

指导性文件
GUIDANCE NOTES
GD21-2023



中国船级社

船舶数字化检验数据交换技术指南

2023

生效日期：2023 年 5 月 1 日

北京

前 言

当前，随着数字技术在船舶行业的快速发展，船舶的智能化、数字化程度越来越高，船舶在营运过程中产生了大量数据，为基于数据的船舶检验创造了条件。提供标准的数据交换规则，已经成为行业面临的迫切需求。

为促进船舶数字化检验数据标准化交换，中国船级社结合业界的数据交换需求，编制了《船舶数字化检验数据交换技术指南》。本指南根据《船舶数字化检验应用指南》提出的数字化检验应用方案，在其基础上制定了用于船舶数字化检验的数据结构和数据交换技术规则，以促进船舶数据标准化交换的工作开展，为船舶数字化检验验证提供遵循依据。

本指南由中国船级社编写和更新，通过网页 <http://www.ccs.org.cn> 发布，使用相关方对本指南如有意见或建议可通过电子邮件反馈至 ic@ccs.org.cn。

目 录

第1章 通则.....	1
第1节 一般规定.....	1
第2节 数据范围与类型.....	2
第2章 数据标识与结构.....	4
第1节 一般规定.....	4
第2节 通用数据标识.....	4
第3节 时序数据标识.....	7
第4节 数据结构.....	15
第3章 数据交换接口.....	31
第1节 一般规定.....	31
第2节 接口服务.....	31
第3节 接口与数据安全.....	38
第4章 数据交换实施与测试验证.....	39
第1节 数据交换实施.....	39
第2节 测试验证.....	39
附录1 CCS 船舶数字模型.....	41
附录2 位置编码.....	42
附录3 通用传感器信号类型.....	44
附录4 通用数据元数据结构示例.....	45
附录5 时序数据结构要求与实现.....	47

第1章 通则

第1节 一般规定

1.1.1 适用范围

1.1.1.1 本指南规定了用于船舶数字化检验的数据标准以及数据交换接口规则。适用于申请中国船级社（以下称“CCS”）开展数字化检验的船舶，其他涉及到数据交换的业务可参照本指南执行。

1.1.1.2 申请数字化检验的船舶在向 CCS 分发数据或授权 CCS 访问数据时，应遵循本指南技术要求，包括数据命名格式、数据结构，并通过标准接口协议提供数据。

1.1.2 定义

1.1.2.1 除另有规定外，本指南有关定义如下：

- (1) **数字化**：系指利用信息系统、各类传感器、机器视觉等技术，获取检验对象的原始数据、各种信息和相关知识，形成可识别、可存储、可计算的数据，以建立相关的数据模型，进行处理、分析和应用。
- (2) **检验对象**：系指船舶文件/资料、船体结构、机械/电子设备(系统)等所有船舶检验目标。
- (3) **数据通道**：用于从船舶机械/电子等设备到船载数据服务器或船舶数字系统的数据传输虚拟通道，定义设备运行状态的数据静态特性，如通过数据通道形式表达主机气缸排气温度、燃油进口压力。
- (4) **数据通道ID**：数据通道标识符，用于唯一标识船上数字化设备的数据通道。其用法见本指南第2章第2.3.2。
- (5) **数据通道列表**：定义数据通道ID和数据通道属性的列表，并通过数据所有者进行共享。
- (6) **数据通道属性**：数据通道的属性，如单位和范围。
- (7) **命名空间**：为避免名称冲突而使用的名称集。
- (8) **时间序列数据**：按时间顺序收集的数据集。

注：上述数据通道相关名词的定义适用于对时序数据的表达。

1.1.3 引用文件

1.1.3.1 相关文件中的条款通过本指南的引用将成为本指南的一部分，凡是标注日期的引用文件，仅标注日期的版本适用于本指南。凡是不标注日期的引用文件，其最新版本（包括所有的修改通报、变更通告）适用于本指南。

- (1) ISO 19847:2018 船舶与海上技术 船载海上共享数据服务器(Ships and marine technology — Shipboard data servers to share field data at sea)

注：GB/T 42054 船舶与海上技术 船载海上共享数据服务器(ISO 19847:2018,IDT)

(2) ISO 19848:2018 船舶与海上技术 船载机械设备数据格式(Ships and marine technology — Standard data for shipboard machinery and equipment)

注：GB/T 42055-2022 船舶与海上技术 船载机械设备数据格式(ISO 19848:2018,IDT)

(3) ISO 80000 系列 量和单位(Quantities and units)

(4) ISO 8601 数据元和交换格式·信息交换·日期和时间表示法(Date elements and interchange formats-Information interchange-Representation of dates and times)

注：GB/T 7408-2005 数据元和交换格式·信息交换·日期和时间表示法(ISO 8601:2000,IDT)

(5) W3C XML 扩展标记语言(XML)1.0，W3C 推荐[Extensible Markup Language (XML) 1.0,W3C Recommendation]

(6) W3C XML 模式 第 1 部分：结构，W3C 推荐(XML Schema Part 1:W3C Recommendation)

(7) W3C XML 模式 第 2 部分：数据类型，W3C 推荐(XML Schema Part 2:Datatypes,W3C Recommendation)

(8) RFC 3339 互联网日期和时间：时间戳(Date and Time on the Internet: Timestamps)

(9) RFC 4180 逗号分隔值(CSV)文件的通用格式和 MIME 类型[Common Format and MIME Type for Comma-Separated Values (CSV) Files]

(10) RFC 5234 语法规则的扩展 BNF: ABNF(Augmented BNF for Syntax Specifications: ABNF)

(11) RFC 3986 统一资源标识符(URI)：通用语法(Uniform Resource Identifier URI:Generic Syntax)

1.1.4 缩略语

1.1.4.1 下列缩略语适用于本指南：

(1) HTTP:超文本传输协议(Hypertext Transfer Protocol)

(2) RESTFUL:也称 REST，是一种网络应用程序的设计风格 and 开发方式(Representational State Transfer)

(3) JSON:JavaScript 对象表示法(JavaScript Object Notation)

(4) URI:统一资源标识符(Uniform Resource Identifier)

(5) URL:统一资源定位器(Uniform resource locator)

(6) UTC:世界标准时间(Universal Time Coordinated)

(7) UTF-8:UCS 编码的 8 位传输格式(UCS Transformation Format 8)

(8) XML:可扩展标记语言(Extensible Markup Language)

(9) CSV:逗号分隔型取值格式(Comma Separated Values)

第 2 节 数据范围与类型

1.2.1 数据范围

1.2.1.1 本指南规定的用于船舶数字化检验的相关数据，包括但不限于船舶文件/资料、船体结构、船载机械/电子设备（系统）等检验对象产生多种结构形式的数据。

1.2.2 数据类型

1.2.2.1 根据数字化检验涉及的数据范围，按照检验数据类型分为结构化数据、半结构化数据和非结构化数据，具体如下图 1.2.2.1 所示。

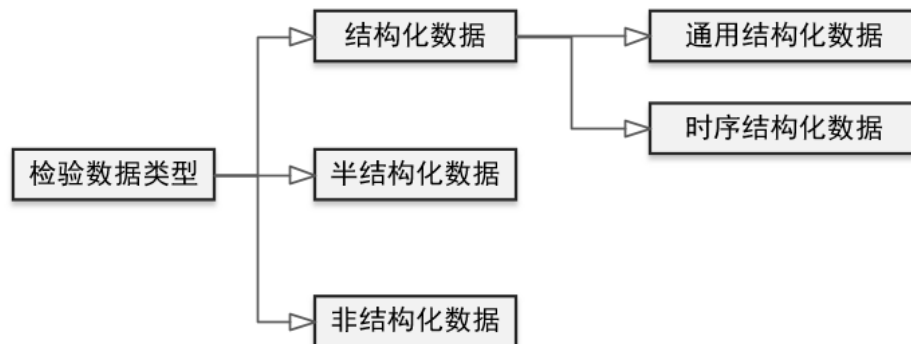


图 1.2.2.1 检验数据类型

1.2.2.2 结构化数据是指可以使用关系型数据库表示和存储，表现为二维的数据特点。例如结构化查询语言（简称“SQL”）数据库。根据数字化检验涉及的数据构成特点，本指南将结构化数据分为通用结构化数据（以下简称“通用数据”）、时序结构化数据（以下简称“时序数据”）。

1.2.2.3 半结构化数据具有基本固定结构模式的特点，属于结构化数据的一种表现形式。例如通过 XML 或 JSON 表示的系统日志、电子记录簿等，本指南在数据组织及数据交换中按照通用数据的方式进行处理。

1.2.2.4 非结构化数据没有固定结构，例如文档、图片、音频和视频等，非结构化数据的表示可通过结构化元数据形式对其进行描述，数据实体作为结构化元数据的元素进行描述和存储，传输时遵循本指南通用数据结构的原则。本指南在数据组织及数据交换中对其按照通用数据的方式进行处理。

1.2.2.5 根据对检验数据类型的梳理和归类，本指南通过第 2 章、第 3 章分别对通用数据和时序数据的命名、数据结构和数据接口进行了详细规定。

第2章 数据标识与结构

第1节 一般规定

2.1.1 一般要求

2.1.1.1 本章规定了通用数据和时序数据的标识方法和数据结构要求。

2.1.1.2 数据标识可识别和区分不同的检验对象，确保相关检验对象在标识命名层面的一致性和唯一性。

2.1.1.3 标识编码是对检验对象数据赋予特定代码的过程。

2.1.1.4 标识编码的原则应满足 CCS《船舶与海上设施数字系统验证指南》2.2.2.1 的要求。

2.1.1.5 船舶设备传感器采集点位对象采用时序数据标识方法。

2.1.1.6 其他检验对象，例如船舶文件资料、电子记录簿等，采用通用数据标识方法。

2.1.1.7 数据结构为独立于数据格式（如 XML、JSON、CSV 等）的逻辑结构规定。

第2节 通用数据标识

2.2.1 一般要求

2.2.1.1 本节规定的通用数据标识是对通用数据类型的标识及属性要求。

2.2.1.2 通用数据基于 CCS 船舶数字模型（简称：emd）进行标识。

2.2.2 通用数据类型标识

2.2.2.1 通用数据类型标识英文代码为 TypeID。

2.2.2.2 通用数据类型标识编码遵循 URI 的格式定义，URI 定义允许多种不同的组合，通用数据类型标识为这些组合的子集，具体格式要求如表 2.2.2.2(1)所示。

通用数据类型标识

表 2.2.2.2 (1)

通用数据类型标识 = [协议]://命名实体/船舶 ID/命名规则/版本/通用数据类型分类码	
命名元素	描述
命名实体	实体机构
船舶 ID	船舶标识
/	路径保留字符
命名规则	标识方案规则，本指南特指 emd
版本	命名规则版本，本指南特指 emd 版本号
通用数据类型分类码	本指南特指 emd 通用数据的分类

(1) 通用数据类型标识组合通过使用在 RFC5234 中定义的扩展 BNF(ABNF)进行定义，在

URI 中“协议”元素是可选的，斜杠(“/”)为描述层次结构的保留字符。

(2) 通用数据类型标识组合详细说明。

① 命名实体

命名实体元素应为产生通用ID的实体拥有或控制的域。

CCS的命名实体为：digitalship.ccs.org.cn

② 船舶ID

(a) 船舶ID是对船舶的通用识别。

(b) 通常，船舶ID宜使用IMO编号（国际航行船舶）或船舶统一识别号CMSA ID（中国籍船舶）。

示例：

——/IMO1234567

——/CN202212345672

③ 命名规则

(a) 命名规则应为数据通道命名规则的指定名称。

(b) 此名称可在命名实体的监督下自由设置，并应在前面具有表示命名实体的符号，以消除重复。

(c) 命名规则是为船上部件和系统定义命名方案（或标识方案）的一组要求。命名规则应定义标识字符串的组成，以及开发标识字符串的方法。

(d) 本指南的命名规则特指 emd，模型获取方法参见附录 1

④ 版本

版本指的是命名规则版本号。

⑤ 通用数据分类码

(a) 通用数据分类码特指 emd 对通用数据的分类编码，分类码的获取参考附录 1。

(b) 通用数据分类码须基于 emd 进行选择。

示例：P1.003.006

该示例表示的是船舶证书分类码。

2.2.2.3 以下是基于 CCS 命名实体给出的命名示例。

示例：船舶证书

通用数据类型标识	http://digitalship.ccs.org.cn/imo1234567/emd/v1.0/P1.003.006
船舶ID	imo1234567
命名实体	digitalship.ccs.org.cn
命名规则	emd
版本	v1.0
通用数据分类码	P1.003.006

注：不要求通用数据类型标识为可解析的URI，即该URI不一定是有效的URL。

2.2.3 通用数据类型属性

2.2.3.1 本指南规定的通用数据类型属性集合如表 2.2.3.1 所示。

通用数据类型属性

表 2.2.3.1

属性代码	属性名称	描述
Data Object Type	数据对象类型	用于标识数据的类别，例如普通文件、媒体文件、结构化数据等
Description Language	描述语言	描述数据对象采用的语言或语法，支持 XML 和 JSON
Object Metadata	对象元数据	是对数据对象元数据结构的描述
Name	名称	用于标识通用数据类型名称
Remarks	备注	

(1) 在实际应用中应按以上属性规定进行描述。

(2) 若以上未列出的属性类型或与本指南规定的明显不同，经 CCS 同意后可在此基础上进行扩展。

2.2.3.2 通用数据类型属性规定说明

(1) 数据对象类型 (Data Object Type)

数据对象类型的属性值遵循表 2.2.3.2(1) 的规定。

数据对象类型属性表

表 2.2.3.2(1)

类型代码	类型名称	描述
Normal File	普通文件	表示证书、报告、图纸、资料等普通版式文件。
Media File	媒体文件	表示音、视频类的媒体文件。
Structured Data	结构化数据	表示简单、或具有一定关系的结构化关系型数据。

数据对象类型属性为强制性属性，如不在此类型代码范围内的特殊数据，经 CCS 同意后可增加新的类型。

示例：

GeneralObject.TypeID	...	GeneralObject.ObjectType	...
.....	...	Normal File	...

(2) 描述语言 (Description Language)

描述语言属性用于规定通用数据类型对象描述采用的语言或语法，本指南支持的语言或语法遵循表 2.2.3.2(2) 的规定。

描述语言属性表

表 2.2.3.2(2)

类型代码	类型名称	描述
XML	可扩展标记语言	表示通过 XML 语言对通用数据对象进行描述。
JSON	JavaScript 对象表示法	表示通过 JSON 语法对通用数据对象进行描述。

描述语言属性为强制性属性，需要明确通用数据类型对象描述采用的语言或语法。

示例：

GeneralObject.TypeID	...	GeneralObject.Language	...
----------------------	-----	------------------------	-----

.....	...	JSON	...
-------	-----	------	-----

(3) 对象元数据 (Object Metadata)

对象元数据实体应采用 Base64 算法进行编码处理，具体对象元数据结构明文示例参见附录 4。

(4) 名称 (Name)

名称属性表示的是通用数据类型对象的名称。

示例：

GeneralObject.TypeID	...	GeneralObject.Name	...
.....	...	船舶电子记录簿	...

