NAMING SYSTEM – SCHEMA DESIGN

Version 2.3

Domain Lead	Vasu Vuppala
Design Team	Vasu Vuppala

Revision History

Version	Date	Author	Description
2.3	5/21/13		Initial version

Table of Contents

List of Figures and Tables

1 Introduction

A Particle Accelerator Facility (PAF) has to uniquely identify its installed devices and their signals. These identifiers are essential for addressing and operating the devices, and in describing their configuration in the facility. Generally, a hierarchical scheme is used to generate the unique identifiers. Note that this identifier uniquely identifies the location where a device will be installed, and not the device itself, i.e. it identifies an address in the facility where a device can be located. The identification of the devices themselves is done through a different scheme (serial numbers). Due to hierarchical nature of the scheme, an identifier is composed of several parts. We call such parts as *names*, and the identification scheme as *Naming Convention*. Figure 1 shows an example naming convention, and LS1_CK01:QH_D1800:ACC_CSET is an identifier based on this convention where LS1 is a System, CK01 is a Subsystem, QH_D1800 is a Device, and ACC_CSET a Signal.

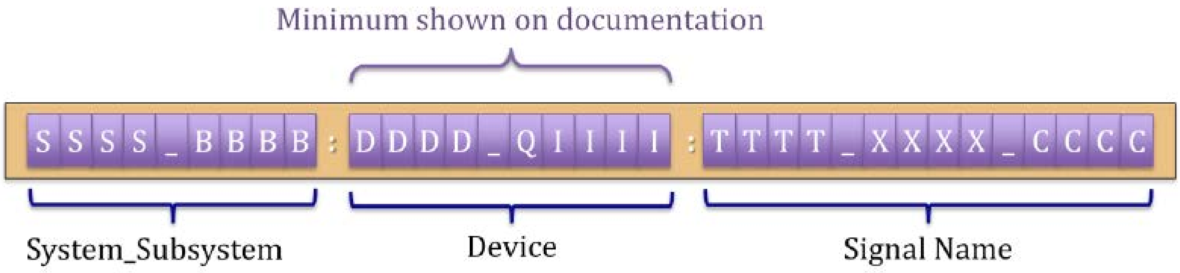

Minimum shown on documentation
S S S S _ B B B B : D D D D _ Q I I I I : T T T T _ X X X X _ C C C C
System_Subsystem
Device
Signal Name

Figure 1 Example Naming Convention

Management of the Naming Convention can become onerous during the construction of a PAF. There are frequent changes to the convention. These changes have to be controlled, and communicated to the entire organization. Changes are to be quickly processed, so that the new names can be used immediately. People need quick and easy access to the current approved naming standard which keeps changing.

Proteus: Naming System is a software tool that helps in the management of the naming convention. It is a web-based system that allows users to search the naming standard, and request for changes. The users can see pending and approved name changes. The administrators can process the changes online, and have the results available to the users instantly.

1.1 Purpose

This document describes the design of *Proteus: Naming System*.

1.2 Definitions, Acronyms, and Abbreviations

Table 1 Definition, Acronyms, and Abbreviations

Item	Description
DBMS	Database Management System

DISCS	Distributed Information Services for Controls Systems
EPICS	Experimental Physics and Industrial Control System
ERD	Entity-Relationship Diagram
FRIB	Facility for Rare Isotope Beam
IOC	Input/Output Controller
RDBMS	Relational Database Management System
REST	Representational State Transfer

2 Database Design

2.1 Conceptual Model

Generally the name is hierarchical, and is composed of several parts. Each part represents a *Name Category*. For example, in Figure 1, *System*, *Subsystem*, *Device*, and *Signal* are name categories.

2.1.1 Name Lifecycle

A name goes through several phases in its lifecycle:

- Inception: When a user requests for a new name
- Modification: When user requests for changes to a new name
- Retirement: When a name is removed from the Naming System

A name can be in the following states:

- Approved: It is approved for publication. Even though the name has not been published, it can be used.
- In Process or Awaiting Approval: A user has requested for a change, and the change is being processed. It is recommended not to use a name which is in this state.
- Published: The name is part of the current Naming Standard.
- Deleted: The name has been removed from the standard. A name in the state, obviously, must not be used.

