Editors:
L. Flon, USC/ISI
S. Malaika, IBM
1-Sep-04

Software Resource Information Model (SRIM)

Version 0.1

Status of This Memo

This document provides information to the community regarding the specification of the Software Resource Information Model (SRIM), based on the DMTF's Common Information Model. Distribution of this document is unlimited.

Copyright Notice

Abstract

This document describes the Relational (SQL) Database Model. It extends the "database" schema in DMTF's Common Information Model (CIM), version 2.9 preliminary [CIM2.9prelim] to include information specific to relational databases. The extensions are based on the ANSI ISO SQL99 specification. This document includes a UML diagram of the classes associated with SQL99, the corresponding managed object format (MOF) for those classes, and an XML representation of the UML.

Contents

1 Introduction

The Common Information Model (CIM) provides consistent information models with well-defined associations that capture management content for applications, systems, networks, devices, and other technology-focused management domains. CIM models establish a common conceptual framework that enables both hardware and software providers to consistently represent management information across vendor boundaries.

The benefits of CIM and the motivation behind extending the CIM information models into additional technology and vendor-agnostic domains are described in [CIMCore]
The same motivations apply for the database management domain. There is significant value to both the customer and the solution provider when database management is unified across the enterprise.

The keywords “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED” and “MAY” used in this document are to be interpreted as described in [RFC2119].

1.1 Overview

The Software Resource Information Model (SRIM) describes the managed objects and their relationships for managing the installation and access to software installed in a grid environment. In particular, the SRIM model at this time addresses the Logical Schema of a relational database.

The CIM 2.9 preliminary schema [CIM2.9prelim]
for database is the foundation for the development of this model. The Logical Schema being introduced in this document is based on SQL1999 [ANSISQL1999] which is upward compatible with SQL2003. Extending the CIM model for databases is con

The Logical Schema work has been split into three stages in order to manage the work more readily as a large number of classes are involved

1. **Stage 1:**
   - Schemas (includes database), Tables (includes columns and triggers), Data Types
2. **Stage 2:**
   - Routines (includes stored procedures, functions) Constraints (includes foreign keys, indexes)
3. **Stage 3:**
   - Other areas such as Access control, Expressions and Statements

1.2 Terminology

This section defines the terminology that is used within this paper and database model. Readers of this document should be familiar with CIM, the existing models, and have a general familiarity with database technology.

Term	Definition
Database	A ***Database*** is a collection of interrelated data, treated as a unit, which is organized into one or more schemas.

Database Environment	A ***database environment*** consists of a database system, one or more databases, and the services that control the administration, usage, monitoring, and maintenance of a database.

2 The Logical Schema Model

This section describes the classes and associations that are defined for the logical schema model, and the relationships between this and the DMTF's database model.

2.1 Background and Assumptions

The GGF's CIM Grid Schema Working Group (CGS-WG) was first formed in 2002 to extend CIM to make it more suitable for discovery and management in a Grid environment. The first effort of the group involved extensions to CIM in the area of job submission, including the addition of classes to represent batch scheduling systems. This work resulted in the GGF information document [JSIM].

This work involves extensions to CIM in the area of database logical schema, for the purpose of discovery and management of databases on the Grid.

2.1.1 Motivation for Developing a Logical Schema Model

There were a number of key factors that contributed to developing a logical schema model

2.1.2 Assumptions

The CIM V2.9 version of the database model focuses on database entities at the level of management at the operating system level, e.g. space allocation on disk drives.

2.2 Conceptual Areas Addressed by the Model

The Logical Schema model defines components for a database environment at a level that permits discovery and management based upon the contents/structure of a database.

2.3 The Database System

The database system represents the software application deployment aspects of the

2.4 Where the Logical Schema Model Lives

The common database describes the vendor and database organization-agnostic

3 Relationships to Other Standards and Specifications

3.1 Overlapping Standards and Specifications

This section provides an overview of other standards and specifications that include some level of support for the management of databases.

3.1.1 SNMP RDBMS MIB specification

The Simple Network Management Protocol (SNMP) RDBMS MIB is the management standard that has the widest adoption rate for relational database implementations. The Internet Engineering Task Force (IETF) first released it in August 1994. The SNMP RDBMS MIB, as specified in RFC 1697, contains information on installed databases, servers, configuration parameters, and a small number of common statistics and events.

3.1.2 CWM Metamodel

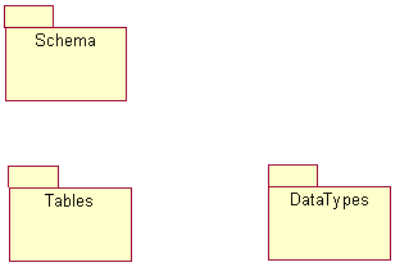
The CWM is a specification that describes the metadata interchange for data warehouses. The OMG released the CWM specification in February 2001. Although the primary focus of this specification is to model warehouse metadata, CWM exposes the information necessary to perform warehouse configuration maintenance operations.

