

# 基于XML的振动数据架构设计

王富海, 李伟峰

(91388部队, 广东 湛江 524022)

**摘要:** 在机械振动测试任务中, 一组机械设备的振动数据是由岗位人员在某型任务、特定工况、特定位置下采集原始数据, 并进行数据处理后得到的。对于完整的振动数据存储, 我们不仅要存储数据文件, 更要存储数据文件的相关描述信息。XML是W3C组织发布的一种标记语言标准, 其具有易于扩展、结构性强、交互好、语义丰富、基于内容的数据标识、可格式化、易于处理、与平台无关的特点。本文通过对振动数据结构进行分析, 给出了振动数据的数据结构, 设计了振动数据的XML架构, 未来可以将存储在不同地方的振动数据通过XML格式的文件有效组织起来, 使用本文设计的XML架构可以对生成的XML格式的振动数据文件完成数据结构及有效性验证。

**关键词:** 振动数据; XML; XML Schema

**中图分类号:** TP391.1      **文献标识码:** A

## Design of Vibration Data Schema Based on XML

WANG Fuhai, LI Weifeng

(Troops 91388 of PLA, Zhanjiang 524022, China)

**Abstract:** In the task of mechanical vibration testing, a set of vibration data is obtained by workers in a specific task, specific working condition and specific location. For the whole storage of vibration data, it is necessary to store data files and the related descriptive information of data files. XML is a markup language standard released by the W3C organization with various features such as good expansibility, strong structure, good interactivity, rich semantics, content-based data markup. It can be formatted and easy to handle as well as platform-independent. Through the analysis on the vibration data structure, the paper proposes the vibration data structure and designs the XML-based vibration data schema. In the future, vibration data stored in different places can be effectively organized through XML files. The XML schema designed in this paper can be applied to complete the data structure and verify the vibration data files.

**Keywords:** vibration data; XML; XML Schema

### 1 引言(Introduction)

在机械振动测试任务中, 测试设备会采集大量的振动数据。一组机械设备的振动数据是由岗位人员在某型任务、特定工况、特定位置下采集原始数据, 并进行数据处理后得到的。对于数据分析工作, 我们不仅关心数据信息, 同时也关心用来描述数据的相关信息, 所以对于完整的振动数据存储, 我们不仅要存储数据文件, 更要存储数据文件的相关描述信息。任务信息、工况信息、测点信息这些数据描述信息一般存在于测试方案等文件中, 为了数据存储的完整性, 我们需要梳理并设计一个较好的数据结构来解决振动数据的存储问题。

XML(Extensible Markup Language)是可扩展标记语言的缩写, 是W3C组织于1998年2月发布的一种标记语言标准, 其目的是提供一种国际统一的规范, 进行各种信息数据的统一表示与交换<sup>[1,2]</sup>。XML具有易于扩展、结构性强、交互好、

语义丰富、基于内容的数据标识、可格式化、易于处理、与平台无关的特点, 解决了在不同系统之间的数据结构差异, 使得数据层在XML技术的支持下得到统一<sup>[3]</sup>。XML Schema中文含义为XML架构, 它定义了一个XML文档的结构和约束条件, 通过它可以对一个XML文件进行有效性验证, 确保数据结构的正确性<sup>[4,5]</sup>。本文通过分析实际振动测试工作, 设计了一个基于XML的振动数据存储架构。

### 2 振动数据结构分析(Structural analysis of vibration data)

振动测试任务是使用加速度传感器对机械设备的振动情况进行数据采集, 从而分析并掌握机械设备振动特点的活动。一个振动测试任务通常包括几十个工况、上百个测点, 最后产生上千组测试数据。一组振动数据是在特定工况、特定测点下采集得到的。如图1所示, 一组完整的振动数据应该包含任务信息、质控信息、工况信息、测点信息、数据记录

五个部分。

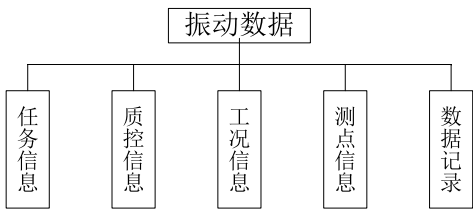


图1 振动数据结构

Fig.1 Structure of vibration data

3 振动数据XML标识(XML markup of vibration data)

