

# 基于ODX车辆诊断研究综述

马开献, 邱琪雯

(维克多汽车技术上海有限公司, 上海 200050)

**摘要:** 介绍ODX的7层数据模型, 重点阐述ODX-D与ODX-C两层与诊断数据密切相关的信息, 最后展望ODX协议的发展趋势。

**关键词:** ODX数据模型; ODX应用; 诊断服务

**中图分类号:** U463.6 **文献标志码:** A **文章编号:** 1003-8639 (2017) 11-0028-05

## Vehicle Diagnostic Research Based on ODX

MA Kai-xian, QIU Qi-wen

(Vector Automotive Technology (Shanghai) Co., Ltd., Shanghai 200050, China)

**Abstract:** This paper introduces ODX data model for seven layers, and focuses on ODX-D and ODX-C layers, because they are the most important layers related to diagnostic data. At last, the development trend of ODX protocol is mentioned.

**Key words:** ODX data model; ODX application; diagnostic services

DOI:10.13273/j.cnki.qcdq.2017.11.009

ODX (Open Diagnostic Data Exchange) 是一种开放式的诊断数据格式, 基于XML语言, 用于车辆全生命周期中诊断数据的交互。ODX最初由ASAM (自动化及测量系统标准协会) 提出并形成标准MCD-2D, 在2008年以ODX 2.2.0为基础形成了ISO标准——ISO 22901-1。

ODX提出之前, 在诊断开发的每个阶段, 应用场景和开发工具都是不一样的, 所以每个阶段所使用的数据格式也不一样。例如在研发阶段, 通常采用.doc或.rtf格式来描述诊断需求; 在测试验证阶段, 以业界常用的Vector公司的CANoe.DiVa工具为例, 采用.cdd格式; 同样, 在生产、实现ECU代码及售后阶段, 诊断需求所采用的描述格式也不一样。然而, 诊断开发流程中的各个阶段又相互依存, 密不可分。如果用以描述诊断需求的数据格式不断转换, 数据的一致性将很难保证, 会出现数据遗漏、转换错误等问题, 既增大了需求管理的复杂度, 又使各阶段工作衔接困难, 增加综合成本。如果在整个诊断开发流程中均使用ODX作为诊断需求描述的数据格式, 因为ODX的开源及标准化的属性, 上述问题都得以解决。

## 1 ODX概述

目前最新版本的ODX为2.2.0, 整个ODX数据模型分为ODX-D、ODX-C、ODX-V、ODX-F、ODX-E、

ODX-FD、ODX-M 7层, 如图1所示。

ODX-D部分主要描述了诊断仪与ECU之间的通信过程, 包括诊断服务的请求、响应格式及所用到的参数类型等; ODX-C部分描述了诊断仪与ECU之间的通信参数, 例如网络层定时参数、应用层定时参数、波特率等; ODX-V

部分描述了车辆的信息, 例如OEM信息、车型信息、车辆拓扑等; ODX-F部分主要对上传下载的数据进行描述, 应用于在线刷新程序; ODX-E部分描述了车辆的配置信息, 包括根据特定的车辆环境、地点, 使能/关闭可选功能, 设定特征曲线等, 主要应用于ECU生产、售后阶段; ODX-FD部分描述了面向功能的诊断信息; ODX-M部分描述了多个ECU共同实现的某些诊断功能信息。ODX数据模型是通过UML模型定义的, 图2所示为ODX-D中的诊断服务描述的UML模型。

而在实际应用中, 需要将UML模型映射成相应的XML代码, 图2模型所映射的部分代码如下。

```
<DIAG-COMMS>
```

```
<DIAG-SERVICE ID="ExampleServiceID"
ADDRESSING="FUNCTIONAL-OR-PHYSICAL">
```



