA\_FUELCONSUMPTION" />

<element name="VEHICLEDATA\_EXTERNTEMP" />

<element name="VEHICLEDATA\_VIN" />

<element name="VEHICLEDATA\_PRNDL" />

<element name="VEHICLEDATA\_TIREPRESSURE" />

<element name="VEHICLEDATA\_ODOMETER" />

<element name="VEHICLEDATA\_BELTSTATUS" />

<element name="VEHICLEDATA\_BODYINFO" />

<element name="VEHICLEDATA\_DEVICESTATUS" />

<element name="VEHICLEDATA\_ECALLINFO" />

<element name="VEHICLEDATA\_AIRBAGSTATUS" />

<element name="VEHICLEDATA\_EMERGENCYEVENT" />

<element name="VEHICLEDATA\_CLUSTERMODESTATUS" />

<element name="VEHICLEDATA\_MYKEY" />

<element name="VEHICLEDATA\_BRAKING" />

<element name="VEHICLEDATA\_WIPERSTATUS" />

<element name="VEHICLEDATA\_HEADLAMPSTATUS" />

<element name="VEHICLEDATA\_BATTVOLTAGE" />

<element name="VEHICLEDATA\_ENGINETORQUE" />

<element name="VEHICLEDATA\_ACCPEDAL" />

<element name="VEHICLEDATA\_STEERINGWHEEL" />

</enum>

<enum name="WiperStatus">

<description>Reflects the status of the wipers.</description>

<element name="OFF">

<description> The wipers are off.</description>

</element>

<element name="AUTO\_OFF">

<description> The wipers are automatically off </description>

</element>

<element name="OFF\_MOVING">

<description> Means that though set to off, somehow the wipers have been engaged (physically moved enough to engage a wiping motion).</description>

</element>

<element name="MAN\_INT\_OFF">

<description> The wipers are manually off after having been working.</description>

</element>

<element name="MAN\_INT\_ON">

<description> The wipers are manually on.</description>

</element>

<element name="MAN\_LOW">

<description> The wipers are manually set to low speed.</description>

</element>

<element name="MAN\_HIGH">

<description> The wipers are manually set to high speed.</description>

</element>

<element name="MAN\_FLICK">

<description> The wipers are manually set for doing a flick.</description>

</element>

<element name="WASH">

<description> The wipers are set to use the water from vehicle washer bottle for cleaning the windscreen.</description>

</element>

<element name="AUTO\_LOW">

<description> The wipers are automatically set to low speed.</description>

</element>

<element name="AUTO\_HIGH">

<description> The wipers are automatically set to high speed.</description>

</element>

<element name="COURTESYWIPE">

<description> This is for when a user has just initiated a WASH and several seconds later a secondary wipe is automatically initiated to clear remaining fluid.</description>

</element>

<element name="AUTO\_ADJUST">

<description> This is set as the user moves between possible automatic wiper speeds.</description>

</element>

<element name="STALLED">

<description> The wiper is stalled to its place. </description>

</element>

<element name="NO\_DATA\_EXISTS">

<description> The sensor / module cannot provide any information for wiper.</description>

</element>

</enum>

<enum name="SamplingRate">

<description>Describes different sampling options for PerformAudioPassThru.</description>

<element name="8KHZ" internal\_name="RATE\_8KHZ"/>

<element name="16KHZ" internal\_name="RATE\_16KHZ"/>

<element name="22KHZ" internal\_name="RATE\_22KHZ"/>

<element name="44KHZ" internal\_name="RATE\_44KHZ"/>

</enum>

<enum name="BitsPerSample">

<description>Describes different quality options for PerformAudioPassThru.</description>

<element name="8\_BIT" internal\_name="RATE\_8\_BIT"/>

<element name="16\_BIT" internal\_name="RATE\_16\_BIT"/>

</enum>

<enum name="AudioType">

<description>Describes different audio type options for PerformAudioPassThru.</description>

<element name="PCM" />

</enum>

<enum name="KeyboardLayout">

<description>Enumeration listing possible keyboard layouts.</description>

<element name="QWERTY" />

<element name="QWERTZ" />

<element name="AZERTY" />

</enum>

<enum name="KeyboardEvent">

<description>Enumeration listing possible keyboard events.</description>

<element name="KEYPRESS" />

<element name="ENTRY\_SUBMITTED" />

<element name="ENTRY\_CANCELLED" />

<element name="ENTRY\_ABORTED" />

</enum>

<enum name="KeypressMode">

<description>Enumeration listing possible keyboard events.</description>

<element name="SINGLE\_KEYPRESS">

<description>Each keypress is individually sent as the user presses the keyboard keys.</description>

