Issues for Standardization of SQL/MM-8: MDR

State Key Laboratory for Software Engineering (SKLSEI), Wuhan University, Wuhan, China, August 21-22, 2009

Dongwon Jeong, Kunsan National University
Jangwon Kim, Korea University
Contents

- Introduction to SQL/MM MDR
- Issue List
- Future Works
❖ Introduction to SQL/MM MDR
❖ Issue List
❖ Future Works
Discussion History

SC 32 Joint Study Periods Meeting (WG2,3,4), Clearwater, FL, February 2007

- First presentation
- Simple Introduction to SQL/MDR (Idea description)

SC 32 Plenary and WG Meetings, New York, USA, July 2007

- Request for a study period on SQL/MM MDR
- Presentation title: SQL/MDR: Query Language for Sharing and Exchanging of Metadata between MDRs - Proposal for a Study Period in WG4 –

WG 4 Meeting, Jagsthausen, Germany, November 2007

- A tough draft has been provided
- Documents: str015 wg4-sqlmm-mdr.doc
Discussion History (cont.)

SC 32 Plenary and WG Meetings, Sydney, Australia, May 2008

- Proposed NWIP
- Resolutions: SC321737a
  SC 32 approves the following subdivisions and SC32 instructs its secretariat to conduct a 60-day letter ballot for Project 1.32.04.01.08.00, and accept the project subdivision if at least four National Bodies support the subdivision.

<table>
<thead>
<tr>
<th>Project Number</th>
<th>Title</th>
<th>Project Editor</th>
<th>Rationale Doc #</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.32.04.01.08.00</td>
<td>SQL Multimedia and Application Packages – Part 8: MDR</td>
<td>Dongwon Jeong</td>
<td>WG4N0070 (rationale 3)</td>
</tr>
</tbody>
</table>

November 15, 2008: Approved

- SC32N1809: Five NBs approved

SC 32/WG 4 Interim Meeting, Portugal, November 2008

In addition, Open Forums and Workshops
Discussion History (cont.)

SC 32 Plenary and WG Meetings, Jeju, Korea, July 2009

• Presentation for the SC 32 Tutorial
• Discussion on WD1 (WG 2 and WG 4 joint meeting)
• Re-definition of the scope: SC 32 N 1911
  - to support retrieval facility
Purpose

- The purpose of this International Standard is to define multimedia and application specific types and their associated routines for retrieval of metadata using the user-defined features in ISO/IEC 9075
- ISO/IEC 13249 SQL/MM is structured as a multi-part standard, and at present, it consists of the following parts:
  - Part 1: Framework
  - Part 2: Full-Text
  - Part 3: Spatial
  - Part 4: Multimedia and Application
  - Part 5: Still Image
  - Part 6: Data Mining
  - Part 7: History
  - Part 8: MDR
Scope

This Part of ISO/IEC 13249

- introduces the metadata registry part of ISO/IEC 13249,
- gives the references necessary for this part of ISO/IEC 13249,
- defines terms, notations, conventions, and definitions specific to this part of ISO/IEC 13249,
- defines concepts specific to this part of ISO/IEC 13249,
- defines metadata registry using user-defined types and associated routines
- covers specifications for retrieving metadata from metadata registries
- does not covers specifications for insertion, deletion, modification of metadata including creation of schema structures for registries
  - Re-defined this scope through the discussion on this issue, Joint meeting of WG 2 and WG 4, SC 32 Plenary and WGs Meetings, Jeju, Korea, July 2009
This part aims to provide a consistent access method for retrieving metadata

user-defined types & routines (based on classes which are defined in ISO/IEC 11179)
Conceptual Model

Retrieve Metadata from Metadata Registries

Scope

- Metadata Definition
- Metadata Update

Metadata Retrieval

User-Defined Types

association

Routines (Methods, Functions)

Registry Storage Model (Tables/Views)
Definition of User-Defined Types and Routines