图1 ODX数据模型

收稿日期: 2017-02-13

作者简介: 马开献, 硕士, 从事汽车电子工作 10 年。

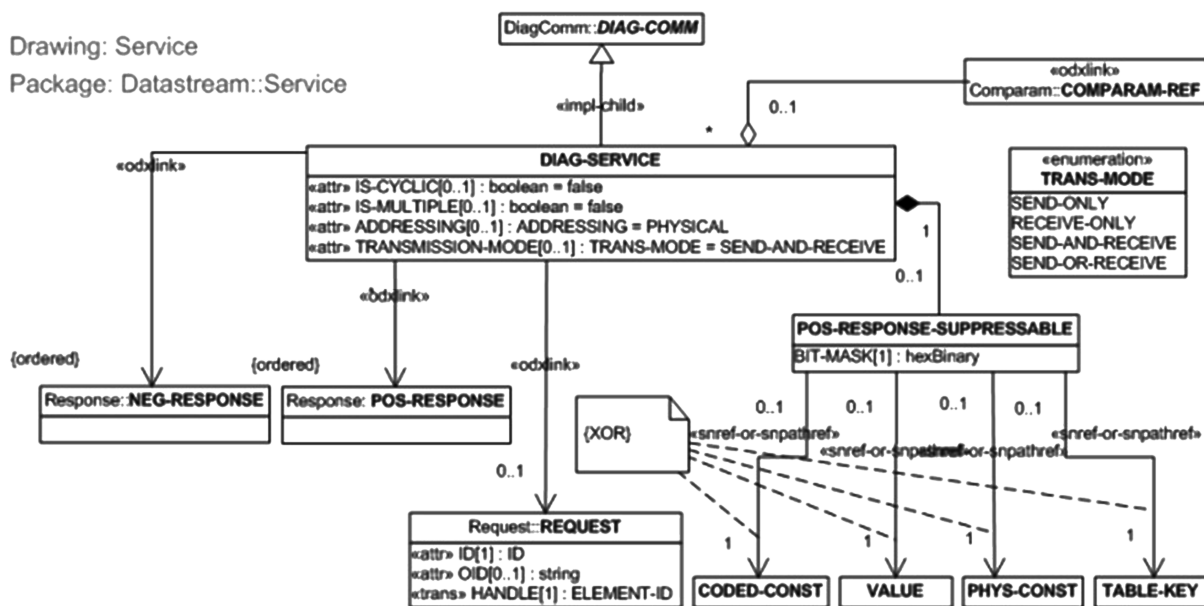


图2 ODX-D层服务描述的UML模型

```

<SHORT-NAME>ExampleService</SHORT-NAME>

```

```

.....

```

```

<REQUEST-REF ID=

```

```

REF="ExampleRequestID"/>

```

```

<POS-RESPONSE-REFS>

```

```

<POS-RESPONSE-REF ID=

```

```

REF="ExampleResponseID"/>

```

```

</POS-RESPONSE-REFS>

```

```

.....

```

```

</DIAG-SERVICE>

```

```

</DIAG-COMMS>

```

```

<REQUESTS>

```

```

<REQUEST ID="ExampleRequestID">

```

```

.....

```

```

<PARAMS>

```

```

<PARAM xsi: type="CODED-CONST">

```

```

.....

```

```

</PARAM>

```

```

.....

```

```

</REQUEST>

```

```

</REQUESTS>

```

ODX-D与ODX-C是ODX数据模型的基础，是诊断数据的描述必不可少的两个部分。对于OEM来讲，ODX-V也是不可或缺的。本文主要介绍ODX-D与ODX-C层的格式及主要内容。

## 2 ODX-D与ODX-C层介绍

### 2.1 ODX-D诊断数据层

ODX数据模型构造的诊断数据分布在5个所谓的诊断层中，这5层分别是协议层（Protocol Layer）、功能组层（Functional-Group Layer）、基础变量层（Base Variant Layer）、ECU变量（ECU Variant Layer）以及ECU共享数据层（ECU-Shared-Data Layer），如图3所示。这5层存在着值继承（Value Inheritance）的关系，如图4所示。