(5) 备注 (Remarks)

在备注中可以描述关于目标通用数据类型对象的任意补充信息。

第 3 节 时序数据标识

2.3.1 一般要求

2.3.1.1 时序数据采用数据通道形式标识，数据通道由数据通道 ID 和数据通道属性构成。

2.3.1.2 本节规定的数据通道命名是基于 emd 的数据通道 ID 及属性的标识方法。

2.3.2 数据通道 ID

2.3.2.1 数据通道 ID 包含以下三种表达形式：

——通用 ID (Universal ID)

——本地 ID (Local ID)

——短 ID (Short ID)

(1) 通用ID为通用识别船载数据通道，是以数据交换为目标的通用标识方法，通用ID为全球唯一标识。

(2) 本地ID用于船内识别船载数据通道。例如，船载计算机系统、综合自动化系统(IAS)和报警监测系统(AMS)，均有自己的数据通道列表，由唯一的通道ID组成，该通道ID可对应于本地ID。

(3) 短ID为本地ID的可选短替代标识符。

(4) 这些ID应不区分大小写，以避免意外错误输入。

(5) 对于同一艘船，本地ID和短ID应唯一。对于不同的船舶，相同类型传感器的数据通道具有相同的本地ID。相反，即使数据通道具有相同的含义，每条船舶的短ID也可能不同。

2.3.2.2 通用 ID

(1) 通用 ID 标识遵循 URI 的格式，URI 格式允许多种不同的组合，但通用 ID 为这些组合的子集，具体格式要求如表 2.3.2.2 (1)所示。

通用 ID 命名格式

表 2.3.2.2 (1)

通用 ID	=[协议]://命名实体/船舶 ID/本地 ID	
命名元素	描述	
命名实体	实体机构	
船舶 ID	船舶标识	
/	保留字符	
本地 ID	路径元素 路径元素.....	

(2) 通用ID组合通过使用在RFC5234中定义的扩展BNF(ABNF)进行定义，在URI中“协议”元素是可选的，斜杠("/")为描述层次结构的保留字符。

(3) 通用ID组合详细说明。

① 命名实体

(a) 命名实体元素应为产生通用ID的实体拥有或控制的域。

(b) CCS的命名实体为： digitalship.ccs.org.cn

② 船舶ID

(a) 船舶ID是对船舶的通用识别。

(b) 通常，船舶ID宜使用IMO编号（国际航行船舶）或船舶识别号CMSA ID（中国籍船舶）。

示例：

——/IMO1234567

——/CN202212345672

2.3.2.3 本地 ID

(1) 本地 ID 使用 ABNF 定义，格式规定如表 2.3.2.3 (1)所示。

本地 ID 命名格式

表 2.3.2.3 (1)

本地 ID	=命名规则/版本/船舶设备分类码/设备监测对象/位置/传感器信号类型	
命名元素	描述	
命名规则	标识方案规则，本指南特指 emd	
/	路径标识符	
版本	emd 版本号	
船舶设备分类码	基于 emd 设备分类标识	
设备监测对象	表示设备监测对象及对象编号的组合	

位置	设备监测对象的位置
传感器信号类型	设备传感器信号类型标识

(2) 本地 ID 组合详细说明。

① 命名规则

(a) 命名规则应为数据通道命名规则的指定名称。

(b) 此名称可在命名实体的监督下自由设置，并应在前面具有表示命名实体的符号，以消除重复。

(c) 命名规则是为船上部件和系统命名方案（或标识方案）的一组要求。命名规则应规定标识字符串的组成，以及开发标识字符串的方法。

(d) 本指南的命名规则特指 **emd**。模型获取方法参见附录 1。

② 版本

版本指的是命名规则的版本号。

③ 船舶设备分类码

(a) 船舶设备分类码特指 **emd** 设备分类标识码，模型分类码的获取参考附录 1。

(b) 船舶设备分类码须基于 **emd** 进行选择。

格式：主设备编码.子设备编码+编号

示例：P4.003.003+1

该示例表示的是 1 号柴油机，其中“+编号”为可选项，适用于多个同类设备的情况。

④ 设备监测对象

(a) 设备监测对象适用于通过船舶设备分类码无法完全表达清楚数据通道的情况。

(b) 如果设备监测对象为多级对象组合表达，对象之间用“.”进行区分。

(c) 该项为非必填项，对象名称原则上采用英文单词或词组进行表达，但单词之间不能包含空格以及 URI 约定的 / ? # [] @ 等保留字符。

格式：设备监测对象.设备监测对象+编号。

示例：CYLINDER.EXHAUST_GAS+2

该示例表示的是 2 号气缸废气，其中“+编号”为可选项，适用于多个同类监测对象的情况。

⑤ 位置

位置标识的是设备监测对象所处的位置。位置编码获取参见附录 2，优先使用附录 2 中的位置编码，如不在附录 2 中允许自行定义，推荐采用有意义的位置英文单词，经 CCS 评估确认后可纳入使用。

示例：Inlet

⑥ 传感器信号类型

传感器信号类型表达的是监测对象传感器的类型代码。传感器信号获取参见附录 3，优先

使用附录 3 中的信号类型,如不在附录 3 中允许自行定义,推荐采用有意义的英文单词。

示例: Speed

2.3.2.4 短 ID

(1) 短 ID 是本地 ID 的可选短替代。数据通道和短 ID 之间必须有一对一的对应关系,因此短 ID 对于一艘船是唯一的。

(2) 短ID的规定如下:

短ID	=非保留字符
-----	--------

(3) 短ID应尽量简短,并使用机器友好符号、用户友好短语或这些符号和短语的组合表示。

示例:

——0001

——TAH001

——ME001_RPM

2.3.2.5 以下是基于 CCS 命名实体给出的命名示例。

示例: 1 号发电机组 2 号缸排气温度

通用ID	http://digitalship.ccs.org.cn/IMO1234567/emd/v1.0/P8.003.006 +1/Cylinder. Exhaust_Gas+2/Outlet/Temperature
本地ID	emd/v1.0/P8.003.006 +1/Cylinder.Exhaust_Gas+2/ Outlet/Temperature
短 ID	0001
船舶ID	IMO1234567
命名实体	digitalship.ccs.org.cn
命名规则	emd
本地名称	G/E EXHAUST GAS TEMP. CYL.2

注: 不要求通用ID为可解析的URI, 即该URI不一定是有效的URL。

2.3.3 数据通道属性

2.3.3.1 本指南规定的的数据通道属性分类如表 2.3.3.1 所示。

数据通道属性表

表 2.3.3.1

属性代码	属性名称	描述
Data Channel Type	数据通道类型	标识数据通道的类型
Format	格式	描述数据格式
Range	范围	描述数据范围, 如值的上下限
Unit	单位	用于测量值的单位和量
Quality Coding	质量编码	表示数据质量评价模式的名称
Name	名称	描述为船载控制系统和其他仪器分配的名称
Remarks	备注	

- (1) 在实际应用中应按以上属性规定进行描述。
- (2) 若以上未列出的属性类型或与本指南规定的明显不同，经 CCS 同意后可在此基础上进行扩充。

2.3.3.2 通道属性规定说明

(1) 数据通道类型 (Data Channel Type)

① 数据通道类型用于标识数据通道的类型，例如测量值、平均值、报警和状态。数据通道类型由以下子属性组成：

——类型 (Type)

——更新频率 (Update Cycle)

——计算周期 (Calculation Period)

② 其中类型 (Type) 子属性规定了数据通道的类型，属性的值遵循表 2.3.3.2 (1) 中的规定。

类型子属性表

表 2.3.3.2(1)

类型代码	类型名称	描述
Inst	测量值	在某一时间点的测量值。
Average	平均值	在某一段时间内的平均值。 “平均值”不是指多个传感器在同一时间的值的平均值，而是指单个传感器的时间序列值的平均值。
Max	最大值	在某一段时间内的最大值。 “最大值”不是指多个传感器同一时间的值的最大值，而是指单个传感器的时间序列值的最大值。
Min	最小值	在某一段时间内的最小值。 “最小值”不是指多个传感器同一时间的值的最小值，而是指单个传感器的时间序列值的最小值。
StandardDeviation	标准差	某一段时间内数值的标准偏差。 “标准偏差”不是指多个传感器在同一时间的值的标准差，而是指单个传感器的时间序列值的标准差。
Calculated	计算值	通过计算而不是测量得到的值。
SetPoint	目标值	自动控制目标值。
ControlOutput	控制输出	自动控制的操纵值。
Alert	报警	可获得的警报值。
Status	状态	可获得的状态值
ManuallyInput	手工输入	船员输入的值。这里假设的值为指示器的读数。

③ 更新频率表示更新测量值的频率，此子属性应在定期更新测量值时使用。

④ 当一个数据通道的值为使用特定时间段的测量值计算的结果时，应使用计算周期表示该时间段。

⑤ 更新频率和计算周期应以大于 0 的十进制数字描述，更新频率和计算周期的单位应为“秒”。

⑥ 类型子属性是强制的，其他属性是可选的。

示例：

DataChannelID.LocalID	...	DataChannelProperty. DataChannelType. Type	DataChannelProperty. DataChannelType. UpdateCycle	DataChannelProperty. DataChannelType. CalculationPeriod
emd/v1.0/ P8.003.006+1/ALTERNATO R_BEARING// TEMPERATURE	...	Inst	5	60

(2) 格式(Format)

① 格式为用于描述数据格式，并由以下子属性构成。

——类型(Type)

——限定(Restriction)

② 类型子属性为强制性，限定子属性为可选，格式属性下可能存在多个限定子属性。

③ 可用的类型如表 2.3.3.2 (2)所示，这些数据类型的规定符合 W3C XML 架构定义语言 (XSD) 1.1 第 2 部分：数据类型。

类型子属性表

表 2.3.3.2(2)

类型代码	类型名称	描述
Decimal	带小数位的数字	带小数位的数字表示实数的子集，实数可用十进制数表示。小数的值为整数除以 10 的非负幂得到的一组数字，即可表示为 $i/10^n$ ，其中 i 和 n 为整数， $n \geq 0$ 。精确度未反映在该值空间中，数字 2.0 与数字 2.00 无区别。十进制的顺序关系为实数的顺序关系，仅限于该子集。
Integer	整数	整数通过将小数位数的值固定为 0 并不允许尾随小数点，从十进制中派生。这就产生了整数的标准数学概念。整数的值空间为无穷集 $\{\dots, -2, -1, 0, 1, 2, \dots\}$ 。整数的基类型为十进制。
Boolean	布尔型	布尔值表示二值逻辑的值。
String	字符型	字符串数据类型表示 XML 中的字符串。
DateTime	日期时间	日期和时间数据类型用于包含日期和时间的值。 格式应遵循 ISO 8601 “YYYY-MM-DDThh:mm:ssZ”，其中： YYYY 表示年份； MM 表示月份； DD 表示日期； T 表示所需时间段的开始； hh 表示小时； mm 表示分钟； ss 表示秒； Z 表示 UTC 时钟。

④ W3C XML 模式中定义的限定见表 2.3.3.2 (3)可用于规定可接受的值。每个数据类型的验证规则和可用约束应遵循 W3C XML 模式。

约束子属性表

表 2.3.3.2(3)

限定	描述	数据类型
Enumeration	定义可接收枚举值列表	字符串
FractionDigits	指定允许的最大小数位数。必须等于或大于零。	非负整数
Length	指定允许的字符或列表项的精确数目。必须等于或大于零。	非负整数
MaxExclusive	定义数值的上限（允许值必须小于该值）。	来自{基本类型定义}的 • 值空间 • 的值
MaxInclusive	定义数值的上限（允许值必须小于或等于该值）。	来自{基本类型定义}的 • 值空间 • 的值
MaxLength	定义所允许的字符或者列表项目的最大数目。必须大于或等于 0。	非负整数
MinExclusive	定义数值的下限。所允许的值必需大于此值。	来自{基本类型定义}的 • 值空间 • 的值
MinInclusive	定义数值的下限。所允许的值必需大于或等于此值。	来自{基本类型定义}的 • 值空间 • 的值
MinLength	定义所允许的字符或者列表项目的最小数目。必须大于或等于 0。	非负整数
Pattern	定义可接受的字符的确切序列。	字符串
TotalDigits	指定允许的确切位数。必须大于零。	正整数
WhiteSpace	指定如何处理空值（直线、制表符、空格和回车）。	

示例：

DataChannelID.LocalID	...	DataChannelProperty. Format.Type	DataChannelProperty. Format.Restriction
.....	...	Integer	MaxExclusive 5000

(3) 范围(Range)

① 范围用于描述数据范围。范围属性由以下子属性组成。

——低(Low)

——高(High)

② “低”子属性表示模拟数据的下限，“高”子属性表示其上限。

③ 子属性值的数据类型为带精度的数值或空。只有在无法指定下限和/或上限时，子属性的值才能为空。

④ 范围的高值必须大于且不等于低值。

⑤ 范围属性仅对模拟数据是强制的。

⑥ 范围不表示值的上下边界。若传感器出现故障或其他异常情况，则该值可能会超出范围。

示例：

DataChannelID.LocalID	...	DataChannelProperty. Range.Low	DataChannelProperty. Range.High
.....	...	0	120

(4) 单位(Unit)

① 单位属性应用于测量值的单位和量。单位属性由以下子属性组成。

——单位符号(Unit Symbol)

——计量名称(Quantity Name)

② ISO 80000 中定义的单位符号和量名称应用于单位符号和量名称子属性。

“单位符号”子属性表示测量值的单位符号（例如，“m”表示长度，“kg”表示质量）。

③ 计量名称子属性可用于标识为数据通道的物理或化学项测量或计算的变量值。

④ 测量值应根据本属性中规定的单位进行描述。

示例：

DataChannelID.LocalID	...	DataChannelProperty. Unit.UnitSymbol	DataChannelProperty. Unit.QuantityName
.....	...	kW	功率

(5) 质量编码(Quality Coding)

质量编码表示数据质量评价模式的名称，该模式应能区分数据通道的有效测量值和无效测量值。

示例：

DataChannelID.LocalID	...	DataChannelProperty.O pcQuality	...
.....	...	IEC 61162-STATUS	...

注：在“IEC 61162—STATUS”下，使用“A”（数据有效）和“V”（数据无效）表示数据质量。

(6) 名称(Name)

名称可以用来描述为船载控制系统和其他仪器分配的名称。

示例：

DataChannelID.LocalID	...	DataChannelProperty.Name	...
.....	...	主机转速	...

(7) 备注(Remarks)

在备注中，描述了数据通道上的任意补充信息。

示例：

DataChannelID.LocalID	...	DataChannelProperty.Remarks	...
.....	...	位置：集控室, 制造商：AAA 公司, 型号：型号—AAA	...