- **MDR User-Defined Types**
  - An MDR user-defined type is mapped to its corresponding a MDR class
  - Examples of the MDR classes = \{data element, conceptual domain, ...\}

- **Type Notation**
  - The MDR_<ClassName> type is an abstraction for classes of metamodel of ISO/IEC 11179
  
  \[
  \text{MDR_<ClassName> type}
  \]

  - For example, the user-defined type for the class ‘Data_Element’ of the ISO/IEC 11179 metamodel is defined as follows:
    
    \[
    \text{MDR_Data_Element}
    \]
Statement for Creation of a UDT

CREATE TYPE MDR_data_element
AS ( <ALLColumns> )

STATIC METHOD MDR_administered_item_administration_record()
RETURNS TABLE(<ALLColumns>),

STATIC METHOD MDR_data_element()
RETURNS TABLE(<ALLColumns>),

METHOD MDR_data_element_name()
RETURNS <data_element_name>
Definition of User-Defined Types and Routines (cont.)

- Definition of the Static Method `MDR_data_element()`
  - Obtain all data of data element

```sql
CREATE STATIC METHOD MDR_data_element()
RETURNS TABLE(<AllColumns>)
FOR MDR_data_element
BEGIN
  RETURN TABLE(
    SELECT <AllColumns>
    FROM user_de_table);
END
```

*user_de_table* is the table name, which is defined by a user so that it guarantees independence of physical structure.

i.e., **BEGIN-END** part should be implemented by users.
Definition of User-Defined Types and Routines (cont.)

- Output Parameters of the Static Method MDR_data_element()

```java
data_element_administration_record MDR_Administration_Record,
representation_class_qualifier CHARACTER VARYING(30),
data_element_precision Integer
```

- The attribute `data_element_administration_record` is the structured type `MDR_Administration_Record`
Retrieve Metadata

❖ Select Statement

SELECT
    DE.data_element_name, DE.status
FROM
    TABLE (MDR_data_element::MDR_data_element()) AS DE
WHERE
    DE.data_element_administration_record.administrative_status()
    = 'Recorded';

In the SELECT clause, data_element_name and status are defined based on attribute names in metamodel of ISO/IEC 11179
Introduction to SQL/MM MDR

Issue List

Future Works
Issue List for Standardization of SQL/MM MDR

- We need to consider the following issues for standardization of SQL/MM MDR (Metadata Registry):

  - How to define user-defined types and routines
    - Have to prevent loss of metadata when we define user-defined types
    - Done → Referred the specification 20944 (binding API)

  - How to validate user-defined types in Part 8
    - The defined UDTs should correspond to classes of ISO/IEC 11179
    - Each UDT contains all metadata information of its correspondent class of ISO/IEC 11179

  - How to consider relations between classes in ISO/IEC 11179
### Relations between Classes

- **1:N relation → DEC : DE**
  - Ex) p54, Figure 11 — Consolidated metamodel (11179-3_2003)

#### Data_Element_Concept
- data_element_concept_administration_record: Administration_Record
- data_element_concept_object_class: Object_Class
- data_element_concept_property: Property
- object_class_qualifier: String
- property_qualifier: String

#### Value_Domain
- value_domain_administration_record: MDR_Administration_Record
- value_domain_datatype: Datatype
- value_domain_format: String
- value_domain_maximum_character_quantity: Integer
- value_domain_unit_of_measure: Unit_of_Measure

#### Data_Element
- data_element_administration_record: Administration_Record
- representation_class_qualifier: string
- data_element_precision: integer

#### Data_Element
- data_element_administration_record: Administration_Record
- representation_class_qualifier: string
- data_element_precision: integer

---

Study Period Meeting on ROR/ODMS/PSO & SQL/MM-8 MDR
Relations between Classes (cont.)