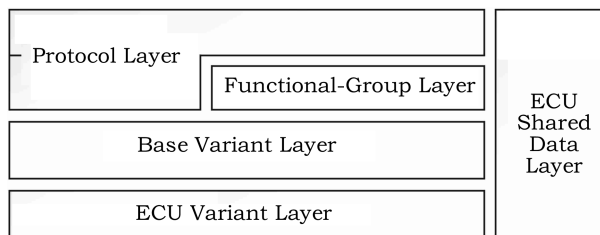


图3 ODX-D诊断数据层结构

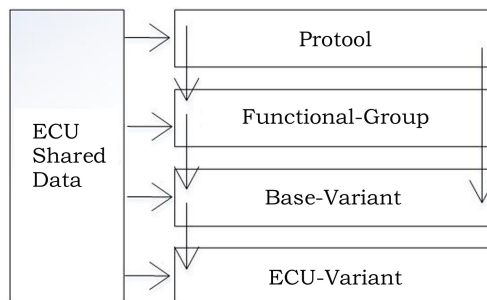


图4 ODX-D诊断数据层值继承图

图4中箭头所指方向,即为值继承方向。ECU变量层继承基础变量层,基础变量继承层功能组层以及协议层,功能组层继承协议层,与此同时,ECU共享数据层被其余4层共同继承。因此,ECU共享数据层的数据是诊断数据层的主要结构。

ECU共享数据层包含了汽车诊断中所需要的协议服务分类,协议服务列表以及诊断通信过程中的测试端向ECU发出的请求信息与ECU回应给测试端的应答信息。

## 2.2 ODX-C通信参数层

ODX-C具有附属结构ODX-CS,两种结构的关系如图5所示。

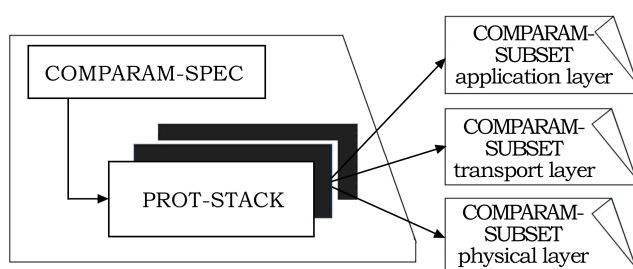


图5 ODX-C与ODX-CS关系图 (简易)

<COMPARAM-SPEC> (ODX-C) 结构中包含有一个或多个<PROT-STACK>, <PROT-STACK>定义了通信参数的设置,并被分成多个<COMPARAM-SUBSET> (ODX-CS) 结构,这些结构分别对应于<PROT-STACK>中所定义的应用层、传输层以及物理层的协议内容。

## 2.3 ODX-D与ODX-C的关联

ODX-D与ODX-C两层,作为ODX数据模型中诊断数据描述必不可少的两个部分,必然是具有一定关联的,图6显示了ODX-D与ODX-C以及ODX-CS之间的关系。

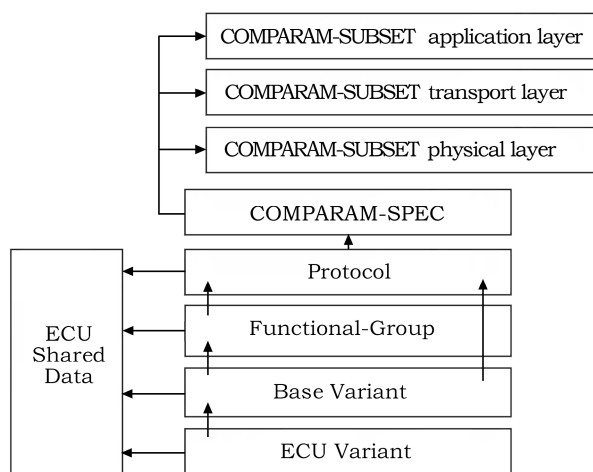


图6 ODX-D与ODX-C关系图 (简易)