第 4 节 数据结构

2.4.1 一般要求

2.4.1.1 本节规定的是对通用和时序数据的数据结构要求。

2.4.1.2 数据结构可以通过使用 XML、JSON 和 CSV 格式来描述。

2.4.1.3 本节给出了用于定义数据结构且独立于实现语言的标准数据类型。

2.4.1.4 标准数据类型可替换为在实现语言中定义的数据类型。

2.4.1.5 本节规定的通用数据列表结构和时序数据通道列表结构，目的是在传递数据前需要根据对应列表格式要求将相关通用数据列表或者数据通道列表分享或提交给 CCS，CCS 在接收具体数据时将根据列表清单对数据进行识别和解析。

2.4.2 实现语言

2.4.2.1 本节中规定的通用数据结构格式示例参照附录 4。

2.4.2.2 本节中规定的时序数据结构应按照附录 5 的要求实施。

2.4.3 标准数据类型

2.4.3.1 为定义数据结构，使用了从表 2.4.3.1 中列出的标准建模语言(UML)基本类型派生的以下标准数据类型。

标准数据类型

表 2.4.3.1

标准数据类型	基本类型	约束	备注
Integer	Integer		
NonNegativeInteger	Integer	> -1	
PositiveInteger	Integer	> 0	
Real	Real		
Boolean	Boolean		
String	String		
DateTime	String	格式遵循 ISO 8601	ABNF 表达式见 RFC3339
Null			空值说明缺少值（可用于任何数据类型）

2.4.4 通用数据列表结构

2.4.4.1 数据模型

通用数据列表结构应包含以下 5 种要素：

(1) 包(Package)

包是由包头和数据主体组成的数据包。

(2) 包头(Header)

包头表示何时创建通用数据通道列表以及由谁创建。

(3) 通用数据列表(GeneralDataList)

通用数据列表由一系列通用数据类型标识和通用数据类型属性组成。

(4) 通用数据类型标识(TypeID)

通用数据类型标识为本指南第 2 章第 2 节中规定的一条通用数据类型标识。

(5) 通用数据类型属性(Property)

通用数据类型属性是对通用数据类型的属性规定。

这些元素按图 2.4.4.1 所示的层次结构排列。

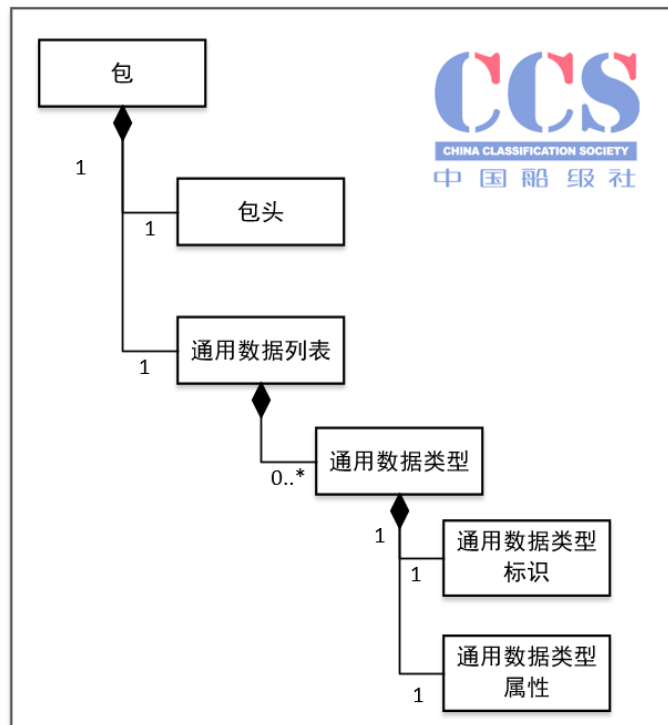


图2. 4. 4. 1 通用数据列表结构模型

2.4.4.2 逻辑结构

(1) 通用数据类型列表结构的逻辑结构如图 2.4.4.2 所示。

(2) 通用数据类型列表结构体现为数据包，包元素由包头和通用数据类型列表元素组成，通用数据类型元素包括通用数据类型标识和属性的元素。

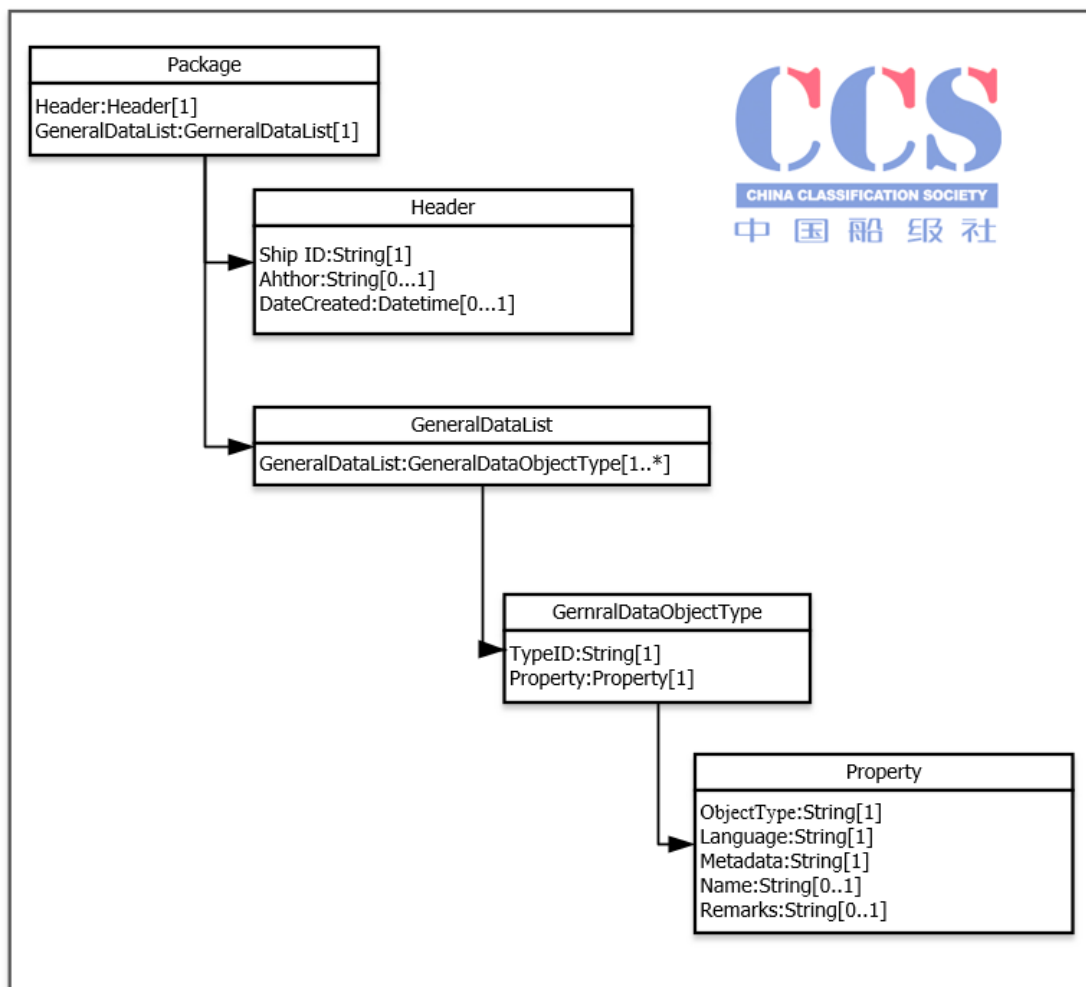


图2.4.4.2 通用数据列表结构逻辑图

2.4.4.3 每种元素的详细信息描述如下。

(1) 包结构(Package)

包结构

表 2.4.4.3(1)

名称	数据类型	备注	强制/可选	最大数
Header	(2) Header	见(2)	强制	1
GeneralDataList	(3) GeneralDataList	见(3)	强制	1

(2) 包头结构(Header)

包头结构

表 2.4.4.3(2)

名称	数据类型	备注	强制/可选	最大数
ShipId	String	船舶标识	强制	1
Author	String	列表所有者	可选	1
DataCreated	Datetime	创建包的日期	可选	1

(3) 通用数据列表(GeneralDataList)

数据通道列表结构

表 2.4.4.3(3)

名称	数据类型	备注	强制/可选	最大数
GeneralDataList	(4) GeneralDataObjectType	见(4)	强制	*

(4) 通用数据类型(GeneralDataObjectType)

数据通道结构

表 2.4.4.3(4)

名称	数据类型	备注	强制/可选	最大数
TypeID	String	见第 2 章第 2.2.2	强制	1
Property	5) Property	见(5)	强制	1

(5) 通用数据属性结构(Property)

属性结构

表 2.4.4.3(5)

名称	数据类型	备注	强制/可选	最大数
ObjectType	String	见第 2 章第 2.2.3.2 (1)	强制	1
Language	String	见第 2 章第 2.2.3.2 (2)	强制	1
Metadata	String	见第 2 章第 2.2.3.2 (3)	强制	1
Name	String	见第 2 章第 2.2.3.2 (4)	可选	1
Remarks	String	见第 2 章第 2.2.3.2 (5)	可选	1

2.4.5 通用数据结构

2.4.5.1 数据模型

通用类型数据应由以下要素组成:

(1) 包(Package)

包元素由包头（元数据）和通用数据（主数据体）组成的数据。

(2) 包头(Header)

① 包头元素包含必要的元数据，可表示何时创建数据、由谁创建的数据。

② 如果包头元素通过任何方式提前提供，则包头元素本身可以被省略。

(3) 通用类型数据(GeneralTypeData)

① 通用类型数据元素是包的主体，包含本章第2节中给出的通用类型数据。

② 通用类型数据元素还表示一组数据集。一旦更新了通用数据列表，并且更新会影响数据的值时，则应新生成该元素。

(4) 数据集(DataSet)

数据集是一组特定通用数据类型标识所组成的对象元数据结构的数据集

这些元素按图 2.4.5.1 所示的层次结构排列。

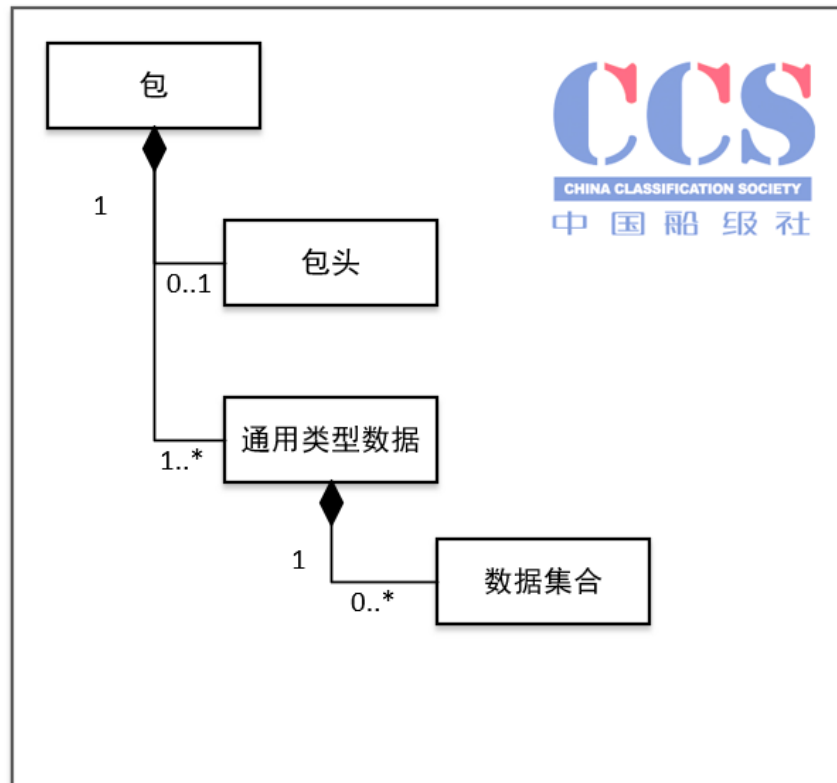


图2.4.5.1 通用类型数据模型

2.4.5.2 逻辑结构

- (1) 通用类型数据应具有图 2.4.5.2 所示的逻辑结构。
- (2) 通用类型数据体现为数据包。包具有包头和通用类型数据构成。

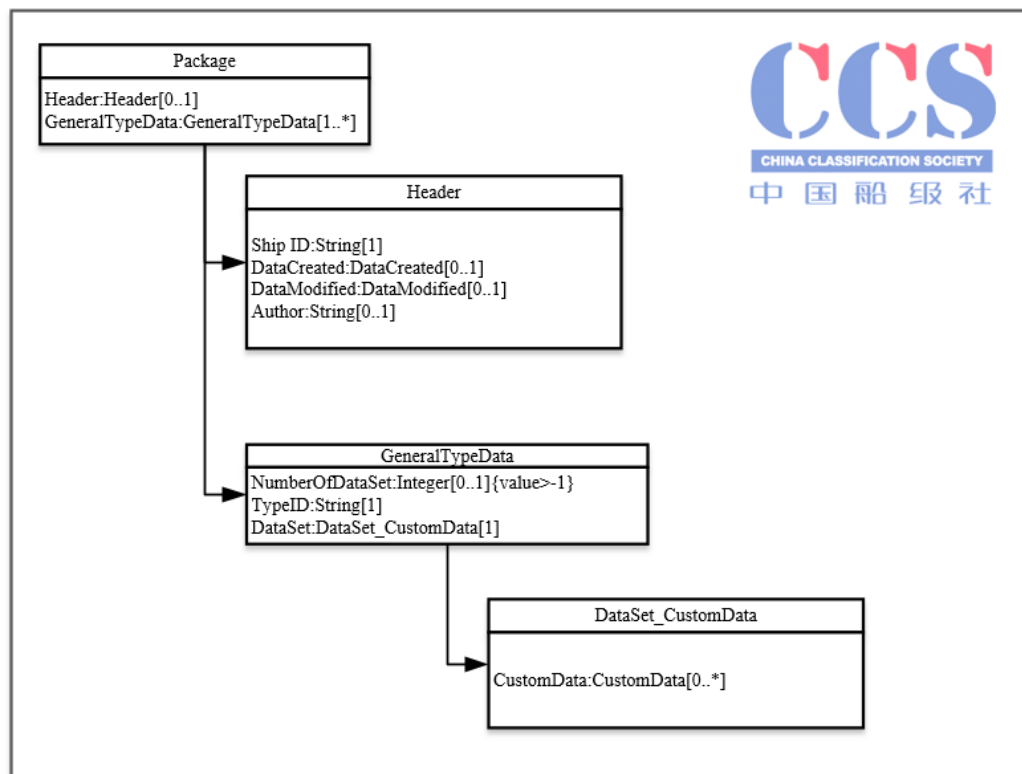


图2.4.5.2 通用类型数据逻辑图

2.4.5.3 每种元素的详细信息描述如下。

(1) 包结构(Package)

包结构 表 2.4.5.3(1)

名称	数据类型	备注	强制/可选	最大数
Header	2)Header	见(2)	可选	1
GeneralTypeData	3)GeneralTypeData	见(3)	强制	*

(2) 包头结构(Header)

包头结构 表 2.4.5.3(2)

名称	数据类型	备注	强制/可选	最大数
ShipId	String	IMO 编号、CMSA ID 等	强制	1
DataCreated	DateTime	包的创建日期	可选	1
DataModified	DateTime	包的修改日期	可选	1
Author	String	数据所有者	可选	1

(3) 通用类型数据结构(GeneralTypeData)

通用类型数据结构 表 2.4.5.3(3)

名称	数据类型	备注	强制/可选	最大数
NumberOfDataSet	NonNegativeInteger	表格数据元素中的数据集数	可选	1
TypeID	String	第 2.2.2 中的一组通用数据类型标识的数组。	强制	1
DataSet	(4) CustomData	见第 2.4.5.1	强制	*

(4) 数据集_自定义数据结构(CustomData)

数据集_自定义数据结构 表 2.4.5.3(4)

名称	数据类型	备注	强制/可选	最大数
CustomData	CustomData	用户自定义格式数据根据第 2.2.3.2 (3) 定义的自定义元数据结构组织数据	强制	*

注：* 表示最大数不限。

2.4.6 时序数据通道列表结构

2.4.6.1 数据模型

数据通道列表应包含以下 5 种要素：

(1) 包(Package)

包是由包头和数据通道(主数据体)组成的数据包。

(2) 包头(Header)

包头表示何时创建数据通道列表以及由谁创建。

(3) 数据通道(DataChannel)

数据通道由属性和数据通道 ID 组成。

(4) 数据通道 ID(DataChannelID)

数据通道 ID 为本指南第 2 章第 2.3.2 中规定的一种数据通道标识符。

(5) 属性(Property)