</element>

<element name="QUEUE\_KEYPRESSES">

<description>The keypresses are queued and a string is eventually sent once the user chooses to submit their entry.</description>

</element>

<element name="RESEND\_CURRENT\_ENTRY">

<description>The keypresses are queue and a string is sent each time the user presses a keyboard key; the string contains the entire current entry.</description>

</element>

</enum>

<enum name="AmbientLightStatus">

<description>Reflects the status of the ambient light sensor.</description>

<element name="NIGHT" />

<element name="TWILIGHT\_1" />

<element name="TWILIGHT\_2" />

<element name="TWILIGHT\_3" />

<element name="TWILIGHT\_4" />

<element name="DAY" />

<element name="UNKNOWN" />

<element name="INVALID" />

</enum>

<enum name="FileType">

<description>Enumeration listing possible file types.</description>

<element name="GRAPHIC\_BMP" />

<element name="GRAPHIC\_JPEG" />

<element name="GRAPHIC\_PNG" />

<element name="AUDIO\_WAVE" />

<element name="AUDIO\_MP3" />

<element name="AUDIO\_AAC" />

<element name="BINARY" />

<element name="JSON" />

</enum>

<enum name="RequestType">

<description>Enumeration listing possible asynchronous requests.</description>

<element name="HTTP" />

<element name="FILE\_RESUME" />

<element name="AUTH\_REQUEST" />

<element name="AUTH\_CHALLENGE" />

<element name="AUTH\_ACK" />

</enum>

<enum name="ECallConfirmationStatus">

<description>Reflects the status of the eCall Notification.</description>

<element name="NORMAL" internal\_name="ECCS\_NORMAL">

</element>

<element name="CALL\_IN\_PROGRESS" internal\_name="ECCS\_CALL\_IN\_PROGRESS">

</element>

<element name="CALL\_CANCELLED" internal\_name="ECCS\_CALL\_CANCELLED">

</element>

<element name="CALL\_COMPLETED">

</element>

<element name="CALL\_UNSUCCESSFUL" internal\_name="ECCS\_CALL\_UNSUCCESSFUL">

</element>

<element name="ECALL\_CONFIGURED\_OFF" internal\_name="ECCS\_ECALL\_CONFIGURED\_OFF">

</element>

<element name="CALL\_COMPLETE\_DTMF\_TIMEOUT" internal\_name="ECCS\_CALL\_COMPLETE\_DTMF\_TIMEOUT">

</element>

</enum>

<enum name="VehicleDataNotificationStatus">

<description>Reflects the status of a vehicle data notification.</description>

<element name="NOT\_SUPPORTED" internal\_name="VDNS\_NOT\_SUPPORTED">

</element>

<element name="NORMAL" internal\_name="VDNS\_NORMAL">

</element>

<element name="ACTIVE" internal\_name="VDNS\_ACTIVE">

</element>

<element name="NOT\_USED" internal\_name="VDNS\_NOT\_USED">

</element>

</enum>

<enum name="EmergencyEventType">

<description>Reflects the emergency event status of the vehicle.</description>

<element name="NO\_EVENT" internal\_name="EET\_NO\_EVENT">

</element>

<element name="FRONTAL" internal\_name="EET\_FRONTAL">

</element>

<element name="SIDE" internal\_name="EET\_SIDE">

</element>

<element name="REAR" internal\_name="EET\_REAR">

</element>

<element name="ROLLOVER" internal\_name="EET\_ROLLOVER">

</element>

<element name="NOT\_SUPPORTED" internal\_name="EET\_NOT\_SUPPORTED">

</element>

<element name="FAULT" internal\_name="EET\_FAULT">

</element>

</enum>

<enum name="FuelCutoffStatus">

<description>Reflects the status of the RCM fuel cutoff.</description>

<element name="TERMINATE\_FUEL" internal\_name="FCS\_TERMINATE\_FUEL">

</element>

<element name="NORMAL\_OPERATION" internal\_name="FCS\_NORMAL\_OPERATION">

</element>

<element name="FAULT" internal\_name="FCS\_FAULT">

</element>

</enum>

<enum name="PowerModeQualificationStatus">

<description>Reflects the status of the current power mode qualification.</description>