ODX-C继承了ODX-CS结构中关于协议内容详细的描述信息,而ODX-D层的Protocol协议层继承了ODX-C的值与信息,所以ODX-D与ODX-C层在汽车诊断中都是必不可少的。

## 3 诊断服务数据寻址方式描述

ODX具有多个数据层,而几乎每层中又都包含有多个层次,而每个数据层以及每一层的内部层次都具有关联性,不同层次都具有类似的寻址方式。ODX数据模型的实际应用是通过XML代码生成的一系列文档进行的,本章节将以ODX-D层的服务描述为例,介绍XML代码是如何实现寻址的。诊断服务描述寻址流程如图7所示。

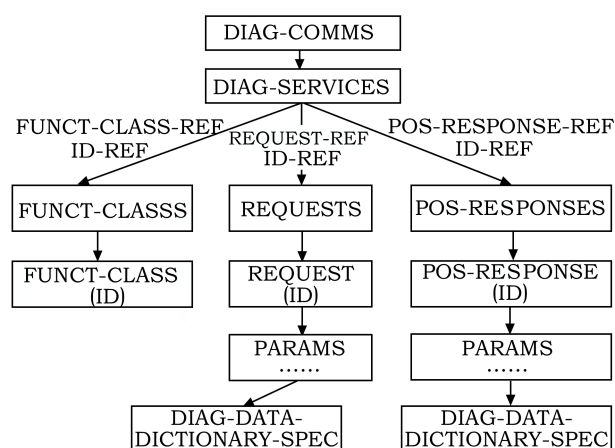


图7 诊断服务描述寻址流程 (简易)

在XML文档中<DIAG-COMMS>、<FUNCT-CLASS>、<REQUESTS>、<POS-RESPONSES>是并列的4个元素标记,即四者同级,不存在从属关系,但是在实际内容上又是相互关联的。例如<FUNCT-CLASS>是用来分类<DIAG-COMM>的,因此两者之间是通过在<FUNCT-CLASS>中查找与<DIAG-COMM>下属的<DIAG-SERVICES>中的<FUNCT-CLASS-REF ID-REF>值相同的<FUNCT-CLASS ID>的值。以此类推,<REQUEST-REF ID-REF>和<POS-RESPONSE-REF ID-REF>就是用来查找对应的<REQUEST ID>和<POS-RESPONSE ID>的。

例如<DIAG-COMMS>结构中存在:

```
<DIAG-SERVICE ID="DLC.UDS_Services.ESD.UDS_Services.DC.Read_Data_By_Identifier" SEMANTIC="DATA-READ">
```

```
<SHORT-NAME>Read_Data_By_Identifier</SHORT-NAME>
```

```

.....
<FUNCT-CLASS-REFS>
<FUNCT-CLASS-REF ID-REF="DLC.UDS_
Services.ESD.UDS_Services.FNC.Data_Transmission_
generic" />
</FUNCT-CLASS-REFS>
<REQUEST-REF ID-REF="DLC.UDS_Services.
ESD.UDS_Services.RQ.Read_Data_By_Identifier_
Request"/>
<POS-RESPONSE-REFS>
<POS-RESPONSE-REF ID-REF="DLC.UDS_
Services.ESD.UDS_Services.PR.Read_Data_By_
Identifier_Response"/>
</POS-RESPONSE-REFS>
</DIAG-SERVICE>

```

1) ReadDataByIdentifier服务的分类ID为: "DLC.UDS\_Services.ESD.UDS\_Services.FNC.Data\_Transmission\_generic", 则查找<FUNCT-CLASS>中的内容, 结果为:

```

<FUNCT-CLASS ID="DLC.UDS_Services.ESD.
UDS_Services.FNC.Data_Transmission_generic">
<SHORT-NAME>Data_Transmission_generic</
SHORT-NAME>
.....
<DESC>
<p>UDS functional unit Data Transmission ( generic
services ) </p>
</DESC>
</FUNCT-CLASS>