属性定义本指南第 2 章第 2.3.3 中规定的数据通道的属性。

这些元素按图 2.4.6.1 所示的层次结构排列。

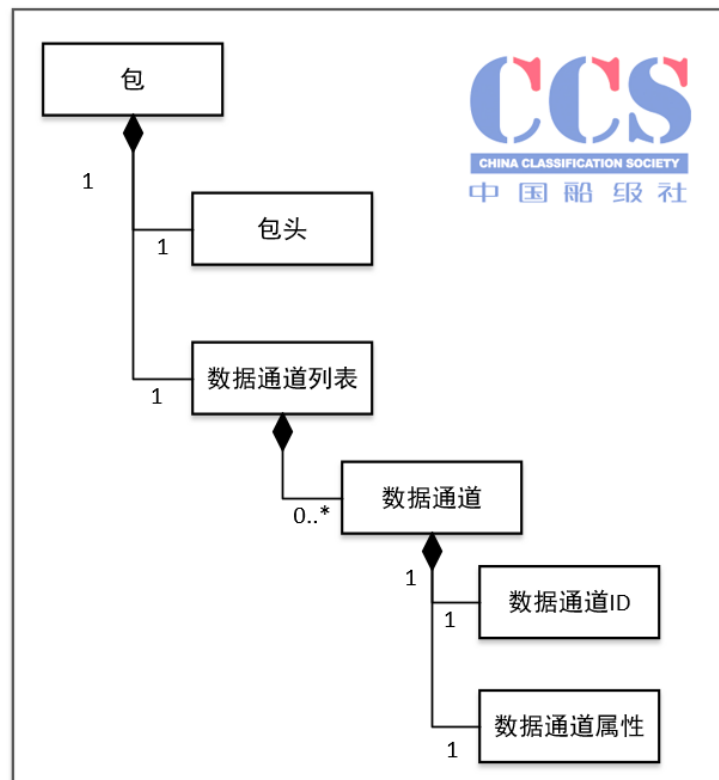


图2.4.6.1 数据通道列表模型

2.4.6.2 逻辑结构

(1) 数据通道列表的逻辑结构如图 2.4.6.2 所示。

(2) 数据通道列表体现为数据包，包元素由包头和数据通道列表元素组成，数据通道列表元素包括一个或多个具有其 ID 和属性的数据通道元素。

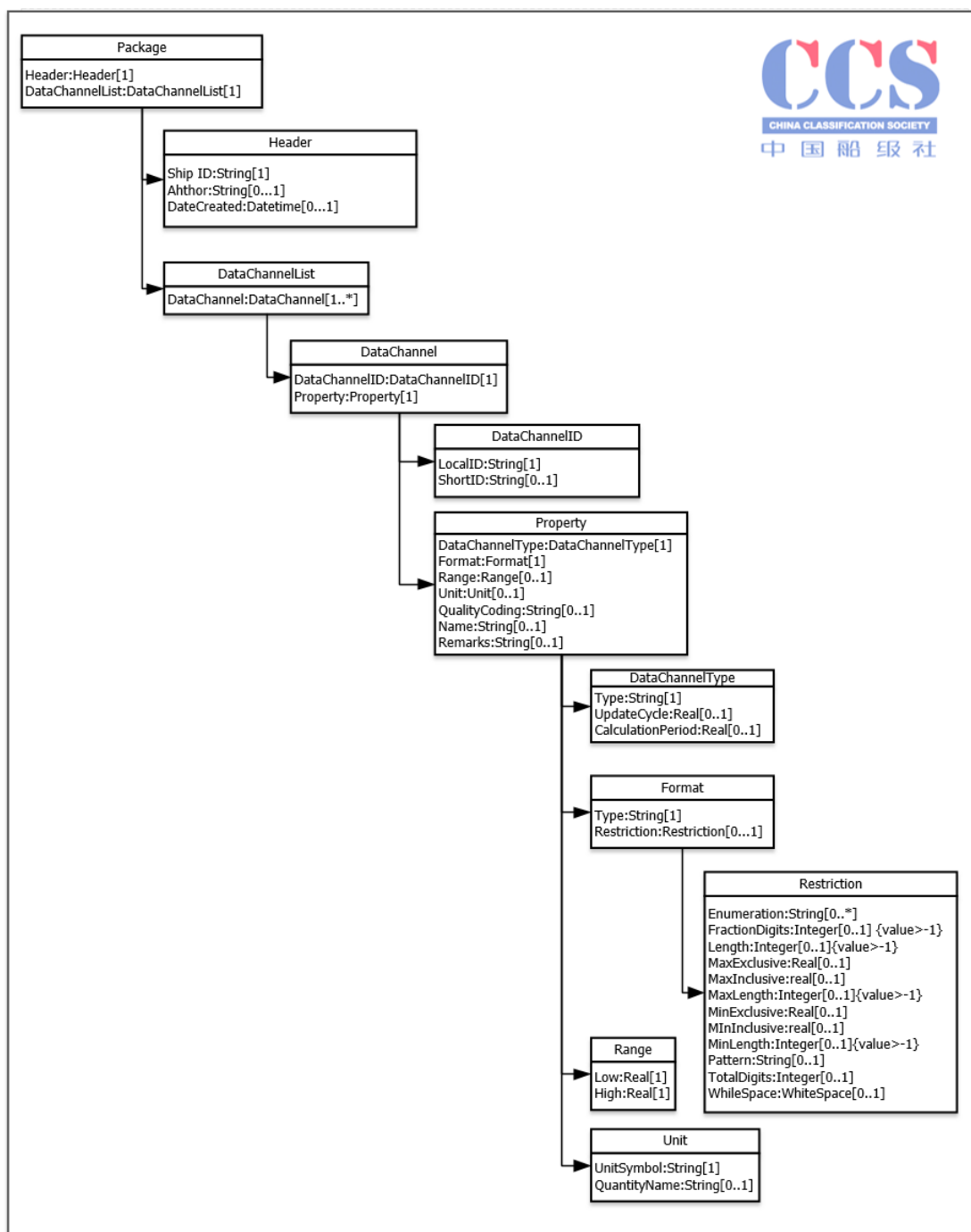


图2.4.6.2 数据通道列表的逻辑图

2.4.6.3 每种元素的详细信息描述如下。

(1) 包结构(Package)

包结构

表 2.4.6.3(1)

名称	数据类型	备注	强制/可选	最大数
Header	(2)Hdr	见(2)	强制	1
DataChannelList	(3)DataChannelList	见(3)	强制	1

(2) 包头结构(Header)

包头结构

表 2.4.6.3(2)

名称	数据类型	备注	强制/可选	最大数
ShipId	String	船舶标识	强制	1
Author	String	数据通道列表所有者	可选	1
DataCreated	Datetime	创建包的日期	可选	1

(3) 数据通道列表结构(DataChannelList)

数据通道列表结构

表 2.4.6.3(3)

名称	数据类型	备注	强制/可选	最大数
DataChannel	(4) DataChannel	见(4)	强制	*

(4) 数据通道结构(DataChannel)

数据通道结构

表 2.4.6.3(4)

名称	数据类型	备注	强制/可选	最大数
DataChannelID	(5) DataChannelID	见(5)	强制	1
Property	(6) Property	见(6)	强制	1

(5) 数据通道 ID 结构(DataChannelID)

数据通道 ID 结构

表 2.4.6.3(5)

名称	数据类型	备注	强制/可选	最大数
LocalID	String	见第 2 章第 2.3.3.2	强制	1
ShortID	String	见第 2 章第 2.3.3.3	可选	1

(6) 属性结构(Property)

属性结构

表 2.4.6.3(6)

名称	数据类型	备注	强制/可选	最大数
DataChannelType	(7) DataChannelType	见(7)和第 2 章第 2.3.3.2(1)	强制	1
Format	(8)Format	见(8)和第 2 章第 2.3.3.2(2)	强制	1
Range	(10)Range	见(10)和第 2 章第 2.3.3.2(3)	强制*	1
Unit	(11)Unit	见(11)和第 2 章第 2.3.3.2(4)	强制*	1
QualityCoding	String	见第 2 章第 2.3.3.2(5)	可选	1
Name	String	见第 2 章第 2.3.3.2(6)	可选	1
Remarks	String	见第 2 章第 2.3.3.2(7)	可选	1
* 仅当数据通道类型为“小数”时才需要范围和单位。				

(7) 数据通道类型结构(DataChannelType)

数据通道类型结构

表 2.4.6.3(7)

名称	数据类型	备注	强制/可选	最大数
Type	String	按照第 2 章第 2.3.3.2(1)所示的数据通道类型格式描述	强制	1
UpdateCycle	Real		可选	1
CalculationPeriod	Real		可选	1

(8) 格式结构(Format)

格式结构

表 2.4.6.3 (8)

名称	数据类型	备注	强制/可选	最大数
Type	String	按照第 2 章第 2.3.3.2(2)所示的数据通道类型格式描述	强制	1
Restriction	(9)Restriction		可选	1

(9) 约束结构(Restriction)

约束结构

表 2.4.6.3 (9)

名称	数据类型	备注	强制/可选	最大数
Enumeration	String	按照第 2 章第 2.3.3.2(2)所示的数据通道类型格式描述	可选	*
FractionDigits	Integer		可选	1
Length	Integer		可选	1
MaxExclusive	Real		可选	1
MaxInclusive	Real		可选	1
MaxLength	Integer		可选	1
MinExclusive	Real		可选	1
MinInclusive	Real		可选	1
MinLength	Integer		可选	1
Pattern	String		可选	1
TotalDigits	Integer		可选	1
WhiteSpace	下列之一： “Preserve” “Replace” “Collapse”		可选	1

(10) 范围结构(Range)

范围结构

表 2.4.6.3(10)

名称	数据类型	备注	强制/可选	最大数
Low	Real	测量范围下限	可选	1
High	Real	测量范围上限	可选	1

(11) 单位结构(Unit)

单位结构

表 2.4.6.3(11)

名称	数据类型	备注	强制/可选	最大数
UnitSymbol	String	在 ISO 80000 或第 2 章第	强制	1

		2.3.3.2 (4)		
QuantityName	String	度量名称在 ISO 80000 中定义	可选	1

2.4.7 时序数据结构

2.4.7.1 数据分类

(1) 时序数据分类

① 时序数据为测量值的集合，其中所有的值均有相应的测量时间，通常这些数据是按时间顺序排列和记录的。

② 在实际应用当中，为了有效的数据传输，本节规定了两种不同的时序数据表示法，分别为表格数据和事件数据，根据测量值的更新间隔合理使用不同的表达方法。

(2) 表格数据

① 表格数据是以定期间隔报告的固定数量的值的向量。

② 数据通道列表规定了时间间隔。

示例：

——多个原始数值从传感器/发射器在同一时间采样

——定期进行的计算结果（如时间平均值、标准偏差等）

③ 测量值按时间戳分组，表格数据形式如表 2.4.7.1(1)所示。

数据示例

表 2.4.7.1(1)

时间戳	数据通道 1	数据通道 2	数据通道 3	数据通道 4
2022-01-01T00:00:00Z	101.2	0.30	10.2	关
2022-01-01T00:00:01Z	0.0	0.30	10.2	关
2022-01-01T00:00:02Z	110.9	0.32	10.2	开

(3) 事件数据

① 事件数据是一组特定时间不固定的数据。

示例：

——报警信息

——状态信息

——手动输入信息

② 事件数据通常测量值根据时间戳和数据通道 ID 以及表 2.4.7.1(2)等形式进行分组。

事件数据示例

表 2.4.7.1(2)

时间戳	数据通道 ID	值
2022-01-01T00:00:00Z	数据通道 1	101.2
2022-01-01T00:00:02Z	数据通道 1	110.9
2022-01-01T00:00:02Z	数据通道 2	0.32
2022-01-01T00:00:02Z	数据通道 4	OPEN

(4) 时序数据组成

- ① 表格/事件数据的每一行称为数据集(DataSet)。
- ② 表格数据由一组不带数据通道 ID 的数据集组成，值可以按其时间顺序标识。
- ③ 表 2.4.7.1(1)的表格数据如下所示：

```
DataSet("2022-01-01T00:00:00Z","101.2","0.30","10.2","CLOSE")
DataSet("2022-01-01T00:00:01Z","0.0","0.30","10.2","CLOSE")
DataSet("2022-01-01T00:00:02Z","110.9","0.32","10.2","OPEN")
```

- ④ 事件数据由具有数据通道 ID 的数据集的集合组成，因为在特定时间不能固定多个值。
- ⑤ 表 2.4.7.1(2)的事件数据如下所示：

```
DataSet("2022-01-01T00:00:00Z ", "数据通道 1", "101.2")
DataSet("2022-01-01T00:00:02Z ", "数据通道 1", "110.9")
DataSet("2022-01-01T00:00:02Z ", "数据通道 2", "0.32" )
DataSet("2022-01-01T00:00:02Z ", "数据通道 4", "OPEN" )
```

2.4.7.2 数据模型

时间序列数据应由以下要素组成：

(1) 包(Package)

包元素由包头（元数据）和时间序列数据（主数据体）组成的数据。

(2) 包头(Header)

① 包头元素包含必要的元数据，用于将包中的数据链接到数据通道列表中的正确通道和现实世界中的时间。

② 可表示何时创建时间序列数据、谁创建的时间序列数据、时间序列数据的测量周期。

③ 如果包头元数据通过任何方式提前提供，则包头元素本身可以被省略，或者只是由短ID列表组成。

(3) 时序数据(TimeSeriesData)

① 时间序列数据元素是包的主体，包含本章第3节中规定的时间序列数据。

② 时间序列数据元素还表示一组数据集，并按数据通道列表版本分组。一旦更新了数据通道列表，并且更新会影响时间序列数据的值，则应重新生成该元素。

(4) 表格数据(TabularData)

① 表格数据元素包含本章第2.4.7.1(2)中规定的值列表。

② 表格表示以相同时间戳和更新间隔报告的测量值的有序列表。

③ 表格数据应按更新间隔进行分组。

(5) 事件数据(EventData)

事件数据元素包含一个（通常）不定期更新的值列表，其中为每个数据点提供了对数据通道列表（本地ID或短ID）的引用，详细信息见本章第2.4.7.1(3)。

(6) 数据集(DataSet)

数据集是一组具有相同时间戳的测量值，有两种类型的数据集，一种用于表格数据，另一种用于事件数据。

这些元素按图2.4.7.2所示的层次结构排列。

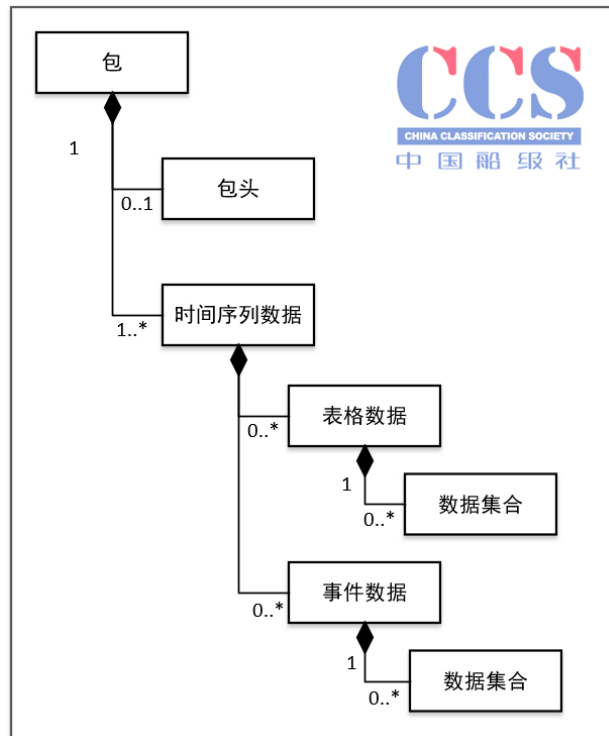


图2.4.7.2 时间序列数据模型

2.4.7.3 逻辑结构

- ① 时间序列数据应具有图 2.4.7.3 所示的逻辑结构。
- ② 时间序列数据体现为数据包，包由包头和时间序列数据构成，时间序列数据包括一个或多个数据集，数据集具有一个时间戳和一个或多个数据。