- Defined UDTs - Has a problem
  - How to query to get results from tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDR_Data_Element_Concept</td>
<td></td>
</tr>
<tr>
<td>data_element_concept_administration_record</td>
<td>MDR_Administration_Record</td>
</tr>
<tr>
<td>data_element_concept_object_class</td>
<td>Object_Class</td>
</tr>
<tr>
<td>data_element_concept_property</td>
<td>Property</td>
</tr>
<tr>
<td>object_class_qualifier</td>
<td>String</td>
</tr>
<tr>
<td>property_qualifier</td>
<td>String</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDR_Value_Domain</td>
<td></td>
</tr>
<tr>
<td>value_domain_administration_record</td>
<td>MDR_Administration_Record</td>
</tr>
<tr>
<td>value_domain_datatype</td>
<td>Datatype</td>
</tr>
<tr>
<td>value_domain_format</td>
<td>String</td>
</tr>
<tr>
<td>value_domain_maximum_character_quantity</td>
<td>Integer</td>
</tr>
<tr>
<td>value_domain_unit_of_measure</td>
<td>Unit_of_Measure</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDR_Data_Element</td>
<td></td>
</tr>
<tr>
<td>data_element_administration_record</td>
<td>MDR_Administration_Record</td>
</tr>
<tr>
<td>representation_class_qualifier</td>
<td>representation_class_qualifier</td>
</tr>
<tr>
<td>data_element_precision</td>
<td>data_element_precision</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDR_Data_Element</td>
<td></td>
</tr>
<tr>
<td>data_element_administration_record</td>
<td>MDR_Administration_Record</td>
</tr>
<tr>
<td>representation_class_qualifier</td>
<td>representation_class_qualifier</td>
</tr>
<tr>
<td>data_element_precision</td>
<td>data_element_precision</td>
</tr>
</tbody>
</table>

Study Period Meeting on ROR/ODMS/PSO & SQL/MM-8 MDR
Relations between Classes (cont.)

_solution

- Put IDs as foreign key into referencing tables

<table>
<thead>
<tr>
<th>MDR_Data_Element_Concept</th>
<th>MDR_Data_Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>data_element_concept_administration_record</td>
<td>Administration_Record</td>
</tr>
<tr>
<td>data_element_concept_object_class</td>
<td>Object_Class</td>
</tr>
<tr>
<td>data_element_concept_property</td>
<td>Property</td>
</tr>
<tr>
<td>object_class_qualifier</td>
<td>String</td>
</tr>
<tr>
<td>property_qualifier</td>
<td>String</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MDR_Value_Domain</th>
<th>MDR_Data_Element</th>
</tr>
</thead>
<tbody>
<tr>
<td>value_domain_administration_record</td>
<td>Administration_Record</td>
</tr>
<tr>
<td>value_domain_datatype</td>
<td>Datatype</td>
</tr>
<tr>
<td>value_domain_format</td>
<td>String</td>
</tr>
<tr>
<td>value_domain_maximum_character_quantity</td>
<td>Integer</td>
</tr>
<tr>
<td>value_domain_unit_of_measure</td>
<td>Unit_of_Measure</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MDR_Data_Element</th>
<th>MDR_Administration_Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>data_element_administration_record</td>
<td>MDR_Administration_Record</td>
</tr>
<tr>
<td>representation_class_qualifier</td>
<td>representation_class_qualifier</td>
</tr>
<tr>
<td>data_element_precision</td>
<td>data_element_precision</td>
</tr>
<tr>
<td>dec_data_identifier</td>
<td>data_identifier (Item.Identifier)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MDR_Data_Element</th>
<th>MDR_Administration_Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>data_element_administration_record</td>
<td>MDR_Administration_Record</td>
</tr>
<tr>
<td>representation_class_qualifier</td>
<td>representation_class_qualifier</td>
</tr>
<tr>
<td>data_element_precision</td>
<td>data_element_precision</td>
</tr>
<tr>
<td>vdl_data_identifier</td>
<td>data_identifier (Item.Identifier)</td>
</tr>
</tbody>
</table>

Spec.
## Relations between Classes (cont.)