```

该语句表示ReadDataByIdentifier服务在ODX文件所存储的该协议中, 被分类为DataTransmission (generic), 即数据转换 (通用)。

2) ReadDataByIdentifier服务的请求ID为: "DLC.UDS\_Services.ESD.UDS\_Services.RQ.Read\_Data\_By\_Identifier\_Request", 则查找<REQUESTS>中的内容, 部分结果为:

```

<REQUEST ID="DLC.UDS_Services.ESD.UDS_
Services.RQ.Read_Data_By_Identifier_Request">
<SHORT-NAME>Read_Data_By_Identifier_
Request</SHORT-NAME>
.....
<PARAMS>
<PARAM SEMANTIC="SERVICE-ID" xsi:
type="CODED-CONST">
<SHORT-NAME>Service_Id</SHORT-NAME>

```

```

.....
<CODED-VALUE>34</CODED-VALUE>
.....
</PARAM>
<PARAM ID="DLC.UDS_Services.ESD.UDS_
Services.RQ.Read_Data_By_Identifier_Request.PA.Data_
Id_1" xsi: type="TABLE-KEY">
<SHORT-NAME>Data_Id</SHORT-NAME>
.....
<TABLE-REF ID-REF="DLC.UDS_Services.ESD.
UDS_Services.TA.DID_Table"/>
</PARAM>
</PARAMS>
</REQUEST>

```

由<CODED-VALUE>34</CODED-VALUE>可知, ReadDataByIdentifier服务请求ID为0x22 (十进制数34的十六进制数), 即为请求信息的第一位。这一位是由协议标准定义的, 不能手动更改, 而请求的后几位是根据应用或者由使用者定义的, 在该实例中是由<TABLE-REF ID-REF="DLC.UDS\_Services.ESD.UDS\_Services.TA.DID\_Table"/>指向的部分定义。而此部分的具体内容是存储在图7所示的<DIAG-DATA-Dictionary-SPEC>结构中的, 该结构是用以解码诊断数据的, 同样是通过查找对应的ID值来寻址, 该部分代码在此不赘述。

3) 从<DIAG-COMMS>的语句中可得, ReadDataByIdentifier服务的肯定应答ID为: "DLC.UDS\_Services.ESD.UDS\_Services.PR.Read\_Data\_By\_Identifier\_Response"则查找<POS-RESPONSES>中的语句, 结果为:

```

<POS-RESPONSE ID="DLC.UDS_Services.ESD.
UDS_Services.PR.Read_Data_By_Identifier_Response">
<SHORT-NAME>Read_Data_By_Identifier_
Response</SHORT-NAME>
.....
<PARAMS>
<PARAM xsi: type="CODED-CONST">
<SHORT-NAME>Service_Id</SHORT-NAME>
.....
<CODED-VALUE>98</CODED-VALUE>
.....
</PARAM>
<PARAM xsi: type="VALUE">
<SHORT-NAME>Data</SHORT-NAME>
.....

```



<DOP-REF ID-REF="ODX-RS.DOP.0to0ByteENDOF  
PDUterminatedIDENTICALBYTEFIELD"

DOCREF="ODX\_RS\_DOP\_LIB\_DLC"  
DOCTYPE="CONTAINER"/>

</PARAM>

</PARAMS>

</POS-RESPONSE>

由此可得, ReadDataByIdentifier服务应答ID为0x62(十进制数98的十六进制数),即为应答信息的第一位。应答信息的后面的位由<DOP-REF ID-REF="ODX-RS.DOP.0to0ByteENDOF PDUterminated IDENTICALBYTEFIELD" DOCREF="ODX\_RS\_DOP\_LIB\_DLC" DOCTYPE="CONTAINER"/>指向的部分定义。