<element name="POWER\_MODE\_UNDEFINED">

</element>

<element name="POWER\_MODE\_EVALUATION\_IN\_PROGRESS">

</element>

<element name="NOT\_DEFINED">

</element>

<element name="POWER\_MODE\_OK">

</element>

</enum>

<enum name="CarModeStatus">

<description>Reflects the status of the current car mode.</description>

<element name="NORMAL" internal\_name="CMS\_NORMAL">

</element>

<element name="FACTORY" internal\_name="CMS\_FACTORY">

</element>

<element name="TRANSPORT" internal\_name="CMS\_TRANSPORT">

</element>

<element name="CRASH" internal\_name="CMS\_CRASH">

</element>

</enum>

<enum name="PowerModeStatus">

<description>Reflects the status of the current power mode.</description>

<element name="KEY\_OUT">

</element>

<element name="KEY\_RECENTLY\_OUT">

</element>

<element name="KEY\_APPROVED\_0">

</element>

<element name="POST\_ACCESORY\_0">

</element>

<element name="ACCESORY\_1">

</element>

<element name="POST\_IGNITION\_1">

</element>

<element name="IGNITION\_ON\_2">

</element>

<element name="RUNNING\_2">

</element>

<element name="CRANK\_3">

</element>

</enum>

<enum name="VRStatus">

<description>VR status</description>

<element name="SUCCESS">

</element>

<element name="FAIL">

</element>

<element name="TIME\_OVER">

</element>

</enum>

## 结构

<struct name="ButtonCapabilities">

<description>Contains the information about capabilities of a button.</description>

<param name="name" type="Common.ButtonName" mandatory="true">

<description>The name of the Button from the ButtonName enum</description>

</param>

<param name="shortPressAvailable" type="Boolean" mandatory="true">

<description>The button supports a short press. Whenever the button is pressed short, onButtonPressed(SHORT) should be invoked.</description>

</param>

<param name="longPressAvailable" type="Boolean" mandatory="true">

<description>The button supports a LONG press. Whenever the button is pressed long, onButtonPressed(LONG) should be invoked.</description>

</param>

<param name="upDownAvailable" type="Boolean" mandatory="true">

<description>The button supports "button down" and "button up". Whenever the button is pressed, onButtonEvent(DOWN) should be invoked. Whenever the button is released, onButtonEvent(UP) should be invoked.</description>

</param>

</struct>

<struct name="VehicleDataResult">

<description>Individual published data request result</description>

<param name="dataType" type="Common.VehicleDataType">

<description>Defined published data element type.</description>

</param>

<param name="resultCode" type="Common.VehicleDataResultCode">

<description>Published data result code.</description>

</param>

</struct>

<struct name="TouchCoord">

<param name="x" type="Integer" mandatory="true" minvalue="0" maxvalue="10000">

<description>The x coordinate of the touch.</description>

</param>

<param name="y" type="Integer" mandatory="true" minvalue="0" maxvalue="10000">

<description>The y coordinate of the touch.</description>

</param>

</struct>

<struct name="TouchEvent">

<param name="id" type="Integer" mandatory="true" minvalue="0" maxvalue="9">

<description>

A touch's unique identifier. The application can track the current touch events by id.

If a touch event has type begin, the id should be added to the set of touches.

If a touch event has type end, the id should be removed from the set of touches.

</description>

</param>

<param name="ts" type="Integer" mandatory="true" array="true" minvalue="0" maxvalue="2147483647" minsize="1" maxsize="1000">

<description>

The time that the touch was recorded. This number can the time since the beginning of the session or something else as long as the units are in milliseconds.

The timestamp is used to determined the rate of change of position of a touch.

The application also uses the time to verify whether two touches, with different ids, are part of a single action by the user.

If there is only a single timestamp in this array, it is the same for every coordinate in the coordinates array.

</description>

</param>

<param name="c" type="Common.TouchCoord" mandatory="true" array="true" minsize="1" maxsize="1000">

</param>

</struct>

<struct name="PresetBankCapabilities">

<description>Contains information about on-screen preset capabilities (whether the HW preset buttons could be duplicated with onscreen buttons).</description>

<param name="onScreenPresetsAvailable" type="Boolean" mandatory="true">

<description>Must be true if onscreen custom presets are available.</description>

</param>

</struct>

<struct name="Image">

<param name="value" maxlength="65535" type="String" mandatory="true">

<description>The path to the dynamic image stored on HU or the static binary image itself.</description>

</param>

<param name="imageType" type="Common.ImageType" mandatory="true">

<description>Describes, whether it is a static or dynamic image.</description>

</param>

</struct>

<struct name="SoftButton">

<param name="type" type="Common.SoftButtonType" mandatory="true">

<description>Describes, whether text, icon or both text and image should be displayed on the soft button. See softButtonType</description>

</param>

<param name="text" maxlength="500" type="String" mandatory="false">

<description>Optional text to be displayed (if defined as TEXT or BOTH)</description>

</param>

<param name="image" type="Common.Image" mandatory="false">

<description>Optional image struct for SoftButton (if defined as IMAGE or BOTH)</description>