4 Logical Schema Model Use Cases

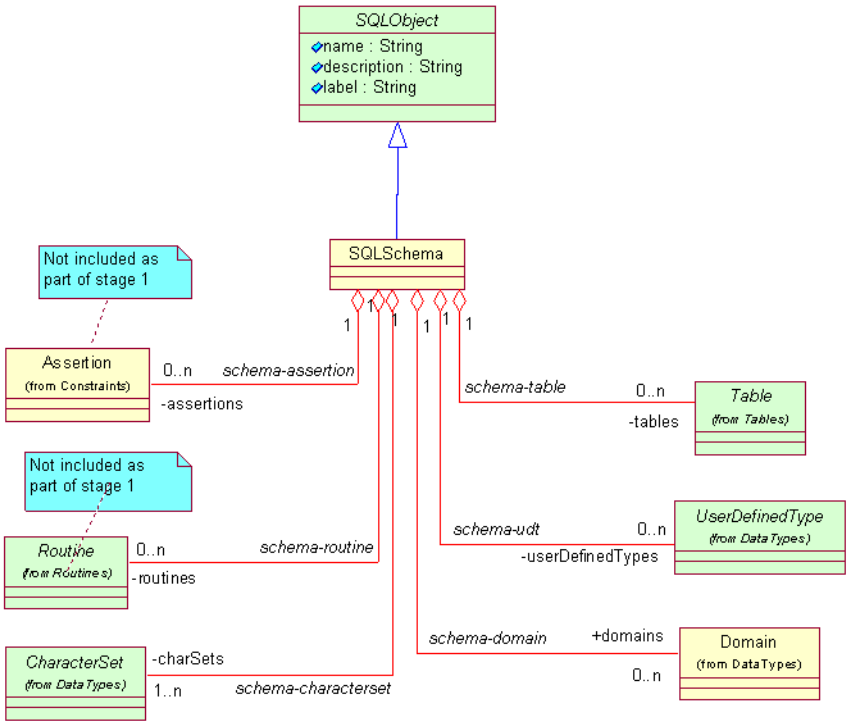
This section contains use cases for the logical schema model. A use case provides an example