<table>
<thead>
<tr>
<th>MDR_Administered_Item</th>
<th>MDR_Administration_Record</th>
</tr>
</thead>
<tbody>
<tr>
<td>administered_item_administration_record</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MDR_Administration_Record</th>
<th>MDR_Item_Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>administered_item_identifier</td>
<td></td>
</tr>
<tr>
<td>administrative_note</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>administrative_status</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>change_description</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>creation_date</td>
<td>Date</td>
</tr>
<tr>
<td>effective_date</td>
<td>Date</td>
</tr>
<tr>
<td>explanatory_comment</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>last_change_date</td>
<td>Date</td>
</tr>
<tr>
<td>origin</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>registration_status</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>unresolved_issue</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>until_date</td>
<td>Date</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MDR_Item_Identifier</th>
<th>Registration_Authority_Identifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>item_registration_authority_identifier</td>
<td></td>
</tr>
<tr>
<td>data_identifier</td>
<td>CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>version</td>
<td>CHARACTER VARYING(30)</td>
</tr>
</tbody>
</table>

**Key**

Ref. 20944
Relations between Classes (cont.)

- **Conceptual_Domain**
  - 1..1 specifying
  - 0..* having
  - `data_element_concept_conceptual_domain_relationship`

- **Data_Element_Concept**
  - `data_element_concept_administration_record [1..1] : Administration_Record`
  - `data_element_concept_object_class [0..1] : Object_Class`
  - `object_class_qualifier [0..1] : String`
  - `data_element_concept_property [0..1] : Property`
  - `property_qualifier [0..1] : String`
  - 0..* related to `data_element_concept_relationship`

- **Data_Element_Concept_Relationship**
  - `data_element_concept_relationship_type_description [1..1] : String`

- **Object_Class**
  - `object_class_administration_record [1..1] : Administration_Record`

- **Concept**
  - 0..* used in
  - 0..* using
  - `concept_relationship`

- **Concept_Relationship**
  - `concept_relationship_type_description [1..1] : String`

- **Property**
  - `property_administration_record [1..1] : Administration_Record`
Relations between Classes (cont.)

```
<table>
<thead>
<tr>
<th>Data_Element_Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>data_element_concept_administration_record [1..1] : Administration_Record</td>
</tr>
<tr>
<td>data_element_concept_object_class [0..1] : Object_Class</td>
</tr>
<tr>
<td>object_class_qualifier [0..1] : String</td>
</tr>
<tr>
<td>data_element_concept_property [0..1] : Property</td>
</tr>
<tr>
<td>property_qualifier [0..1] : String</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>MDR_Data_Element_Concept</th>
</tr>
</thead>
<tbody>
<tr>
<td>data_element_concept_administration_record : MDR_Administration_Record</td>
</tr>
<tr>
<td>oc_data_identifier : data_identifier of Object Class (Item_Identifier)</td>
</tr>
<tr>
<td>data_element_concept_property : MDR_Property</td>
</tr>
<tr>
<td>object_class_qualifier : CHARACTER VARYING(30)</td>
</tr>
<tr>
<td>property_qualifier : CHARACTER VARYING(30)</td>
</tr>
</tbody>
</table>
```
❖ Introduction to SQL/MM MDR
❖ Issue List
❖ Future Works
Future Works

- Define methods and routines of UDTs
  - To retrieve metadata among classes which are defined in MDR
- Verification and Validation of UDTs
  - Whether there is information loss or not
  - Using a scenario or implementation based on real MDR systems
Q/A

Dongwon Jeong
djeong@kunsan.ac.kr; http://ist.kunsan.ac.kr
Information Sciences & Technology Laboratory,
Informatics & Statistics Department,
Kunsan National University
High-level metamodel of ISO/IEC 11179

- High-level overview of the central regions of the metamodel

![Diagram of the metamodel showing relationships between Data_Element_Concept, Conceptual_Domain, Data_Element, and Value_Domain.](image)