</param>

<param name="isHighlighted" type="Boolean" mandatory="false">

<description>If true, must be highlighted</description>

<description>If false, must be not</description>

</param>

<param name="softButtonID" type="Integer" minvalue="0" maxvalue="65535" mandatory="true">

<description>Value which must be returned via OnButtonPress / OnButtonEvent</description>

</param>

<param name="systemAction" type="Common.SystemAction" mandatory="false">

<description>Parameter indicates whether clicking a SoftButton must call a specific system action. See SystemAction</description>

</param>

</struct>

<struct name="TTSChunk">

<description>A TTS chunk, that consists of the text/phonemes to be spoken</description>

<param name="text" type="String" mandatory="true" maxlength="500">

<description>The text or phonemes to be spoken.</description>

</param>

<param name="type" type="Common.SpeechCapabilities" mandatory="true">

<description>Describes, whether it is text or a specific phoneme set. See SpeechCapabilities.</description>

</param>

</struct>

<struct name="HMIApplication">

<description>Data type containing information about application needed by HMI.</description>

<param name="appName" type="String" maxlength="100" mandatory="true">

<description>The mobile application name, e.g. "Ford Drive Green".</description>

</param>

<param name="ngnMediaScreenAppName" type="String" maxlength="100" mandatory="false">

<description>Provides an abbreviated version of the app name (if needed), that may be displayed on the NGN media screen.</description>

<description>If not provided, the appName should be used instead (and may be truncated if too long)</description>

</param>

<param name="icon" type="String" mandatory="false">

<description>Path to application icon stored on HU.</description>

</param>

<param name="deviceName" type="String" mandatory="true">

<description>The name of device which the provided application is running on.</description>

</param>

<param name="appID" type="Integer" mandatory="true">

<description>Unique (during ignition cycle) id of the application. To be used in all RPCs sent by both HU system and SDL</description>

</param>

<param name="hmiDisplayLanguageDesired" type="Common.Language" mandatory="true">

<description>The language the application intends to use on HU </description>

</param>

<param name="isMediaApplication" type="Boolean" mandatory="true">

<description>Indicates whether it is a media or a non-media application.</description>

</param>

<param name="appType" type="Common.AppHMIType" minsize="1" maxsize="100" array="true" mandatory="false">

<description>List of all applicable app types stating which classifications to be given to the app.</description>

<description>e.g. for platforms like GEN2, this determines which "corner(s)" the app can populate.</description>

</param>

</struct>

<struct name="MenuParams">

<param name="parentID" type="Integer" minvalue="0" maxvalue="2000000000" mandatory="false">

<description>unique ID of the sub menu, the command must be added to.</description>

<description>If not provided, the command must be added to the top level of the in application menu.</description>

</param>

<param name="position" type="Integer" minvalue="0" maxvalue="1000" mandatory="false">

<description>Position within the items that are at the top level of the in application menu.</description>

<description>0 should insert at the front.</description>

<description>1 should insert at the second position.</description>

<description>if position is greater than or equal to the number of items on the top level, the the sub menu/command should be appended to the end.</description>

<description>If this param is omitted the entry should be added at the end.</description>

</param>

<param name="menuName" type="String" maxlength="500" mandatory="true">

<description>The name of the sub menu/command.</description>

</param>

</struct>

<struct name="Choice">

<description>A choice is an option given to the user which can be selected either by menu, or through voice recognition system.</description>

<param name="choiceID" type="Integer" minvalue="0" maxvalue="65535" mandatory="true">

<description> The unique within the concerned application identifier for this choice </description>

</param>

<param name="menuName" type="String" maxlength="500" mandatory="false">

<description> The name of the choice </description>

</param>

<param name="image" type="Common.Image" mandatory="false">

<description> The image for representing the choice </description>

</param>

<param name="secondaryText" maxlength="500" type="String" mandatory="false">

<description>Optional secondary text to display; e.g. address of POI in a search result entry</description>

</param>

<param name="tertiaryText" maxlength="500" type="String" mandatory="false">

<description>Optional tertiary text to display; e.g. distance to POI for a search result entry</description>

</param>

<param name="secondaryImage" type="Image" mandatory="false">

<description>Optional secondary image struct for choice</description>

</param>

</struct>

<struct name="VrHelpItem">

<param name="text" maxlength="500" type="String" mandatory="true">

<description>Text to display for VR Help item</description>

</param>

<param name="image" type="Common.Image" mandatory="false">

<description>Image struct for VR Help item</description>

</param>

<param name="position" type="Integer" minvalue="1" maxvalue="100" mandatory="true">