根据振动数据的数据结构编制数据标识字典见表1，用来对振动数据要素进行XML标识，表中的中文名称为XML标识的中文意义。

表1 振动数据XML标识

Tab.1 XML markup of vibration data

| 类别    | 中文名称   | XML标识                  |
|-------|--------|------------------------|
| 任务信息  | 任务信息   | TaskInformation        |
|       | 任务编号   | TaskID                 |
|       | 任务名称   | TaskName               |
|       | 任务地点   | TaskArea               |
|       | 任务类型   | TaskType               |
|       | 密级     | SecretClass            |
|       | 任务开始时间 | TaskStartDateTime      |
|       | 任务结束时间 | TaskStopDateTime       |
| 质控信息  | 质控信息   | QCDetails              |
|       | 责任单位   | Agency                 |
|       | 数据处理者  | Operator               |
|       | 数据审核者  | Checker                |
|       | 文件创建时间 | CreateDate             |
|       | 数据处理软件 | DataProcessingSoftWare |
| 测试工况集 | 测试工况名称 | TestConditionName      |
|       | 一级定位信息 | FirstLevelLocation     |
|       | 开启设备名称 | OpenedEquipmentName    |
|       | 开启设备状态 | OpenedEquipmentStatus  |
|       | 备注信息   | Remark                 |
|       | 测点信息   | TestPosition           |
| 测点信息  | 测点编号   | TestPositionID         |
|       | 设备名称   | InstrumentName         |
|       | 测试位置   | TestPositon            |
|       | 测试方向   | TestPositonDirection   |
|       | 测试级别   | TestClass              |
|       | 数据记录   | DataRecord             |
| 数据记录  | 数据集    | DataSet                |
|       | 频率     | Frequency              |
|       | 功率谱    | PowerSpectrum          |

4 振动数据的XML架构设计(XML schema design of vibration data)

4.1 根元素架构设计

在如图2所示，根元素为VibrationTestData，表示振动测试数据。其子元素为任务信息、质控信息、测试工况集、测点信息、数据记录。constraints为相关约束信息，描述了对元素的约束及元素之间相关要素的依赖关系。

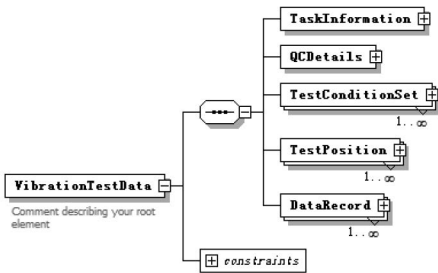


图2 根元素架构设计

Fig.2 Schema design of root element

4.2 任务信息架构设计

任务信息是对测试任务的一些具体描述，如图3所示，包含任务编号、任务名称、任务区域、任务类型、任务及数据密级、任务开始时间、任务结束时间。任务编号具有唯一性，用来区分不同的测试任务。

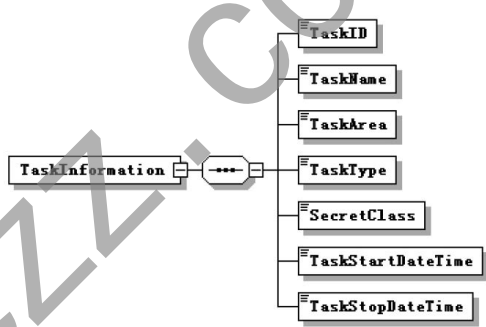


图3 任务信息架构设计

Fig.3 Schema design of task information

4.3 质控信息架构设计

质控信息是数据质量控制方面的信息，如图4所示，包括数据处理机构、数据处理人员、数据文件审核人员、文件创建日期、数据处理软件。

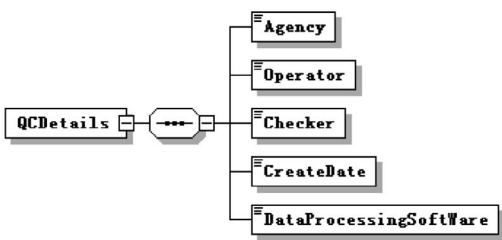


图4 质控信息架构设计

Fig.4 Schema design of quality information

4.4 测试工况集架构设计

测试工况集用来描述振动测试时相关设备的开启情况，如图5所示，包括该设备的位置信息、开启设备的状态、备注信息。测试工况分为单机工况和组合工况。单机工况是指单台设备工作时的技术状态，常用于系统静态测试。组合工况是多台套设备同时工作的技术状态，常用于系统动态测试。由于不同的测试任务可能会有相同的工况名称，但描述的具体工况信息可能是不同的，仅用工况名称不足以区分测试工况信息，所以将任务编号和测试工况名称设为测试工况集的属性，用来唯一标识区分测试工况信息。