5 The SQL Model (Stage 1)

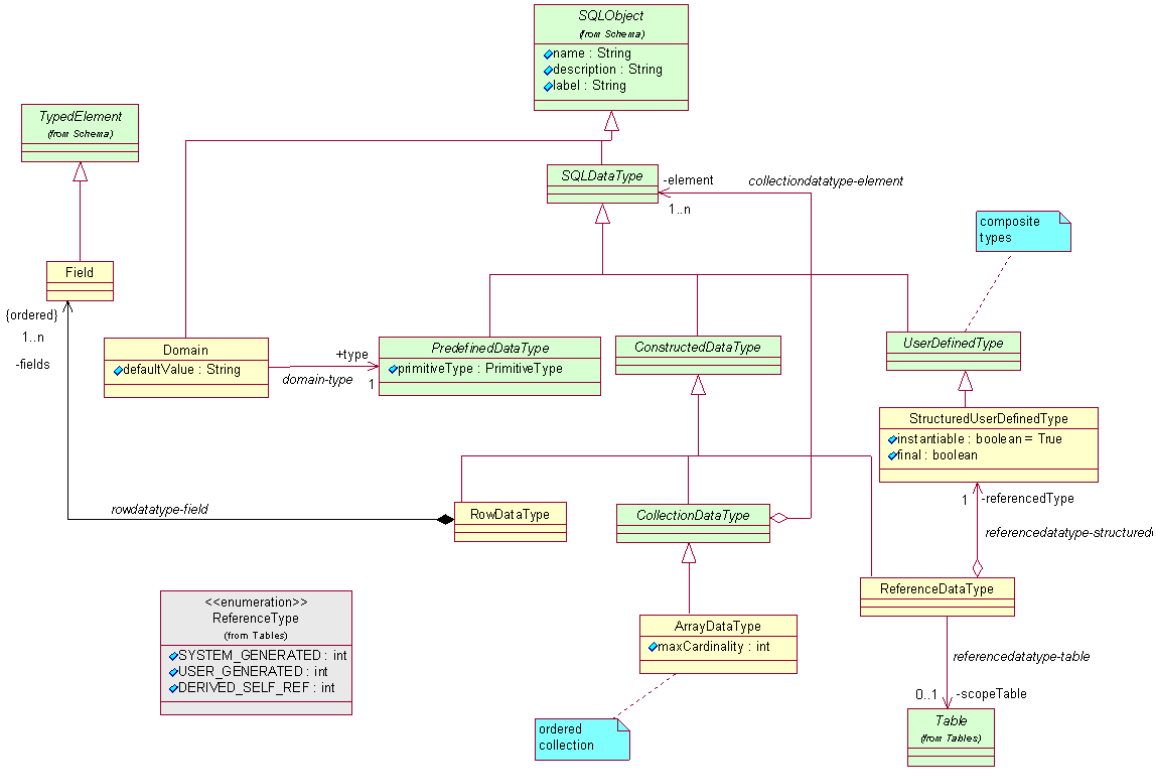
5.1 The Top Level



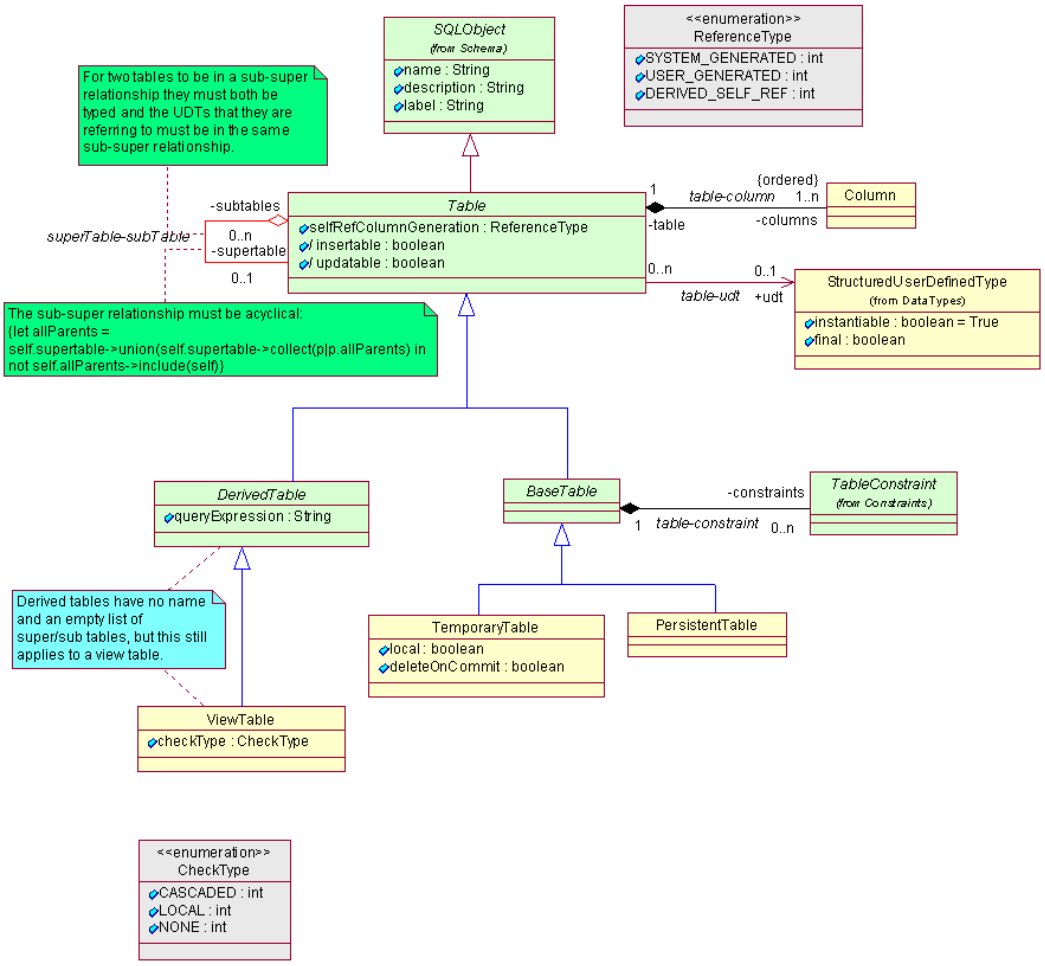
5.2 The SQLSchema



5.3 SQLDataType



5.4 The SQLTable



6 Future Work

Appendix A - Change History

Appendix B - References

[JSIM]
Job Submission Interface Model, downloadable from
http://www.ggf.org/documents/GWD-I-E/GFD-I.028.pdf

[CIMCore]
CIM Core Model White Paper, downloadable from
http://www.dmtf.org/standards/documents/CIM/DSP0111.pdf

[CIM2.9prelim]

DMTF CIM Schema, Version 2.9 Preliminary,
http://www.dmtf.org/standards/cim/cim_schema_v29_prelim

[CIMDatabase]
CIM Database Model White Paper
http://www.dmtf.org/standards/published_documents/DSP0133.pdf

[ANSISQL]
American National Standards Institute (ANSI) – SQL 1999
http://www.ansi.org/

Appendix C - Extending the Model

Appendix D - Considerations for Implementation

Acknowledgements

Full Copyright and Intellectual Property Notices

Full Copyright Notice

Intellectual Property Statement

The GGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the GGF Secretariat.

The GGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the GGF Executive Director (see contact information at GGF Web site).