<description>Position to display item in VR Help list</description>

</param>

</struct>

<struct name="TimeFormat">

<param name="hours" type="Integer" minvalue="0" maxvalue="59" mandatory="true">

<description>The hour of the media clock.</description>

<description>Some units only support a max of 19 hours. If out of range, it should be rejected.</description>

</param>

<param name="minutes" type="Integer" minvalue="0" maxvalue="59" mandatory="true"/>

<param name="seconds" type="Integer" minvalue="0" maxvalue="59" mandatory="true"/>

</struct>

<struct name="TouchEventCapabilities">

<param name="pressAvailable" type="Boolean" mandatory="true">

</param>

<param name="multiTouchAvailable" type="Boolean" mandatory="true">

</param>

<param name="doublePressAvailable" type="Boolean" mandatory="true">

</param>

</struct>

<struct name="ImageResolution">

<param name="resolutionWidth" type="Integer" minvalue="1" maxvalue="10000" mandatory="true">

<description>The image resolution width.</description>

</param>

<param name="resolutionHeight" type="Integer" minvalue="1" maxvalue="10000" mandatory="true">

<description>The image resolution height.</description>

</param>

</struct>

<struct name="ScreenParams">

<param name="resolution" type="Common.ImageResolution" mandatory="true">

<description>The resolution of the prescribed screen area.</description>

</param>

<param name="touchEventAvailable" type="Common.TouchEventCapabilities" mandatory="false">

<description>Types of screen touch events available in screen area.</description>

</param>

</struct>

<struct name="ImageField">

<param name="name" type="Common.ImageFieldName" mandatory="true">

<description>The name that identifies the field. See ImageFieldName.</description>

</param>

<param name="imageTypeSupported" type="Common.FileType" maxsize="100" array="true" mandatory="false" minsize="1">

<description>The image types that are supported in this field. See FileType.</description>

</param>

<param name="imageResolution" type="Common.ImageResolution" mandatory="false">

<description>The image resolution of this field.</description>

</param>

</struct>

<struct name="DisplayCapabilities">

<description>Contains information about the display capabilities.</description>

<param name="displayType" type="Common.DisplayType" mandatory="true">

<description>The type of the display. See DisplayType</description>

</param>

<param name="textFields" type="Common.TextFieldName" minsize="0" maxsize="100" array="true" mandatory="true">

<description>A set of all fields for text displaying supported by HU. See TextFieldName.</description>

<description>If there are no textfields supported, the empty array must be returned</description>

</param>

<param name="imageFields" type="Common.ImageField" minsize="1" maxsize="100" array="true" mandatory="false">

<description>A set of all fields that support images. See ImageField</description>

</param>

<param name="mediaClockFormats" type="Common.MediaClockFormat" minsize="1" maxsize="100" array="true" mandatory="true">

<description>A set of all supported formats of the media clock. See MediaClockFormat</description>

</param>

<param name="imageCapabilities" type="Common.ImageType" array="true" minsize="0" maxsize="2" mandatory="false">

</param>

<param name="graphicSupported" type="Boolean" mandatory="true">

<description>The display's persistent screen supports referencing a static or dynamic image.</description>

</param>

<param name="templatesAvailable" type="String" minsize="0" maxsize="100" maxlength="100" array="true" mandatory="true">

<description>A set of all predefined persistent display templates available on headunit. To be referenced in SetDisplayLayout.</description>

</param>

<param name="screenParams" type="Common.ScreenParams" mandatory="false">

<description>A set of all parameters related to a prescribed screen area (e.g. for video / touch input).</description>

</param>

<param name="numCustomPresetsAvailable" type="Integer" minvalue="1" maxvalue="100" mandatory="false">

<description>The number of on-screen custom presets available (if any); otherwise omitted.</description>

</param>

</struct>

<struct name="SoftButtonCapabilities">

<description>Contains information about a SoftButton's capabilities.</description>

<param name="shortPressAvailable" type="Boolean" mandatory="true">

<description>The button supports a short press.</description>

<description>Whenever the button is pressed short, onButtonPressed( SHORT) must be invoked.</description>

</param>

<param name="longPressAvailable" type="Boolean" mandatory="true">

<description>The button supports a LONG press.</description>

<description>Whenever the button is pressed long, onButtonPressed( LONG) must be invoked.</description>

</param>

<param name="upDownAvailable" type="Boolean" mandatory="true">

<description>The button supports "button down" and "button up".</description>

<description>Whenever the button is pressed, onButtonEvent( DOWN) must be invoked.</description>

<description>Whenever the button is released, onButtonEvent( UP) must be invoked.</description>

</param>

<param name="imageSupported" type="Boolean" mandatory="true">

<description>Must be true if the button supports referencing a static or dynamic image.</description>