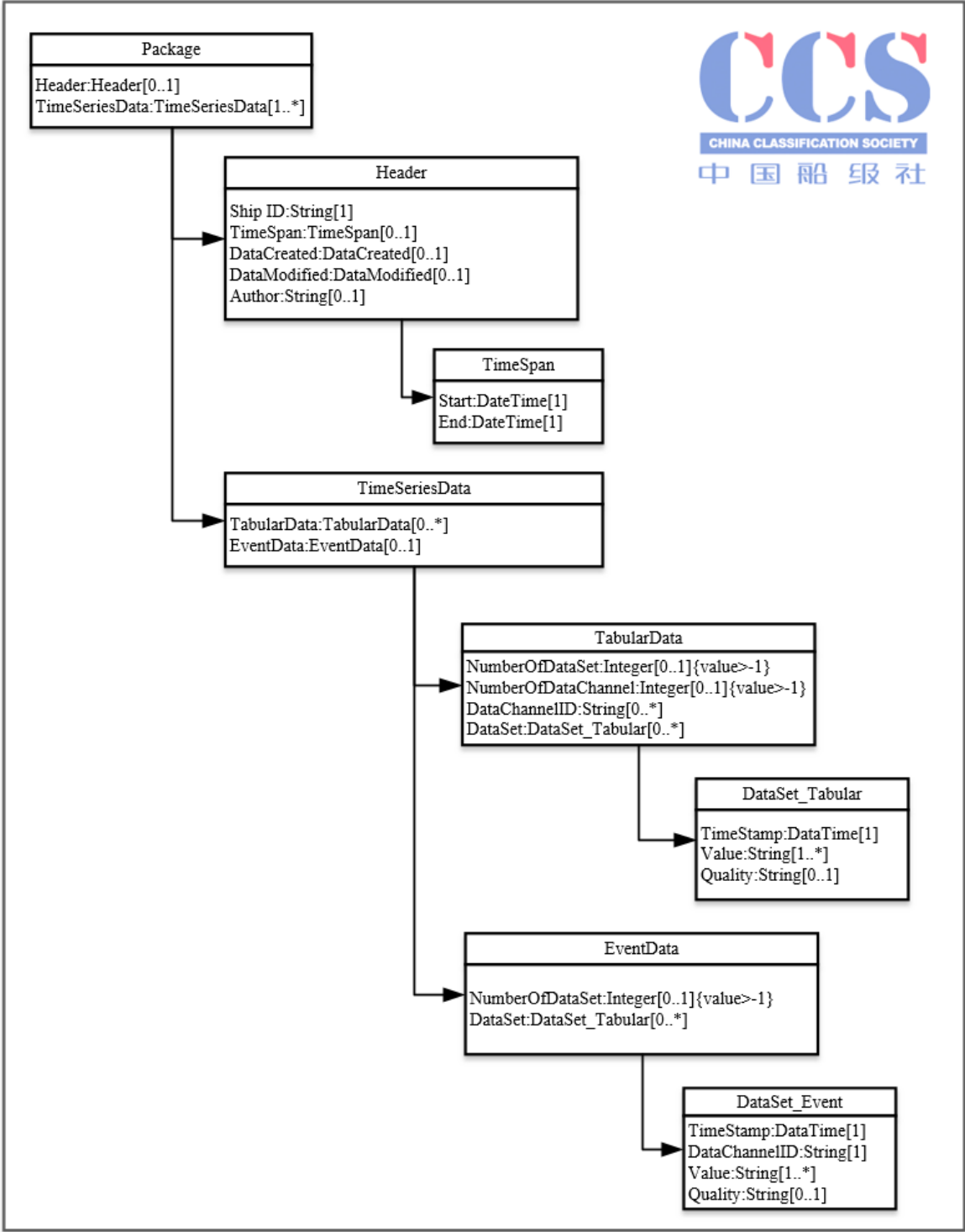


图 2.4.7.3 时间序列数据逻辑图

2.4.7.4 每种元素的详细信息描述如下。

(1) 包结构(Package)

包结构 表 2.4.7.4(1)

名称	数据类型	备注	强制/可选	最大数
----	------	----	-------	-----

Header	(2)Header	见(2) 对于定期数据交换,可省略此元素以减小数据大小	可选	1
TimeSeriesData	(4) TimeSeriesData	见(4) 时间序列数据按数据通道列表版本分组	强制	*

(2) 包头结构(Header)

包头结构

表 2.4.7.4(2)

名称	数据类型	备注	强制/可选	最大数
ShipId	String	IMO 编号、CMSA ID 等	强制	1
TimeSpan	(3)TimeSpan	见(3)	可选	1
DataCreated	DateTime	包的创建日期	可选	1
DataModified	DateTime	包的修改日期	可选	1
Author	String	数据所有者	可选	1

(3) 时间间隔结构(TimeSpan)

时间间隔结构

表 2.4.7.4(3)

名称	数据类型	备注	强制/可选	最大数
Start	DateTime	数据集的起始时间戳	强制	1
End	DateTime	数据集的结束时间戳	强制	1

(4) 时间序列数据结构(TimeSeriesData)

时间序列数据结构

表 2.4.7.4(4)

名称	数据类型	备注	强制/可选	最大数
TabularData	(5) TabularData	见(5)	可选	*
EventData	(6) EventData	见(6)	可选	1

(5) 表格数据结构(TabularData)

表格数据结构

表 2.4.7.4(5)

名称	数据类型	备注	强制/可选	最大数
NumberOfDataSet	NonNegativeInteger	表格数据元素中的数据集合数	可选	1
NumberOfDataChannel	NonNegativeInteger	表格数据元素中的数据通道数	可选	1
DataChannelID	String	第 2 章第 2.3.2 的一组数据通道 ID 的数组。 数据通道 ID 的顺序应与(7) 表格数据/值相同。 若无法指定数据通道 ID 的顺序,则应添加引用编号。 此外, 如果数据通道 ID 的顺序可	可选	*

		固定且永远不变，则可省略这些元素。		
DataSet	(7)DataSet_Tabular	见本章第 2.4.7.1	可选	*

(6) 事件数据结构(EventData)

事件数据结构

表 2.4.7.4(6)

名称	数据类型	备注	强制/可选	最大数
NumberOfDataSet	NonNegativeInteger	事件数据元素中的数据集数	可选	1
DataSet	(8)DataSet_Event	见本章第 2.4.7.1	可选	*

(7) 数据集_表格结构(DataSet_Tabular)

数据集_表格结构

表 2.4.7.4(7)

名称	数据类型	备注	强制/可选	最大数
Timestamp	DateTime	测量时间	强制	1
Value	String	每个数据通道在时间戳处的测量值数组。 值元素的顺序应与 (5) 表格数据/ 数据通道 ID 相同。 若不能指定值元素的顺序，则应添加引用编号	强制	*
Quality	String	以上值的数据质量	可选	1

(8) 数据集_事件结构(DataSet_Event)

数据集_事件结构

表 2.4.7.4(8)

名称	数据类型	备注	强制/可选	最大数
Timestamp	DateTime	测量时间。	强制	1
DataChannelID	String	第 2 章第 2.3.2 的一组数据通道 ID	强制	1
Value	String	测量值	强制	1
Quality	String	以上值的数据质量	可选	1

第3章 数据交换接口

第1节 一般规定

3.1.1 一般要求

3.1.1.1 本章规定的接口规则目的是指导船东或船公司根据规则编制接口服务，向 CCS 提供数字化检验所需数据。

3.1.1.2 数据传输格式应符合本指南第2章规定的标识与结构标准。

3.1.1.3 船东或船舶管理公司编制的接口服务应有日志监控功能，日志内容应至少涵盖最近30天内所有完整历史记录。

3.1.1.4 当数据传输失败时，接口服务应具有故障排除后再次传输的能力。

3.1.1.5 接口服务所在服务器时钟应和 UTC 同步。

第2节 接口服务

3.2.1 一般要求

3.2.1.1 适用于船舶检验数据交换的接口格式规定。

3.2.1.2 接口服务应满足第4章第2节的测试验证要求。

3.2.2 接口标准

3.2.2.1 接口编程遵循 REST 架构原则。

3.2.2.2 接口数据交换框架结构见图3.2.2.2，具体要求如下。

- (1) 船舶与企业船舶数字平台之间的数据交换规则由企业自行定义。
- (2) 企业船舶数据平台提供船舶检验数据分享或提交 CCS 需按照本指南的规则执行。

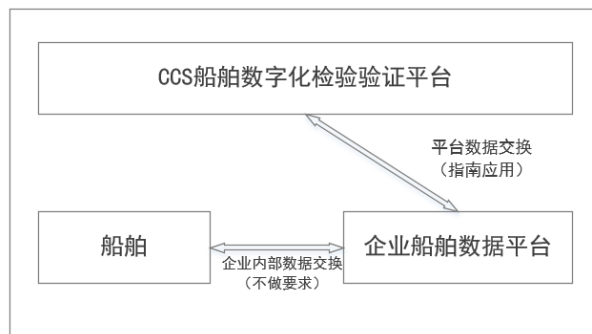


图 3.2.2.2 接口数据交换框架

3.2.2.3 调用过程

(1) 接口支持推送和调取两种模式，如图 3.2.2.3 所示，请求处理使用 Http/Https 协议，如果请求未成功，返回 Http 状态码，如果请求成功，得到对应的调用结果。

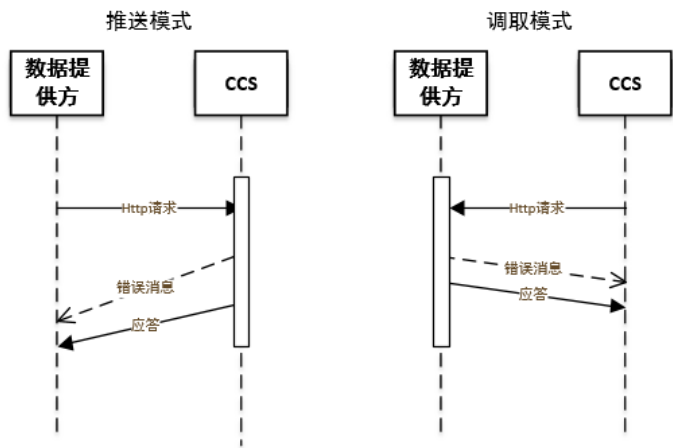


图3. 2. 2. 3 请求-应答调用过程示例

3.2.3 接口格式

3.2.3.1 请求-应答协议结构

- (1) 请求-应答协议结构如下表 3.2.3.1 所示。
- (2) 当使用数据通道类型选项、查询选项和标签选项时，应符合 RFC 3986 的要求，选项和参数不需要区分字母大写和小写。

请求-应答协议的结构 表 3.2.3.1

方法	服务根	资源路径	类型选项	数据通道类型选项	查询选项	标签选项
见 3.2.3.2	见 3.2.3.3	见 3.2.3.4	见 3.2.3.5	见 3.2.3.6	见 3.2.3.7	见 3.2.3.8

3.2.3.2 方法

- (1) 请求-应答协议中使用的方法如表 3.2.3.2 所示。
- (2) 当接口发布方接收到请求时，应在日志中记录 Http 状态代码、Http 方法类型、时间和日期以及结果。

请求-应答协议方法 表 3.2.3.2

方法	描述
GET	GET方法从目标方服务器检索指定的URL资源。 该方法仅适用于在数据提供方服务器获取时序数据通道列表和时序数据时适用。
POST	POST方法将消息主体的数据添加到目标方服务器。 无需指定资源路径。
PUT	PUT方法将消息主体的数据更新至目标方服务器。 无需指定资源路径。

TRACE	TRACE方法获取目标方服务器指定的URL资源的计数。 该方法仅适用于时序数据的资源计数。 计数信息的单元如下： type=ts_data: [Time Series].[Local ID] type=ts_sdd: [Local ID].[Data Channel Type].[History]
-------	---

3.2.3.3 服务根

(1) 服务根格式和详细信息如表 3.2.3.3 所示。

服务根详细信息

表 3.2.3.3

服务根	描述	强制/可选	示例
<Host>	接口服务器IP地址或域名	强制	192.168.1.253
<Port>	Http端口号	可选	8080

3.2.3.4 资源路径

(1) 资源路径仅适用于执行 GET 和 TRACE 方法。

(2) 时序数据传输有如下特殊要求：

① 当数据通道类型为“本地 ID(LocalID)”时，资源路径格式和详细信息如表 3.2.3.4 所示。

② 当数据通道类型为“短 ID(ShortID)”时，指定数据通道列表中给出的一个短 ID。不能通过将短 ID 用“/”或“，”分开，指定多个短 ID。

③ 当数据通道类型为“本地 ID(LocalID)”时，如果使用 GET 方法，可在资源路径中使用通配符。通配符的更多信息见本节 3.2.3.9。

路径资源详细信息

表 3.2.3.4

资源路径	描述	强制/可选	示例
<Ship ID>	船舶标识	可选	imo1234567
<Naming Rule>	见第2章第2.3.2，基于emd需要指定版本号	可选	/emd /v1.0
<Local Data Name>	仅当类型选项为“ts_data”、“ts_sdd”时才可使用。	可选	时序数据： P8.003.006+1/Cylinder.Exhaust_Gas+2/

3.2.3.5 类型选项

(1) 表 3.2.3.5 列出了可处理的数据类型和设置类型的方法。

(2) 类型选项在 Http 头中指定，也可用于 GET 方法以外的其他方法。

(3) 类型选项默认为“ts_data”类型。

类型选项的详细信息

表 3.2.3.5

数据类型	描述	出现次数	示例
ts_data	处理时间序列数据	1	type:ts_data
ts_sdd	处理数据通道列表	1	type:ts_sdd
ge_data	通用数据类型数据	1	type:ge_data

ge_sdd	通用数据类型列表	1	type:ge_sdd
--------	----------	---	-------------

3.2.3.6 数据通道类型

(1) 指定访问船载数据服务器中资源的 ID 类型, 仅适用于时序数据相关通道数据的获取。

(2) 表 3.2.3.6 中列出了可处理的数据通道类型以及指定它们的方法。数据通道类型选项默认为“本地 ID”。

(3) 数据通道类型选项在消息体中指定。

数据通道类型选项详细信息

表 3.2.3.6

数据类型	描述	出现次数	示例
Local ID	使用本地ID访问船载数据服务器上的资源。	1	?idtype=LocalID
Short ID	使用短ID访问船载数据服务器上的资源。	1	?idtype=ShortID

3.2.3.7 查询选项

(1) 查询选项用于指定数据格式、更改排序顺序方式、用于过滤等。

(2) 表 3.2.3.7 列出了可使用的查询选项以及指定它们的方法。

(3) 一次可指定多个查询选项。

(4) 当日期和时间用于选项参数时, 使用 ISO 8601 格式添加。

(5) 查询选项在消息体中指定。

查询选项详细信息

表 3.2.3.7

选项名称	描述	适用	默认	示例
?offset	指定获取数据的UTC时间和日期。	时序数据; 通用数据	服务器接收 “方法”时的 UTC	?offset=2
?before	指定是在基准时间和日期之前或之后进行搜索 (before: true, after: false)。仅当类型选项为“data”时才可使用。	时序数据	TRUE	?before=FALSE
?implemented	指定数据实体采用的格式:XML、JSON、CSV,默认JSON	时序数据; 通用数据	JSON	?implemented=JSON
?limit	当类型为“ts_data”时, 指定在第二时间刻度上获得的时间帧。 当类型为“ts_sdd”时, 指定要获取的数据通道列表修改历史中的项目数。 指定1时返回最新信息。 当类型为“ts_sdd”时, 返回最新的数据通道列表。 当类型为“siod”时, 返回最新的数据源信息。	时序数据	1	?limit=5
?revisionfrom	当获取是数据通道列表的修订历史记录	时序数据		?revisionfrom=