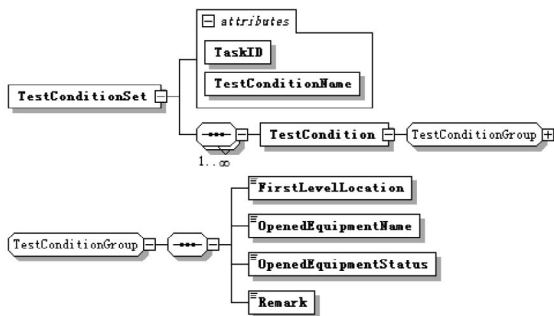


图5 测试工况集架构设计

Fig.5 Schema design of test condition

#### 4.5 测点信息架构设计

测点信息用来描述振动传感器的具体安装位置，如图6所示，包括被测设备名称、测点位置、测试方向、测试级。通常会在一个被测设备上的不同位置安装多个传感器，所以为测试位置设置测点编号属性，用来唯一标识区分测点信息。

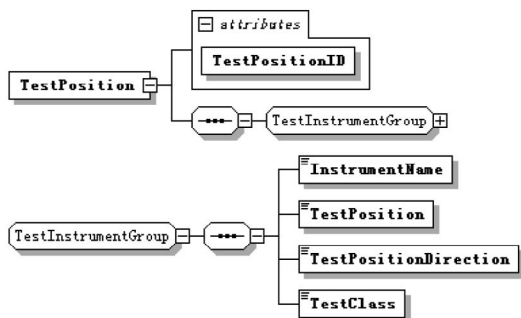


图6 测点信息架构设计

Fig.6 Schema design of test position

#### 4.6 数据集架构设计

数据集振动测试采集的原始数据经过处理后的FFT数据，如图7所示，包括频率和幅值。设置任务编号、测试工况名称、测点编号作为数据记录属性，用来作为数据记录的唯一标识。

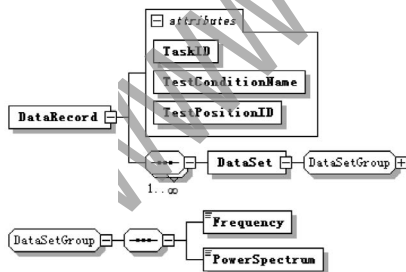


图7 数据集架构设计

Fig.7 Schema design of dataset

#### 4.7 约束信息

在特定测试任务下，任务编号、测试工况名、测点编号具有唯一性。同时测试工况集和数据记录需要引用任务编号，数据记录需要引用测试工况名和测点编号。唯一性和键引用分别描述了数据元素约束和数据元素之间的约束关系，保证了数据在逻辑上的一致性、正确性、有效性和相容性。

表2介绍了本文设计的数据架构的唯一性和键引用部分信

息。完整的约束信息代码如下。

```
<xs:unique name="TaskIDUnique">
  <xs:selector xpath="//TaskInformation/
TaskID"/>
  <xs:field xpath="."/>
</xs:unique>
<xs:keyref name="TaskIDKeyRef"
refer="TaskIDUnique">
  <xs:selector xpath="//TestConditionSet"/>
  <xs:field xpath="@TaskID"/>
</xs:keyref>
<xs:keyref name="TaskIDKeyRef2"
refer="TaskIDUnique">
  <xs:selector xpath="//DataRecord"/>
  <xs:field xpath="@TaskID"/>
</xs:keyref>
<xs:unique name="TestConditionUnique">
  <xs:selector xpath="//TestConditionSet"/>
  <xs:field xpath="@TestConditionName"/>
</xs:unique>
<xs:keyref name="TestConditionKeyRef"
refer="TestConditionUnique">
  <xs:selector xpath="//DataRecord"/>
  <xs:field xpath="@TestConditionName"/>
</xs:keyref>
<xs:unique name="TestPositionIDUnique">
  <xs:selector xpath="//TestPosition"/>
  <xs:field xpath="@TestPositionID"/>
</xs:unique>
<xs:keyref name="TestPositionIDKeyRef"
refer="TestPositionIDUnique">
  <xs:selector xpath="//DataRecord"/>
  <xs:field xpath="@TestPositionID"/>
</xs:keyref>
```

表2 唯一性和键引用

Tab.2 Unique and key reference

| 类型     | 名称                   | 引用                   |
|--------|----------------------|----------------------|
| unique | TaskIDUnique         |                      |
| keyref | TaskIDKeyRef         | TaskIDUnique         |
| keyref | TaskIDKeyRef2        | TaskIDUnique         |
| unique | TestConditionUnique  |                      |
| keyref | TestConditionKeyRef  | TestConditionUnique  |
| unique | TestPositionIDUnique |                      |
| keyref | TestPositionIDKeyRef | TestPositionIDUnique |

#### 5 结论(Conclusion)

XML作为一种标记语言标准，对数据具有自解释能力，拥有良好的可读性。通过设计合适的XML架构，可以容易地验证数据的正确性。本文分析了振动数据的数据结构，  
(下转第34页)