</param>

</struct>

<struct name="AudioPassThruCapabilities">

<description>

Describes different audio type configurations for PerformAudioPassThru.

e.g. 8kHz,8-bit,PCM

</description>

<param name="samplingRate" type="Common.SamplingRate" mandatory="true"/>

<param name="bitsPerSample" type="Common.BitsPerSample" mandatory="true"/>

<param name="audioType" type="Common.AudioType" mandatory="true"/>

</struct>

<struct name="Coordinate">

<param name="xCoord" type="Integer" mandatory="true">

</param>

<param name="yCoord" type="Integer" mandatory="true">

</param>

</struct>

<struct name="TextFieldStruct">

<param name="fieldName" type="Common.TextFieldName" mandatory="true">

<description>The name of the field for displaying the text.</description>

</param>

<param name="fieldText" type="String" maxlength="500" mandatory="true">

<description>The text itself.</description>

</param>

</struct>

<struct name="KeyboardProperties">

<description>Configuration of on-screen keyboard (if available).</description>

<param name="language" type="Common.Language" mandatory="false">

<description>The keyboard language.</description>

</param>

<param name="keyboardLayout" type="Common.KeyboardLayout" mandatory="false" >

<description>Desired keyboard layout.</description>

</param>

<param name="keypressMode" type="Common.KeypressMode" mandatory="false" >

<description>

Desired keypress mode.

If omitted, this value will be set to RESEND\_CURRENT\_ENTRY.

</description>

</param>

<param name="limitedCharacterList" type="String" maxlength="1" minsize="1" maxsize="100" array="true" mandatory="false">

<description>Array of keyboard characters to enable.</description>

<description>All omitted characters will be greyed out (disabled) on the keyboard.</description>

<description>If omitted, the entire keyboard will be enabled.</description>

</param>

<param name="autoCompleteText" type="String" maxlength="1000" mandatory="false">

<description>Allows an app to prepopulate the text field with a suggested or completed entry as the user types</description>

</param>

</struct>

<struct name="Turn">

<param name="navigationText" type="Common.TextFieldStruct" mandatory="false">

<description>Uses navigationText from TextFieldStruct.</description>

</param>

<param name="turnIcon" type="Common.Image" mandatory="false">

</param>

</struct>

<struct name="VehicleType">

<param name="make" type="String" maxlength="500" mandatory="false">

<description>Make of the vehicle</description>

<description>e.g. Ford</description>

</param>

<param name="model" type="String" maxlength="500" mandatory="false">

<description>Model of the vehicle</description>

<description>e.g. Fiesta</description>

</param>

<param name="modelYear" type="String" maxlength="500" mandatory="false">

<description>Model Year of the vehicle</description>

<description>e.g. 2013</description>

</param>

<param name="trim" type="String" maxlength="500" mandatory="false">

<description>Trim of the vehicle</description>

<description>e.g. SE</description>

</param>

</struct>

<struct name="DeviceInfo">

<param name="name" type="String" mandatory="true">

<description>The name of the device connected.</description>

</param>

<param name="id" type="Integer" mandatory="true">

<description>The ID of the device connected</description>

</param>

</struct>

<!--IVI part-->

<struct name="GPSData">

<description>Struct with the GPS data.</description>

<param name="longitudeDegrees" type="Float" minvalue="-180" maxvalue="180" mandatory="false">

</param>

<param name="latitudeDegrees" type="Float" minvalue="-90" maxvalue="90" mandatory="false">

</param>

<param name="utcYear" type="Integer" minvalue="2010" maxvalue="2100" mandatory="false">

<description>The current UTC year.</description>

</param>

<param name="utcMonth" type="Integer" minvalue="1" maxvalue="12" mandatory="false">

<description>The current UTC month.</description>

</param>

<param name="utcDay" type="Integer" minvalue="1" maxvalue="31" mandatory="false">

<description>The current UTC day.</description>

</param>

<param name="utcHours" type="Integer" minvalue="0" maxvalue="23" mandatory="false">

<description>The current UTC hour.</description>

</param>

<param name="utcMinutes" type="Integer" minvalue="0" maxvalue="59" mandatory="false">

<description>The current UTC minute.</description>

</param>

<param name="utcSeconds" type="Integer" minvalue="0" maxvalue="59" mandatory="false">

<description>The current UTC second.</description>

</param>

<param name="compassDirection" type="Common.CompassDirection" mandatory="false">

<description>See CompassDirection.</description>

</param>

<param name="pdop" type="Float" minvalue="0" maxvalue="10" mandatory="false">

<description>PDOP.</description>

</param>

<param name="hdop" type="Float" minvalue="0" maxvalue="10" mandatory="false">

<description>HDOP.</description>

</param>

<param name="vdop" type="Float" minvalue="0" maxvalue="10" mandatory="false">

<description>VDOP.</description>

</param>

<param name="actual" type="Boolean" mandatory="false">

<description>

True, if actual.