	时，指定日期和时间。 如果类型选项为“ts_sdd”，则可使用 revisionfrom选项。			2016-03-31T00: 00: 00Z
?revisionto	获取是数据通道列表列表的修订历史记录时，指定日期和时间。 可与revisionfrom同时使用。 如果类型选项是“ts_sdd”，则可使用 revisionto选项。	时序数据		?revisionto= 2015-04-01T23: 59: 59Z
?orderby	为获得的数据指定升序或降序的时间顺序。 (asc: 升序; desc: 降序)	时序数据	asc	? orderby= ASC

3.2.3.8 标签选项

- (1) 标签选项仅适用于时序数据接口调用方法。
- (2) 当使用 GET 方法获取时间序列数据时，指定栏目标题输出格式的优先级。
- (3) 如果未指定，按顺序使用短 ID、本地 ID。
- (4) 表 3.2.3.8 列出了可使用的标签选项以及指定它们的方法。
- (5) 标签选项在消息体中指定。

标签选项详细信息

表 3.2.3.8

标签类型	描述	出现次数	示例
短ID	使用具有最高优先级的ShortID。 栏目标头与ShortID、LocalID的优先级一起使用。	1	?label=ShortID
本地ID	使用具有最高优先级的LocalID。	1	?label= LocalID

3.2.3.9 资源路径中通配符的使用

- (1) 该条款仅适用于时序数据接口 GET 方法。
- (2) 有关资源路径的结构，参见第 2 章第 3 节 2.3.2 中的本地 ID。
- (3) 当获取保存在车载数据服务器中的时间序列数据、数据通道列表和数据源信息时，使用资源路径中的通配符创建过滤器。
- (4) 通配符字符串：
 - #：多级通配符。多级通配符用于在 URI 中匹配任意数量级别。
 - +: 单级通配符。单级通配符仅匹配一个 URI 级别。

3.2.3.10 方法的返回代码

- (1) 数据传输处理结果和 Http 状态码（见表 3.2.3.10）。

Http 状态码

表 3.2.3.10

状态代码	消息	描述
200	完成	
201	已创建	返回新生成资源的URL。
400	错误请求	语法错误

401	未认证	用户确认错误
403	禁止	访问未经授权的目录和文件
404	未找到	数据不可用
405	方法不允许（客户端错误）	
408	请求超时	它没有在特定时间内产生
413	请求实体太大	接受意外请求
500	服务器内部错误	执行未经授权的方法

3.2.4 接口示例

3.2.4.1 表格数据示例

数据示例

表 3.2.4.1

时间戳	船舶标识	本地ID	值
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Speed	268
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Speed	269
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Speed	255
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Temperature	80
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Temperature	81
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Temperature	83
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Speed	270
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Speed	271
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Speed	280
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Temperature	82
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Temperature	83
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Temperature	84
2022/01/25 06:47:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Speed	270
2022/01/25 06:47:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Speed	271
2022/01/25 06:47:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Speed	280
2022/01/25 06:47:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Temperature	82
2022/01/25 06:47:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Temperature	83
2022/01/25 06:47:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Temperature	84

3.2.4.2 POST 方法

POST http://localhost/?implemented=JSON

{数据格式}

- (1) 数据格式见附录 5。
- (2) 处理结果在 Http 请求中返回，返回值参见表 3.2.3.10 Http 状态码。

3.2.4.3 GET 方法

- (1) 对表 3.2.4.1 中的实际记录数据执行 GET 方法。

GET http://localhost/IMO1234567/+//P4.006.003/+//?offset = 2022-01-25T06: 45:

03Z&before=true&limit=3

(2) 搜索条件:

在 IMO1234567 船舶本地 ID 中包含 “P4.006.003” 2022 年 11 月 25 日, 6:45:03 或更早的日期和时间, 在过去方向搜索 3 秒内的数据。

(3) 搜索结果如下:

时间戳	船舶标识	本地ID	值
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Speed	268
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Speed	269
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Speed	255
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Temperature	80
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Temperature	81
2022/01/25 06:43:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Temperature	83
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Speed	270
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Speed	271
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Speed	280
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+1//DrivingEnd/Temperature	82
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+2//DrivingEnd/Temperature	83
2022/01/25 06:45:00	IMO1234567	/emd/v1.0/P4.006.003+3//DrivingEnd/Temperature	84

3.2.4.4 PUT 方法

PUT http://localhost/

{数据格式}

- (1) 数据格式见附录 5。更新对应于数据设置结构的时间戳和本地 ID 的实际记录数据。
- (2) 处理结果在 Http 请求中返回, 返回值参见表 3.2.3.10 Http 状态码。

3.2.4.5 TRACE 方法

- (1) 对表 3.2.4.1 中的实际记录数据执行 TRACE 方法:

TRACE http://localhost/IMO1234567/+// ExhaustGas/#?offset=2022-01-25T06:43:02Z&before=true

(2) 搜索条件:

在 IMO1234567 船舶本地 ID 中包含 “P4.006.003” 在 2022 年 01 月 25 日, 6:43:02 或更早的日期和时间, 在时间节点之前的方向选择。

(3) 正常结束的返回消息:

{Result:6}

第 3 节 接口与数据安全

3.3.1 一般要求

3.3.1.1 接口服务应采用适当的身份认证方法和接口数据范围的授权功能。

3.3.1.2 接口服务应采用 SSL 协议进行数据传输，提供数据传输的安全支持。

3.3.1.3 接口服务安全同时需满足 CCS《船舶网络系统要求及安全评估指南》第 3.3.3 通信安全的相关要求。

3.3.1.4 推荐船东或船舶管理公司在数据交换时能够应用区块链技术，以达到数据确权，防篡改、防抵赖的目的。

3.3.1.5 通过数据交换提交到 CCS 的数据所有权归数据授权方所有，仅用于 CCS 开展船舶的数字化检验。

第4章 数据交换实施与测试验证

第1节 数据交换实施

4.1.1 一般要求

4.1.1.1 实施船舶数字化检验的船东或船舶管理公司按照 CCS《船舶数字化检验应用指南》第2章要求制定实施计划。

4.1.1.2 检验数据交换过程按本指南的相关要求进行标识，并组织数据结构。

4.1.1.3 在实施数据交换前要与 CCS 明确数据交换接口采用推送或调取模式。

4.1.1.4 在实施数据交换前要与 CCS 明确所交换的数据权属及用途。

4.1.2 实施步骤

4.1.2.1 检验数据交换前，船东或船舶管理公司按照实施计划准备数字化检验所需数据。

4.1.2.2 船东或船舶管理公司（或委托的供方服务机构）提供数据交换接口服务，并开展测试验证。

4.1.2.3 经 CCS 验证同意后，按照实施计划开展数据交换。

第2节 测试验证

4.2.1 一般要求

4.2.1.1 接口发布方应对数据交换接口模块进行测试验证，并将方法和结果记录在测试验证报告中。

4.2.1.2 测试验证报告要体现测试验证项目、方法、判断标准和结果。

4.2.1.3 测试验证适用于船舶申请数字化检验附加标志时。

4.2.1.4 CCS 将参考接口发布方的测试验证报告对接口进行测评和适配，确保其功能状态监控正常，数字化检验所需数据能够完整提交到 CCS。

4.2.2 测试验证项目

4.2.2.1 系统时钟

(1) 接口发布方数据服务器系统内部时钟具有与 UTC 同步的功能。

(2) 无论数据服务器是否与 UTC 同步，都需要确认服务器的系统时钟精度与 UTC 的误差为 1 秒/每小时或更小。

(3) 确认当船载数据服务器与 UTC 不同步时，系统时钟出现异常应具有警报提示的功能。

4.2.2.2 数据格式

通过检查接口文档确认数据交换接口处理的数据格式符合本指南第 2 章第 4 节的要求。

4.2.2.3 数据传输服务

- (1) 通过观察确认接口数据传输服务所请求的数据。
- (2) 通过观察确认数据传输服务中发送的数据并根据方法做了正确的处理。
- (3) 数据传输服务可能采用的所有方法进行测试。

4.2.2.4 日志管理

(1) 通过观察确认接口服务具有系统日志功能，通过分析评估确认所有系统日志记录被保存，且在之后至少 30 天内可供查阅，并能够将以下事项记录到日志中。

- ① 请求来源的访问控制日志，能够记录审核通过或失败日志。
- ② 记录请求来源所请求的接口方法、参数，能够记录调用成功和失败日志。
- ③ 能够记录接口调用耗时。

附录 1 CCS 船舶数字模型

1 一般要求

1.1 CCS 船舶数字模型是对船舶数字化分类进行了层次结构划分，为每个项目给出了编码和名称，编码基于通用十进制分类(UDC)编码系统，有关 CCS 船舶数字模型详细说明、工具及接口，请参见 <http://digitalship.ccs.org.cn/emd/> 栏目。

2 船舶数字模型一级分类说明

标准模型一级分类

表 1

编码	中文名称	英文名称
P1	文件资料	Document
P2	船体结构	Hull Structure
P3	甲板设备	Deck Equipment
P4	主推进系统	Main Propulsion System
P5	操舵系统	Steering System
P6	锅炉系统	Boiler and Associated Equipment
P7	防污染设备	Pollution Prevention System
P8	电力系统	Electric Power System
P9	航行设备系统	Navigation aid Equipment
P10	无线电设备系统	Radio Communications Equipment
P11	救生消防	Lifesaving and Fire Fighting
P12	动力定位系统	Dynamic Positioning System
P13	起重设备	Lifting Appliance
P14	能效管理系统	Energy Efficiency Management System
OTH	其他	Other

注：未列入上表的类别，经 CCS 评估确认后可纳入使用。

附录 2 位置编码

1 位置是指在船上的位置。规定了以下 3 种类型的位置：

1.1 绝对位置：相对于船上固定坐标系的位置。

1.2 通用位置：描述一个典型位置的项，通常出现在普遍船舶上。

1.3 相对位置：个体在一组部件中相对彼此的相对位置。

2 使用 emd 进行传感器命名，通常使用位置编码(表 2)。

位置编码

表 2

位置分类	代码	中文描述	英文描述
绝对位置	Forward	船的前部	Front part of the ship
	Bow	船头	Bow of the ship
	Port	朝向船航向的左侧（左舷）	Left side of the ship towards the ship's heading
	Starboard	朝向船航向的右侧（右舷）	Right side of the ship towards the ship's heading
	ForwardPort	朝向船航向的左前侧	Left front side of the ship towards the ship's heading
	ForwardStarboard	朝向船航向的右前侧	Right front side of the ship towards the ship's heading
	AftPort	朝向船航向的左后侧	Left rear side of the ship towards the ship's heading
	AftStarboard	朝向船航向的右后侧	Right rear side of the ship towards the ship's heading
	Center	船左舷和舷侧之间的中间部分	Center part between ship's port side and starboard side
	Middle	船舶纵向中心	Center of the ship in the longitudinal direction
	MiddlePort	船舶左舷的纵向中心	The port side of center the longitudinal direction of the ship
	MiddleStarboard	船舶右舷的纵向中心	The starboard side of the center of the ship in the longitudinal direction
	Aft	船的后部	Rear part of the ship
	n_Platform	第 n 号平台	Number Platform
	Deck_n	第 n 号甲板	Dock Number
	Fr_n	肋位号 n	Frame Number
	Stern	船尾	Stern of the ship
通用位置	PhaseR	三相交流系统中的第一相、R 相、U 相	First phase, R phase, U phase in a three-phase AC system

	PhaseS	三相交流系统中的第二相 S 相、V 相	Second phase S phase, V phase in a three-phase AC system
	PhaseT	三相交流系统中的第三相、T 相、W 相	Third phase, T phase, W phase in a three-phase AC system
	Pipe	管道	Pipeline
相对位置	Upper	设备顶部	Top part of the equipment
	Lower	设备底部	Bottom part of the equipment
	Inlet	进口	Inlet
	Outlet	出口	Outlet
	Open	开口位置	Open position
	Close	闭合位置	Closed position
	DrivingEnd	驱动端	Near side of prime mover/motor
	FreeEnd	自由端	Far side of prime mover/motor

注：未列入上表的类别，经 CCS 评估确认后可纳入使用。

附录 3 通用传感器信号类型

传感器信号类型常用的信号类型参见表 3。

传感器信号类型

表 3

序号	信号代码	中文名称	英文名称
1	Pressure	压强	Pressure
2	Temperature	温度	Temperature
3	Speed	速度	Speed
4	Flow	流量	Flow
5	Amplitude	振幅	Amplitude
6	Power	功率	Power
7	Current	电流	Current
8	Frequency	频率	Frequency
9	Angle	角度	Angle
10	Voltage	电压	Voltage
11	Latitude	纬度	Latitude
12	Longitude	经度	Longitude
13	Failure	故障	Failure
14	Alarm	报警	Alarm
15	Abnormal	异常	Abnormal
16	Length	长度	Length
17	Direction	方向	Direction
18	Aperture	开度	Aperture
19	Concentration	浓度	Concentration
20	Level	液位	Level
21	Distance	距离	Distance
22	Torque	转矩	Torque
23	Resistance	电阻	Resistance
24	On-Off	开关状态	On-Off Status
25	Hight	高度	Hight
26	Stress	应力	Stress
27	Strain	应变	Strain

附录 4 通用数据元数据结构示例

1 文件传输对象元数据结构示例

1.1 XML 格式:

```
<Documents>
  <Row>
    <DocumentKey>文件唯一标识</DocumentKey>
    <DocumentName>文件名称</DocumentName>
    <DocumentSize>文件大小</DocumentSize>
    <ValidDate>有效期</ValidDate>
    <DocumentBody>文件内容 Base64 编码</DocumentBody>
    <Remark>备注</Remark>
  </Row>
</Documents>
```

1.2 JSON 格式:

```
{
  "Documents": [
    {
      "DocumentKey": "文件唯一标识",
      "DocumentName": "文件名称",
      "DocumentSize": "文件大小",
      "ValidDate": "有效期",
      "DocumentBody": "文件内容 base64 编码",
      "Remark": "备注"
    }
  ]
}
```

2 船舶营运实验与维修结构示例

2.1 XML 格式

```
<Definition>
  <DataObjectCode>数据对象代码</DataObjectCode>
  <DataObject>数据对象</DataObject>
  <OperationProjects>
    <Row>
      <OperationProjectCode>操作项目代码</OperationProjectCode>
      <OperationProject>操作项目</OperationProject>
      <EquipmentNo>设备编号</EquipmentNo>
      <OperationProjectData>
        <Row>
          <OperationType>操作类型</OperationType>
          <OperationResult>操作结果</OperationResult>
          <No>编号</No>
        </Row>
      </OperationProjectData>
    </Row>
  </OperationProjects>
</Definition>
```

```
<Temperature>温度</Temperature>
<Speed>转速</Speed>
<OperationTime>操作时间</OperationTime>
<Remarks>备注</Remarks>
<Enclosures>
  <Row>
    <AttachmentID></AttachmentID>
    <AttachmentName ></AttachmentName>
    <AttachmentAddress ></AttachmentAddress>
  </Row>
</Enclosures>
</Row>
</OperationProjectData>
</Row>
</OperationProjects>
</Definition>
```