由于ODX是一个用以存储大量数据的数据库,所以每一个ODX项目都会包含多个ODX文件,在寻址过程中,往往也会从一个文件跳转到另一个文件,也是通过查找ID的方式来实现的。例如上文的<DOP-REF ID-REF="ODX-RS.DOP.0to0ByteENDOF PDUterminated IDENTICALBYTEFIELD" DOCREF="ODX\_RS\_DOP\_LIB\_DLC" DOCTYPE="CONTAINER"/>则是跳转到了另一个ODX文件,该文件名为"ODX\_RS\_DOP\_LIB\_DLC",所对应的具体内容ID为"ODX-RS.DOP.0to0ByteENDOF PDUterminated IDENTICALBYTEFIELD"。

ODX文件的寻址方式比较简单,主要就是通过查找相同的ID与存储文件名来实现的。

#### 4 结语

在国外尤其是欧美的OEM中,ODX技术的应用 DOI:10.13273/j.cnki.qcdq.2017.11.029

相对比较成熟,但是随着研究与应用的深入,涌现出不同的需求,这也是标准不断完善和新工具诞生的源动力。例如在2011年,ASAM发布的ODX Authoring Guidelines,为建立ODX文件时的命名(比如文件名称、ODX-Link、DOP)、SDG的描述、服务的描述(SEMANTIC、Service-Id、否定响应码)、DTC的描述、数据交互过程给出了统一的建议及相关示例,这就进一步推进了诊断数据格式ODX在OEM和供应商之间的交互应用。标准化组织、OEM和工具供应商的共同努力将继续推动ODX的应用和发展。

国内OEM的诊断技术和诊断开发流程都在不断积累和完善,ODX能够减少诊断数据以及工具管理的成本,这必将使得ODX在国内汽车诊断行业中被更广泛地应用。国内OEM将逐步统一诊断数据格式,同时和工具供应商一起合作研究开发适用于自己的诊断工具链。

由于ODX是一种开源化的诊断数据格式,OEM及供应商将共同面临其安全保密性的问题,因此,基于这种开源的数据格式,数据的加密、保护必定也会有相应的研究与发展。

#### 参考文献:

- [1] 靳然. ODX应用现状与发展趋势[EB/OL].http://auto.vogel.com.cn/news\_view.html?id=358026.
- [2] ISO22901-1, Road vehicles-Open diagnostic data exchange-Part1: Data model specification [S].

(编辑 心翔)

## eNow展示太阳能驱动的零排放运输制冷单元

eNow公司已经在一辆于城市环境中送货的卡车上展示了一款太阳能零排放商业用途运输制冷单元(TRU),新的TRU品牌“Rayf Refrigeration”在五个月的测试期间实现了98%的N<sub>2</sub>O和97%的PM减排,传统上,TRU由高污染的小型柴油发动机提供动力。

该Rayf Refrigeration系统采用eNow太阳能与Johnson Truck Bodies制冷机组以及Enerson的高效压缩机技术相结合,该单元的冷板和电池最初是从公用电源充电一夜,但在送货路线上,由安装在卡车车顶上的eNow太阳能光伏(PV)面板提供电源。

该1800瓦的eNow太阳能系统提供超过足够的能源通过典型的一天中开启和关闭车门来保持最佳温度,同时,该冷藏卡车在加州夏季的高温中提供新鲜的乳制

品。

eNow团队计算,CO<sub>2</sub>的平均排放量从2525磅/周下降到159磅/周;N<sub>2</sub>O排放量从7162克减少到1克。这是在调整供电电网的电力过夜之后实现的(太阳能的任务为0)。

除了消除有害排放物之外,预计Rayf Refrigeration装置将运营成本降低达90%,该节省是通过消除柴油燃料和维护成本以及由于eNow太阳能的一致电量维护而延长了电池寿命来实现的。

圣华金谷空气污染管制区通过技术进步计划资助了该Rayf Refrigeration计划。

(信息来源:2017.10.12 Green Car Congress)戴朝典 编译