False, if infered.

</description>

</param>

<param name="satellites" type="Integer" minvalue="0" maxvalue="31" mandatory="false">

<description>Number of satellites in view</description>

</param>

<param name="dimension" type="Common.Dimension" mandatory="false">

<description>See Dimension</description>

</param>

<param name="altitude" type="Float" minvalue="-10000" maxvalue="10000" mandatory="false">

<description>Altitude in meters</description>

</param>

<param name="heading" type="Float" minvalue="0" maxvalue="359.99" mandatory="false">

<description>The heading. North is 0. Resolution is 0.01</description>

</param>

<param name="speed" type="Float" minvalue="0" maxvalue="500" mandatory="false">

<description>The speed in KPH</description>

</param>

</struct>

<struct name="SingleTireStatus">

<param name="status" type="Common.ComponentVolumeStatus" mandatory="true">

<description>The status of component volume. See ComponentVolumeStatus.</description>

</param>

</struct>

<struct name="DIDResult">

<description>Individual requested DID result and data</description>

<param name="resultCode" type="Common.VehicleDataResultCode" mandatory="true">

<description>Individual DID result code.</description>

</param>

<param name="didLocation" type="Integer" minvalue="0" maxvalue="65535" mandatory="true">

<description>Location of raw data (the address from ReadDID request)</description>

</param>

<param name="data" type="String" maxlength="5000" mandatory="false">

<description>Raw DID-based data returned for requested element.</description>

</param>

</struct>

<struct name="HeadLampStatus">

<param name="lowBeamsOn" type="Boolean" mandatory="true">

<description>Status of the low beam lamps.</description>

</param>

<param name="highBeamsOn" type="Boolean" mandatory="true">

<description>Status of the high beam lamps.</description>

</param>

<param name="ambientLightSensorStatus" type="Common.AmbientLightStatus" mandatory="true">

<description>Status of the ambient light sensor.</description>

</param>

</struct>

<struct name="TireStatus">

<description>The status and pressure of the tires.</description>

<param name="pressureTelltale" type="Common.WarningLightStatus" mandatory="false">

<description>Status of the Tire Pressure Telltale. See WarningLightStatus.</description>

</param>

<param name="leftFront" type="Common.SingleTireStatus" mandatory="false">

<description>The status of the left front tire.</description>

</param>

<param name="rightFront" type="Common.SingleTireStatus" mandatory="false">

<description>The status of the right front tire.</description>

</param>

<param name="leftRear" type="Common.SingleTireStatus" mandatory="false">

<description>The status of the left rear tire.</description>

</param>

<param name="rightRear" type="Common.SingleTireStatus" mandatory="false">

<description>The status of the right rear tire.</description>

</param>

<param name="innerLeftRear" type="Common.SingleTireStatus" mandatory="false">

<description>The status of the inner left rear.</description>

</param>

<param name="innerRightRear" type="Common.SingleTireStatus" mandatory="false">

<description>The status of the inner right rear.</description>

</param>

</struct>

<struct name="BeltStatus">

<param name="driverBeltDeployed" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="passengerBeltDeployed" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="passengerBuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="driverBuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="leftRow2BuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="passengerChildDetected" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="rightRow2BuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="middleRow2BuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="middleRow3BuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="leftRow3BuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="rightRow3BuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="leftRearInflatableBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="rightRearInflatableBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="middleRow1BeltDeployed" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

<param name="middleRow1BuckleBelted" type="Common.VehicleDataEventStatus" mandatory="false">

<description>See VehicleDataEventStatus.</description>

</param>

</struct>

<struct name="BodyInformation">

<param name="parkBrakeActive" type="Boolean" mandatory="true">

<description>Must be true if the park brake is active</description>

</param>

<param name="ignitionStableStatus" type="Common.IgnitionStableStatus" mandatory="true">

<description>Information about the ignition switch. See IgnitionStableStatus.</description>