2.2 JSON 格式

```
{
  "dataObjectCode": "数据对象代码",
  "dataObject": "数据对象",
  "operationProjects": [{
    "operationProjectCode": "操作项目代码",
    "operationProject": "操作项目",
    "equipmentNo": "设备编号",
    "operationProjectData": [{
      "operationType": "操作类型",
      "operationResult": "操作结果",
      "no": "编号",
      "temperature": "温度",
      "speed": "转速",
      "operationTime": "操作时间",
      "remarks": "备注",
      "enclosures": [{
        "attachmentID": "",
        "attachmentName": "",
        "attachmentAddress": ""
      }]
    }]
  }]
}
```

附录 5 时序数据结构要求与实现

1 一般规定

1.1 交换数据

1.1.1 本指南第2章第4节所述结构的数据将由XML、JSON或CSV实现。

1.1.2 无论采用何种实现语言，数据都应遵守以下要求。

1.1.3 数据应以文本形式书写，并采用 UTF-8 编码，无字节顺序标记(BOM)。

1.1.4 建议使用XML/JSON交换时间序列数据的数据通道列表和时序数据（包含单个数据集或多个数据集）。在使用JSON时，由于没有模式验证机制，建议使用能够与实际船上数据服务器通信的环境开发应用程序。对于特别大的时间序列数据，建议使用CSV，在这种情况下，需要提前共享数据通道列表。建议使用XML/JSON交换数据通道列表。

2 XML 的实现

2.1 一般要求

2.1.1 在本节中，对使用XML和XML模式实现本指南第2章第4节中规定的数据格式的情况进行了说明。

2.1.2 XML模式是根据下面描述的规则制定的。

2.1.2.1 应规定最小和最大发生次数。

2.1.2.2 应创建XML模式以命令XML向所有元素和属性添加命名空间。

示例：<nr:NamingRule nr:ID="Naming_Rule"/>

2.1.2.3 命名规则特定的元素和属性应属于其自身的命名空间。

2.1.2.4 文件中的任何换行或缩进字符都不应视为重要信息。

2.2 标准 & XML 模式数据类型。

本指南第2章第2.4.3中的标准数据类型替换为表2.2中的XML模式数据类型。

标准数据类型和 XML 模式数据类型之间的对应关系

表 2.2

标准数据类型	XML 模式数据类型	备注
Integer	Integer	整型
PositiveInteger	PositiveInteger	1 或更大的整型
NotNegativeInteger	NotNegativeInteger	0 或更大的整型
Real	Decimal	小数
DateTime	DateTime	ISO 8601 的时间和日期
CharacterString	String	随机字符串
Boolean	Boolean	真值
Null	XML 架构 <element name="example" type="float"	在指定空值时，在 XML 模式的元素定义中指定属性 nillable="true"，在

	nillable="true"/> XML 文件 <example nil="true"/>	XML 文档的相应元素中指定属性 nillable="true"
--	--	-------------------------------------

2.3 数据通道列表

2.3.1 命名空间

2.3.1.1 以下命名空间应添加到数据通道列表中的所有元素和属性中。

XML 命名: `sdd=urn:ISO 19848:船舶_数据_定义`

2.3.1.2 在使用命名规则特定的元素/属性时, 命名空间应避免重复。

示例:

XMLNamespace: `nr = urn:ISO19848:Ship_Data_Definition:CCS_EMODEL`

2.3.2 XML 架构语法

本指南第2章第2.4.6中的数据结构应使用以下XML架构进行定义和验证。

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  xmlns:sdd="urn:ISO19848:SHIP_DATA_DEFINITION:CCS_EMODEL"
  attributeFormDefault="unqualified" elementFormDefault="qualified"
  targetNamespace="urn:ISO19848:SHIP_DATA_DEFINITION:CCS_EMODEL"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!--2.4.6.3(1) 包结构-->
  <xs:element name="Package" type="sdd:Package"/>
  <xs:complexType name="Package">
    <xs:sequence>
      <!--2.4.6.3(2) 包头结构-->
      <xs:element name="Header" type="sdd:Header" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
  <!--2.4.6.3(2) 包头结构-->
  <xs:complexType name="Header">
    <xs:sequence>
      <!--IMO编号, 船舶统一识别号'或者其他标识编码-->
      <xs:element name="ShipID" type="xs:string" minOccurs="1" maxOccurs="1"/>
      <!--数据所有者-->
      <xs:element name="Author" type="xs:string" minOccurs="0" maxOccurs="1"/>
      <!--数据创建时间-->
      <xs:element name="DateCreated" type="xs:dateTime" minOccurs="0" maxOccurs="1"/>
      <!--自定义包头的扩展定义-->
      <xs:any processContents="lax" namespace="##other" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--2.4.6.3 (3) 数据通道列表结构-->
  <xs:complexType name="DataChannelList">
    <xs:sequence>
      <!--2.4.6.3 (4) 数据通道结构-->
      <xs:element name="DataChannel" type="sdd:DataChannel" minOccurs="1"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
```

```

<!--2.4.6.3 (4) 数据通道结构-->
<xs:complexType name="DataChannel">
  <xs:sequence>
    <!--2.4.6.3 (5) 数据通道ID结构-->
    <xs:element name="DataChannelID" type="sdd:DataChannelID" minOccurs="1"
maxOccurs="1"/>
    <!--2.4.6.3 (6) 属性结构-->
    <xs:element name="Property" type="sdd:Property" minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
</xs:complexType>
<!--2.4.6.3 (5) 数据通道ID结构 -->
<xs:complexType name="DataChannelID">
  <xs:sequence>
    <xs:element name="LocalID" type="xs:string" minOccurs="1" maxOccurs="1"/>
    <xs:element name="ShortID" type="xs:string" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
</xs:complexType>
<!--2.4.6.3 (6) 属性结构-->
<xs:complexType name="Property">
  <xs:sequence>
    <!--2.3.3.2 (1) 数据通道类型的标识，例如平均值、报警和状态等-->
    <xs:element name="DataChannelType" type="sdd:DataChannelType" minOccurs="1"
maxOccurs="1"/>
    <!--2.3.3.2 (2) 用于描述数据格式的格式定义-->
    <!--假设数据类型是 "浮点","整数","布尔","文本" 和"符号"-->
    <xs:element name="Format" type="sdd:Format" minOccurs="1" maxOccurs="1"/>
    <!--2.3.3.2 (3) 测量值范围-->
    <xs:element name="Range" type="sdd:Range" minOccurs="0" maxOccurs="1"/>
    <!--2.3.3.2 (4) 测量值的单位和数量-->
    <xs:element name="Unit" type="sdd:Unit" minOccurs="0" maxOccurs="1" />
    <!--2.3.3.2 (5) 测量值的数据质量评估方案名称-->
    <xs:element name="QualityCoding" type="xs:string" minOccurs="0" maxOccurs="1"/>
    <!--2.3.3.2 (6) 控制系统和其他仪表中指定的名称-->
    <xs:element name="Name" type="xs:string" minOccurs="0" maxOccurs="1"/>
    <!--2.3.3.2 (7) 备注-->
    <xs:element name="Remarks" type="xs:string" minOccurs="0" maxOccurs="1"/>
    <!--自定义扩展属性-->
    <xs:any processContents="lax" namespace="##other" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<!--2.4.6.3 (7) 数据通道类型结构-->
<!--2.3.3.2 (1) 数据通道类型的标识，例如平均值、报警和状态等-->
<xs:complexType name="DataChannelType">
  <xs:sequence>
    <xs:element name="Type" minOccurs="1" maxOccurs="1">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Inst"/>
          <xs:enumeration value="Average"/>
          <xs:enumeration value="Max"/>
          <xs:enumeration value="Min"/>
          <xs:enumeration value="StandardDeviation"/>
          <xs:enumeration value="Calculated"/>
          <xs:enumeration value="SetPoint"/>
          <xs:enumeration value="Output"/>
          <xs:enumeration value="Alert"/>
          <xs:enumeration value="Status"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

```

        <xs:enumeration value="ManuallyInput"/>
    </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="UpdateCycle" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
<xs:element name="CalculationPeriod" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
</xs:sequence>
</xs:complexType>
<!--2.4.6.3 (8) 格式结构-->
<!--2.3.3.2 (2) 测量值格式-->
<xs:complexType name="Format">
    <xs:sequence>
        <!--格式类型-->
        <xs:element name="Type" minOccurs="1" maxOccurs="1">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="Decimal"/>
                    <xs:enumeration value="Integer"/>
                    <xs:enumeration value="Boolean"/>
                    <xs:enumeration value="String"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:element>
        <!--2.4.6.3 (9) 取值限制-->
        <xs:element name="Restriction" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:choice maxOccurs="unbounded">
                    <xs:element name="Enumeration" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
                    <xs:element name="FractionDigits" type="xs:integer" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="Length" type="xs:integer" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="MaxExclusive" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="MaxInclusive" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="MaxLength" type="xs:integer" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="MinExclusive" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="MinInclusive" type="xs:decimal" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="MinLength" type="xs:integer" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="Pattern" type="xs:string" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="TotalDigits" type="xs:integer" minOccurs="0" maxOccurs="1"/>
                    <xs:element name="WhiteSpace" minOccurs="0" maxOccurs="1">
                        <xs:simpleType>
                            <xs:restriction base="xs:string">
                                <xs:enumeration value="preserve"/>
                                <xs:enumeration value="replace"/>
                                <xs:enumeration value="collapse"/>
                            </xs:restriction>
                        </xs:simpleType>
                    </xs:element>
                </xs:choice>
            </xs:complexType>
        </xs:element>
    </xs:sequence>
</xs:complexType>
<!--2.4.6.3 (10)范围结构-->
<!--2.3.3.2 (3) 测量值范围-->
<xs:complexType name="Range">
    <xs:sequence>
        <!--测量值上限-->
        <xs:element name="High" type="sdd:emptyOrDecimal" minOccurs="1" maxOccurs="1"/>

```

```

        <!--测量值下限-->
        <xs:element name="Low" type="sdd:emptyOrDecimal" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
</xs:complexType>
<!--2.4.6.3 (11) 单位结构-->
<!--2.3.3.2 (4) 测量值的单位和计量-->
<xs:complexType name="Unit">
    <xs:sequence>
        <!--单位符号-->
        <xs:element name="UnitSymbol" type="xs:string" minOccurs="1" maxOccurs="1"/>
        <!--ISO80000中定义的测量值计量名称 -->
        <xs:element name="QuantityName" type="xs:string" minOccurs="0" maxOccurs="1"/>
        <!--自定义单位元素的扩展点，例如“数量符号”、“比例因子”等-->
        <xs:any processContents="lax" namespace="##other" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<!--空值、可接受小数-->
<xs:simpleType name="emptyOrDecimal">
    <xs:union memberTypes="sdd:empty xs:decimal"/>
</xs:simpleType>
<xs:simpleType name="empty">
    <xs:restriction base="xs:string">
        <xs:enumeration value=""/>
    </xs:restriction>
</xs:simpleType>
</xs:schema>

```

2.3.3 XML 表示

此示例使用namespace:nr作为自定义元素。自定义元素以斜体书写以供参考。

```

<?xml version="1.0" encoding="utf-8"?>
<Package
  xmlns="urn:ISO19848:SHIP_DATA_DEFINITION"
  xmlns:nr="urn:ISO19848:SHIP_DATA_DEFINITION:CCS_EMODEL">
  <Header>
    <ShipID>IMO1234567</ShipID>
    <Author>Author1</Author>
    <DateCreated>2022-12-01T00:00:00+00:00</DateCreated>
    <nr:CustomHeaderElement>Vender specific headers</nr:CustomHeaderElement>
  </Header>
  <DataChannelList>
    <DataChannel>
      <DataChannelID>
        <LocalID>/emd/v1.0/P4.006.003 +1//DrivingEnd/Temperature</LocalID>
        <ShortID>0010</ShortID>
      </DataChannelID>
      <Property>
        <DataChannelType>
          <Type>Inst</Type>
          <UpdateCycle>1</UpdateCycle>
        </DataChannelType>
        <Format>
          <Type>Decimal</Type>
          <Restriction>
            <FractionDigits>1</FractionDigits>
            <MaxInclusive>200.0</MaxInclusive>

```

```

        <MinInclusive>-150.0</MinInclusive>
    </Restriction>
</Format>
<Range>
    <High>150.0</High>
    <Low>0.0</Low>
</Range>
<Unit>
    <UnitSymbol>°C</UnitSymbol>
    <QuantityName>温度</QuantityName>
</Unit>
<QualityCoding>OPC_QUALITY</QualityCoding>
<Name>1号主机热水温度 </Name>
<Remarks> 位置: ECR, 制造厂: AAA Company, 型号: TYPE-AAA </Remarks>
<nr:CustomPropertyElement>Vender specific Property</nr:CustomPropertyElement>
</Property>
</DataChannel>
<DataChannel>
    <DataChannelID>
        <LocalID>/emd/v1.0/P4.006.003+1//DrivingEnd/SPEED</LocalID>
        <ShortID>0020</ShortID>
    </DataChannelID>
    <Property>
        <DataChannelType>
            <Type>Average</Type>
            <UpdateCycle>60</UpdateCycle>
            <CalculationPeriod>3600</CalculationPeriod>
        </DataChannelType>
        <Format>
            <Type>Integer</Type>
            <Restriction>
                <FractionDigits>1</FractionDigits>
                <MaxInclusive></MaxInclusive>
                <MinInclusive></MinInclusive>
            </Restriction>
        </Format>
        <Range>
            <High>7000</High>
            <Low>0</Low>
        </Range>
        <Unit>
            <UnitSymbol>RPM</UnitSymbol>
            <QuantityName>转速</QuantityName>
        </Unit>
        <QualityCoding>OPC_QUALITY</QualityCoding>
        <Name>1号主机增压器转速 </Name>
        <Remarks> 位置: ECR, 制造厂: AAA Company, 型号: TYPE-AAA </Remarks>
        <nr:CustomPropertyElement>Vender specific Property</nr:CustomPropertyElement>
    </Property>
</DataChannel>
</DataChannelList>

```

2.4 时间序列数据

2.4.1 命名空间

在时间序列数据中，应添加以下命名空间。

XMLNamespace:sdt=urn:ISO19848:Ship_Data_Transport:CCS_EMODEL

2.4.2 XML 架构语法

本指南第 2 章第 2.4.5 中的数据结构应使用以下 XML 模式进行定义和验证。

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  xmlns:sdt="urn:ISO19848:SHIP_DATA_TRANSPORT:CCS_EMODEL "
  attributeFormDefault="unqualified" elementFormDefault="qualified"
  targetNamespace="urn:ISO19848:SHIP_DATA_TRANSPORT:CCS_EMODEL"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!--2.4.7.4 (1) 包结构-->
  <xs:element name="Package" type="sdt:Package"/>
  <!--2.4.7.4 (1) 包结构-->
  <xs:complexType name="Package">
    <xs:sequence>
      <!--2.4.7.4 (2) 包头结构-->
      <xs:element name="Header" type="sdt:Header"
minOccurs="0" maxOccurs="1"/>
      <!--2.4.7.4 (4) 时间系列数据结构 -->
      <xs:element name="TimeSeriesData" type="sdt:TimeSeriesData"
minOccurs="1" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--2.4.7.4 (2) 包头结构-->
  <xs:complexType name="Header">
    <xs:sequence>
      <!--IMO 编号, 船舶统一识别号或其他船舶唯一编号-->
      <xs:element name="ShipID" type="xs:string"
minOccurs="1" maxOccurs="1"/>
      <!--2.4.7.4 (3) 时间跨度结构-->
      <xs:element name="TimeSpan" type="sdt:TimeSpan"
minOccurs="0" maxOccurs="1"/>
      <!--数据创建时间-->
      <xs:element name="DateCreated" type="xs:dateTime"
minOccurs="0" maxOccurs="1"/>
      <!--数据修改时间-->
      <xs:element name="DateModified" type="xs:dateTime"
minOccurs="0" maxOccurs="1"/>
      <!--数据所有者-->
      <xs:element name="Author" type="xs:string"
minOccurs="0" maxOccurs="1"/>
      <!--自定义包头的扩展定义-->
      <xs:any processContents="lax" namespace="##other" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <!--2.4.7.4 (3) 时间跨度结构-->
  <xs:complexType name="TimeSpan">
    <xs:sequence>
```

```

        <!--数据集的起始时间戳-->
        <xs:element name="Start" type="xs:dateTime" minOccurs="1" maxOccurs="1"/>
        <!--数据集的结束时间戳-->
        <xs:element name="End" type="xs:dateTime" minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
</xs:complexType>
<!--数据通道ID结构-->
<!--值必须是本地ID或短ID字符串-->
<xs:complexType name="DataChannelID">
    <xs:simpleContent>
        <xs:extension base="xs:string">
            <!--Ordinal number of the DataChannelID array-->
            <xs:attribute name="id" type="xs:positiveInteger" use="required" />
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<!--2.4.7.4 (4) 时间系列数据结构 -->
<xs:complexType name="TimeSeriesData">
    <xs:sequence>
        <!--2.4.7.4 (5) 表格数据结构 -->
        <xs:element name="TabularData" type="sdt:TabularData"
minOccurs="0" maxOccurs="unbounded" />
        <!--2.4.7.4 (6) 事件数据结构 -->
        <xs:element name="EventData" type="sdt:EventData"
minOccurs="0" maxOccurs="1" />
        <!--自定义数据类型的扩展定义-->
        <xs:any processContents="lax" namespace="##other" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<!--2.4.7.4 (5) 表格数据结构 -->
<xs:complexType name="TabularData">
    <xs:sequence>
        <!--表格数据结构下的数据结合元素数量 -->
        <xs:element name="NumberOfDataSet" type="xs:nonNegativeInteger"
minOccurs="0" maxOccurs="1" />
        <!--数据通道数量-->
        <xs:element name="NumberOfDataChannel" type="xs:nonNegativeInteger"
minOccurs="0" maxOccurs="1"/>
        <!--2.4.7.4 (5) 目标数据通道标识符-->
        <xs:element name="DataChannelID" type="sdt:DataChannelID"
minOccurs="0" maxOccurs="unbounded"/>
        <!--2.4.7.4 (7) 数据集结构 -->
        <xs:element name="DataSet" type="sdt:DataSet_Tabular"
minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
</xs:complexType>
<!--2.4.7.4 (6) 事件数据结构 -->
<xs:complexType name="EventData">

```

```

<xs:sequence>
  <!--事件数据结构数据集元素数量 -->
  <xs:element name="NumberOfDataSet" type="xs:nonNegativeInteger"
minOccurs="0" maxOccurs="1" />
  <!--2.4.7.4 (6) 数据集结构 -->
  <xs:element name="DataSet" type="sdt:DataSet_Event"
minOccurs="0" maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
<!--2.4.7.4 (7) 表格数据数据集结构 -->
<xs:complexType name="DataSet_Tabular">
  <xs:sequence>
    <!--测量值-->
    <!--测量值可以为空-->
    <xs:element name="Value" type="sdt:Value"
minOccurs="1" maxOccurs="unbounded"/>
  </xs:sequence>
  <!--应使用ISO 8601 日期和时间结构-->
  <xs:attribute name="timeStamp" type="xs:dateTime" use="required"/>
</xs:complexType>
<!--2.4.7.4 (8) 事件数据数据集结构-->
<xs:complexType name="DataSet_Event">
  <xs:sequence>
    <xs:element name="DataChannelID" type="xs:string"
minOccurs="1" maxOccurs="1"/>
    <!--测量值 -->
    <!--测量值可以为空-->
    <xs:element name="Value" type="xs:string"
minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
  <!--应使用ISO 8601 日期和时间结构-->
  <xs:attribute name="timeStamp" type="xs:dateTime" use="required"/>
  <xs:attribute name="quality" type="xs:string" use="optional"/>
</xs:complexType>
<!--表格数据值结构 -->
<xs:complexType name="Value">
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <!--值的数组结构要求。 数组的顺序应和表格数据/数据通道ID数组相同-->
      <xs:attribute name="ref" type="xs:positiveInteger" use="required" />
      <xs:attribute name="quality" type="xs:string" use="optional"/>
    </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:schema>

```

2.4.3 XML表示

XML表示如下：

```

<?xml version="1.0" encoding="utf-8" ?>
<Package xmlns="urn:ISO19848:SHIP_DATA_TRANSPORT:CCS_EMODEL">
  <Header>
    <ShipID>IMO1234567</ShipID>
    <TimeSpan>
      <Start>2022-01-01T12:00:00Z</Start>
      <End>2022-01-03T12:00:00Z</End>
    </TimeSpan>
    <DateCreated>2022-01-03T12:00:00Z</DateCreated>
    <DateModified>2022-01-03T12:00:00Z</DateModified>
    <Author>Author1</Author>
  </Header>
  <TimeSeriesData>
    <TabularData>
      <NumberOfDataSet>2</NumberOfDataSet>
      <NumberOfDataChannel>2</NumberOfDataChannel>
      <!-- ShortID of the DataChannel -->
      <DataChannelID id="1">0010</DataChannelID>
      <DataChannelID id="2">0020</DataChannelID>
      <DataSet timeStamp="2022-01-01T12:00:00Z">
        <Value ref="1" quality="0">100.0</Value>
        <Value ref="2" quality="0">200.0</Value>
      </DataSet>
      <DataSet timeStamp="2022-01-02T12:00:00Z">
        <Value ref="1" quality="0">100.5</Value>
        <Value ref="2" quality="0">205.0</Value>
      </DataSet>
    </TabularData>
    <TabularData>
      <NumberOfDataSet>3</NumberOfDataSet>
      <NumberOfDataChannel>1</NumberOfDataChannel>
      <DataChannelID id="1">0010</DataChannelID>
      <DataSet timeStamp="2022-01-01T12:00:00Z">
        <Value ref="1" quality="0">100.0</Value>
      </DataSet>
      <DataSet timeStamp="2022-01-02T00:00:00Z">
        <Value ref="1" quality="0">100.2</Value>
      </DataSet>
      <DataSet timeStamp="2022-01-02T12:00:00Z">
        <Value ref="1" quality="0">100.5</Value>
      </DataSet>
    </TabularData>
    <EventData>
      <NumberOfDataSet>3</NumberOfDataSet>
      <DataSet timeStamp="2022-01-01T12:00:01Z" quality="0">
        <DataChannelID>0011</DataChannelID>
        <Value>HIGH</Value>
      </DataSet>
      <DataSet timeStamp="2022-01-01T12:00:01Z" quality="0">
        <DataChannelID>0021</DataChannelID>
        <Value>HIGH</Value>
      </DataSet>
      <DataSet timeStamp="2022-01-01T12:00:23Z" quality="0">
        <DataChannelID>0011</DataChannelID>
        <Value>NORMAL</Value>
      </DataSet>
    </EventData>
  </TimeSeriesData>

```

```

<TimeSeriesData>
  <TabularData>
    <NumberOfDataSet>1</NumberOfDataSet>
    <NumberOfDataChannel>2</NumberOfDataChannel>
    <DataChannelID id="1">0010</DataChannelID>
    <DataChannelID id="2">0020</DataChannelID>
    <DataSet timeStamp="2022-01-03T12:00:00Z">
      <Value ref="1" quality="0">101.0</Value>
      <Value ref="2" quality="0">210.0</Value>
    </DataSet>
  </TabularData>
  <TabularData>
    <NumberOfDataSet>2</NumberOfDataSet>
    <NumberOfDataChannel>1</NumberOfDataChannel>
    <DataChannelID id="1">0010</DataChannelID>
    <DataSet timeStamp="2022-01-03T00:00:00Z">
      <Value ref="1" quality="0">100.8</Value>
    </DataSet>
    <DataSet timeStamp="2022-01-03T12:00:00Z">
      <Value ref="1" quality="0">101.0</Value>
    </DataSet>
  </TabularData>
</TimeSeriesData>
</Package>

```

3 JSON 实现

3.1 一般要求

3.1.1 该章节规定了使用JSON描述数据的方法。

3.1.2 JSON实现不是强制性的，但应按照以下方式实现。

3.1.2.1 文件中的任何换行或缩进字符都不应视为重要信息。

3.1.2.2 具有命名空间的XML元素应转换为JSON元素。例如：

<xml:Element>Value</xml:Element> 转换为 { "xml:Element" : "Value" }。

3.2 数据通道列表

JSON表示示例如下：

```

{
  "Package": {
    "Header": {
      "ShipID": "IMO1234567",
      "Author": "Author1",
      "DateCreated": "2015-12-01T00:00:00Z",
      "nr:CustomHeaderElement": "Vender specific headers"
    },
    "DataChannelList": {
      "DataChannel": [{
        "DataChannelID": {
          "LocalID": "/emd/v1.0/P4.006.003 +1//DrivingEnd/Temperature",
          "ShortID": "0010",
        },
        "Property": {

```

```

        "DataChannelType": {
            "Type": "Inst",
            "UpdateCycle": "1"
        },
        "Format": {
            "Type": "Decimal",
            "Restriction": {
                "FractionDigits": "1",
                "MaxInclusive": "200.0",
                "MinInclusive": "-150.0"
            }
        },
        "Range": {
            "High": "150.0",
            "Low": "0.0"
        },
        "Unit": {
            "UnitSymbol": "°C",
            "QuantityName": "Temperature"
        },
        "QualityCoding": "OPC_QUALITY",
        "Name": "1 号主机热水温度",
        "Remarks": "位置: ECR, 制造厂: AAA Company, 规格: TYPE-AAA ",
        "nr:CustomPropertyElement": "Vender specific Property"
    }, {
        "DataChannelID": {
            "LocalID": " /emd/v1.0/P4.006.003+1//DrivingEnd/Speed",
            "ShortID": "0020",
        },
        "Property": {
            "DataChannelType": {
                "Type": "Average",
                "UpdateCycle": "60",
                "CalculationPeriod": "3600"
            },
            "Format": {
                "Type": "Integer",
                "Restriction": {
                    "FractionDigits": "0",
                    "MaxInclusive": "",
                    "MinInclusive": ""
                }
            },
            "Range": {
                "High": "7000",
                "Low": "0"
            },
            "Unit": {
                "UnitSymbol": "RPM",
                "QuantityName": "转速"
            },
            "QualityCoding": "OPC_QUALITY",
            "Name": "1 号主机增压器转速",
            "Remarks": "位置: ECR, 制造厂: AAA Company, 规格: TYPE-AAA ",
            "nr:CustomPropertyElement": "Vender specific Property"
        }
    }
}

```

```

    }
  }
}

```

3.3 时间序列数据

JSON表示的示例如下：

```

{
  "Package": {
    "Header": {
      "ShipID": "IMO1234567",
      "TimeSpan": {
        "Start": "2022-01-01T12:00:00Z",
        "End": "2022-01-03T12:00:00Z"
      },
      "DateCreated": "2022-01-03T12:00:00Z",
      "DateModified": "2022-01-03T12:00:00Z",
      "Author": "Shipboard data server",
    },
    "TimeSeriesData": [{
      "TabularData": [{
        "NumberOfDataSet": "2",
        "NumberOfDataChannel": "2",
        "DataChannelID": ["0010", "0020"],
        "DataSet": [{
          "TimeStamp": "2022-01-01T12:00:00Z",
          "Value": ["100.0", "200.0"],
          "Quality": ["0", "0"]
        }, {
          "TimeStamp": "2016-01-02T12:00:00Z",
          "Value": ["100.5", "205.0"],
          "Quality": ["0", "0"]
        }
      ]
    }, {
      "NumberOfDataSet": "3",
      "NumberOfDataChannel": "1",
      "DataChannelID": ["0110"],
      "DataSet": [{
        "TimeStamp": "2022-01-01T12:00:00Z",
        "Value": ["100.0"],
        "Quality": ["0"]
      }, {
        "TimeStamp": "2022-01-02T00:00:00Z",
        "Value": ["100.2"],
        "Quality": ["0"]
      }, {
        "TimeStamp": "2022-01-02T12:00:00Z",
        "Value": ["100.3"],
        "Quality": ["0"]
      }
    ]
  }],
  "EventData": {
    "NumberOfDataSet": "3",
    "DataSet": [{
      "TimeStamp": "2022-01-01T12:00:01Z",
      "DataChannelID": "0011",

```

```

        "Value": "HIGH",
        "Quality": "0"
      }, {
        "TimeStamp": "2022-01-01T12:00:01Z",
        "DataChannelID": "0021",
        "Value": "HIGH",
        "Quality": "0"
      }, {
        "TimeStamp": "2022-01-01T12:00:23Z",
        "DataChannelID": "0011",
        "Value": "NORMAL",
        "Quality": "0"
      }
    ]
  }, {
    "TabularData": [{
      "NumberOfDataSet": "1",
      "NumberOfDataChannel": "2",
      "DataChannelID": ["0010", "0020"],
      "DataSet": [{
        "TimeStamp": "2022-01-03T12:00:00Z ",
        "Value": ["101.0", "210.0"],
        "Quality": ["0", "0"]
      }
    ]
  }, {
    "NumberOfDataSet": "2",
    "NumberOfDataChannel": "1",
    "DataChannelID": ["0110"],
    "DataSet": [{
      "TimeStamp": "2022-01-03T00:00:00Z",
      "Value": ["100.8"],
      "Quality": ["0"]
    }, {
      "TimeStamp": "2022-01-03T12:00:00Z",
      "Value": ["101.0"],
      "Quality": ["0"]
    }
  ]
}
}
}

```

4 CSV 实现

4.1 一般要求

4.1.1 此节规定了使用CSV描述数据的方法。CSV仅适用于时间序列数据，应与XML/JSON中描述的数据通道列表结合使用。

4.1.2 CSV应按照以下要求实现。

- 应根据RFC 4180描述CSV
- 表格数据和事件数据应单独说明
- CSV所有行中的列数应相同

4.1.3 当数据中的数据通道可清楚识别时，可省略标题行（第一行）。

4.2 表格数据

4.2.1 CSV描述表格数据如下：

时间戳	数据通道1	数据通道2	...	数据通道N
时间戳1	监测值1-1	监测值1-2	...	...
...	...	...	...	...
时间戳M	...	...	...	检测值M-N

其中：

- (1) 时间戳M：为监测日期和时间。
- (2) 数据通道N：为以下ID之一，通用ID、本地ID、短ID。
- (3) 监测值M-N：为数据通道N在时间戳M处的监测值。

使用短ID的表格数据的CSV示例如下：

时间戳,	0010, 0020
2022-01-01T12:00:00Z,	100.0, 200.0
2016-01-02T12:00:00Z,	100.5, 205.0
2022-01-03T12:00:00Z,	101.0, 210.0

注：0010、0020为某些数据通道的短ID。

4.3 事件数据

4.3.1 CSV描述事件数据示例如下：

时间戳,	数据通道ID,	监测值
时间戳1,	数据通道ID1,	监测值1-1
...	...	...
时间戳_M,	数据通道ID_N,	监测值M-N

其中：

- (1) 时间戳M：为监测日期和时间。
- (2) 数据通道N：为以下ID之一，通用ID、本地ID、短ID。
- (3) 监测值M-N：为数据通道N在时间戳M处的监测值。

使用短ID的事件数据的CSV示例如下：

时间戳,	数据通道ID, 监测值
2022-01-01T12:00:01Z,	0011, HIGH
2022-01-01T12:00:01Z,	0021, HIGH
2022-01-01T12:01:23Z,	0011, NORMAL

注：0011、0021为某些数据通道的短ID。