</param>

<param name="ignitionStatus" type="Common.IgnitionStatus" mandatory="true">

<description>The status of the ignition. See IgnitionStatus.</description>

</param>

<param name="driverDoorAjar" type="Boolean" mandatory="false">

<description>References signal "DrStatDrv\_B\_Actl".</description>

</param>

<param name="passengerDoorAjar" type="Boolean" mandatory="false">

<description>References signal "DrStatPsngr\_B\_Actl".</description>

</param>

<param name="rearLeftDoorAjar" type="Boolean" mandatory="false">

<description>References signal "DrStatRl\_B\_Actl".</description>

</param>

<param name="rearRightDoorAjar" type="Boolean" mandatory="false">

<description>References signal "DrStatRr\_B\_Actl".</description>

</param>

</struct>

<struct name="DeviceStatus">

<param name="voiceRecOn" type="Boolean" mandatory="false">

<description>Must be true if the voice recording is on.</description>

</param>

<param name="btIconOn" type="Boolean" mandatory="false">

<description>Must be true if Bluetooth icon is displayed.</description>

</param>

<param name="callActive" type="Boolean" mandatory="false">

<description>Must be true if there is an active call..</description>

</param>

<param name="phoneRoaming" type="Boolean" mandatory="false">

<description>Must be true if ther is a phone roaming.</description>

</param>

<param name="textMsgAvailable" type="Boolean" mandatory="false">

<description>Must be true if the text message is available.</description>

</param>

<param name="battLevelStatus" type="Common.DeviceLevelStatus" mandatory="false">

<description>Device battery level status. See DeviceLevelStatus.</description>

</param>

<param name="stereoAudioOutputMuted" type="Boolean" mandatory="false">

<description>Must be true if stereo audio output is muted.</description>

</param>

<param name="monoAudioOutputMuted" type="Boolean" mandatory="false">

<description>Must be true if mono audio output is muted.</description>

</param>

<param name="signalLevelStatus" type="Common.DeviceLevelStatus" mandatory="false">

<description>Device signal level status. See DeviceLevelStatus.</description>

</param>

<param name="primaryAudioSource" type="Common.PrimaryAudioSource" mandatory="false">

<description>See PrimaryAudioSource.</description>

</param>

<param name="eCallEventActive" type="Boolean" mandatory="false">

<description>Must be true if emergency call event is active.</description>

</param>

</struct>

<struct name="ECallInfo">

<param name="eCallNotificationStatus" type="Common.VehicleDataNotificationStatus">

<description>References signal "eCallNotification\_4A". See VehicleDataNotificationStatus.</description>

</param>

<param name="auxECallNotificationStatus" type="Common.VehicleDataNotificationStatus">

<description>References signal "eCallNotification". See VehicleDataNotificationStatus.</description>

</param>

<param name="eCallConfirmationStatus" type="Common.ECallConfirmationStatus">

<description>References signal "eCallConfirmation". See ECallConfirmationStatus.</description>

</param>

</struct>

<struct name="AirbagStatus">

<param name="driverAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsDrvBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="driverSideAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsDrvSideBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="driverCurtainAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsDrvCrtnBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="passengerAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsPasBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="passengerCurtainAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsPasCrtnBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="driverKneeAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsKneeDrvBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="passengerSideAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsPasSideBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="passengerKneeAirbagDeployed" type="Common.VehicleDataEventStatus">

<description>References signal "VedsKneePasBag\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

</struct>

<struct name="EmergencyEvent">

<param name="emergencyEventType" type="Common.EmergencyEventType">

<description>References signal "VedsEvntType\_D\_Ltchd". See EmergencyEventType.</description>

</param>

<param name="fuelCutoffStatus" type="Common.FuelCutoffStatus">

<description>References signal "RCM\_FuelCutoff". See FuelCutoffStatus.</description>

</param>

<param name="rolloverEvent" type="Common.VehicleDataEventStatus">

<description>References signal "VedsEvntRoll\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="maximumChangeVelocity" type="Common.VehicleDataEventStatus">

<description>References signal "VedsMaxDeltaV\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

<param name="multipleEvents" type="Common.VehicleDataEventStatus">

<description>References signal "VedsMultiEvnt\_D\_Ltchd". See VehicleDataEventStatus.</description>

</param>

</struct>

<struct name="ClusterModeStatus">

<param name="powerModeActive" type="Boolean">

<description>References signal "PowerMode\_UB".</description>

</param>

<param name="powerModeQualificationStatus" type="Common.PowerModeQualificationStatus">

<description>References signal "PowerModeQF". See PowerModeQualificationStatus.</description>

</param>

<param name="carModeStatus" type="Common.CarModeStatus">

<description>References signal "CarMode". See CarMode.</description>

</param>

<param name="powerModeStatus" type="Common.PowerModeStatus">

<description>References signal "PowerMode". See PowerMode.</description>

</param>

</struct>

<struct name="MyKey">

<param name="e911Override" type="Common.VehicleDataStatus" mandatory="true">

<description>Indicates whether e911 override is on. See VehicleDataStatus.</description>

</param>

</struct>