Many of the events in the lifecycle occur due to change requests from users.

2.1.2 Concepts, Entities, Objects

- Name Event[1]:
  - Definition: An event (or change request) in the lifecycle of a name
  - Examples: Add a name, modify the description of a name
- Name Category:
  - Definition: Parts of the hierarchical naming convention
  - Examples: System, Subsystem, device-type, signal-type

[1] ‘Change Request’ or ‘Name Request’ may have been a better moniker.

- Name:
  - Definition: Instance of a Name Category
  - Examples: LS1, CK01, QH
- Release:
  - Definition: Names are periodically published to a standard document. This is called a new Release. It is identified by a unique name. It is associated with a date (release date).
  - Examples: R003 2013-05-01, R002 2013-03-15

2.1.3 Relationships

- A Name belongs to only one Name Category. A Name Category may have several associated Names.
- A Name can have one or more Name Events associated with it. A Name Event is associated with only one Name.

2.1.4 Names vs Events

It is natural to consider Name and Name Event as two entities and model them separately in the database. However, every Name has at least one Name Event: inception. Hence, another way to model this would be to have only Name Event, and derive everything about Name from it. This makes it simpler. All the Name Events related to a Name provide the Name's history, from inception to retirement.

2.1.5 Name Status

If we do not model Name as an entity, a Name's status has to be derived. Name Events and Name Releases are used to derive the status of a name as shown in Table 2. We first take the latest valid event (not cancelled or rejected) on a name, and then use it against the table to get the name's status.

Table 2 Name Status

Event Type	Event Status	Event Request Date		Name Status
Add or modify	Being Processed	After latest release date		In Process
Add or Modify	Approved	After latest release date		Approved
Delete	Approved	any		Deleted
Add or modify	Approved	Before latest release date		Published

2.1.6 Conceptual Diagram

Figure 2 shows the entity types (or classes) and the relationships among them. The notation used in the figure is described in Appendix A – Notation.

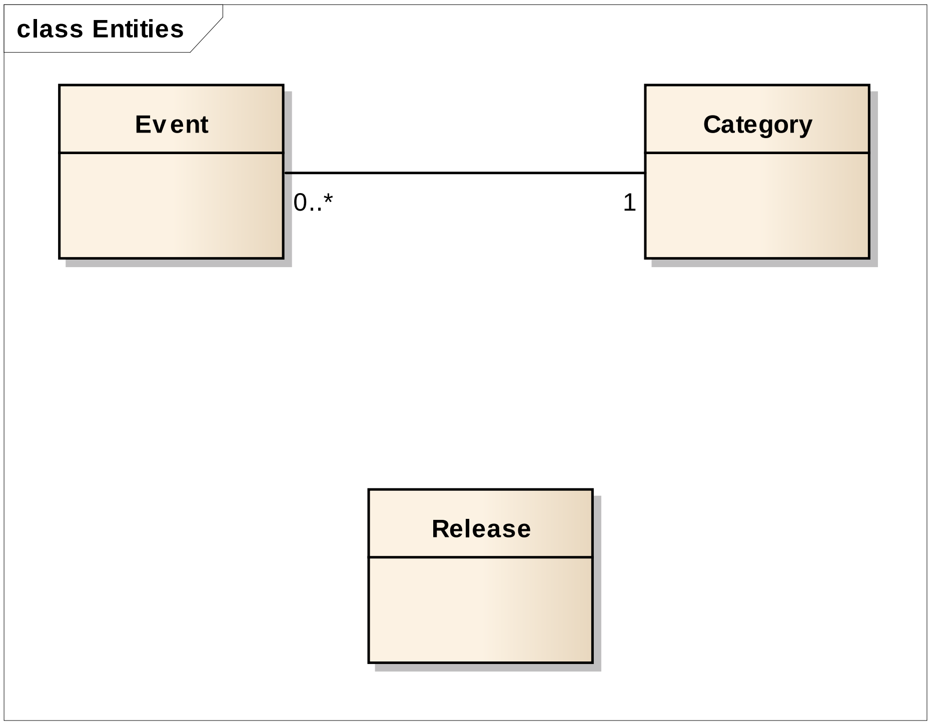


Figure 2 Conceptual Model

2.2 Physical Model

2.2.1 Relational Model

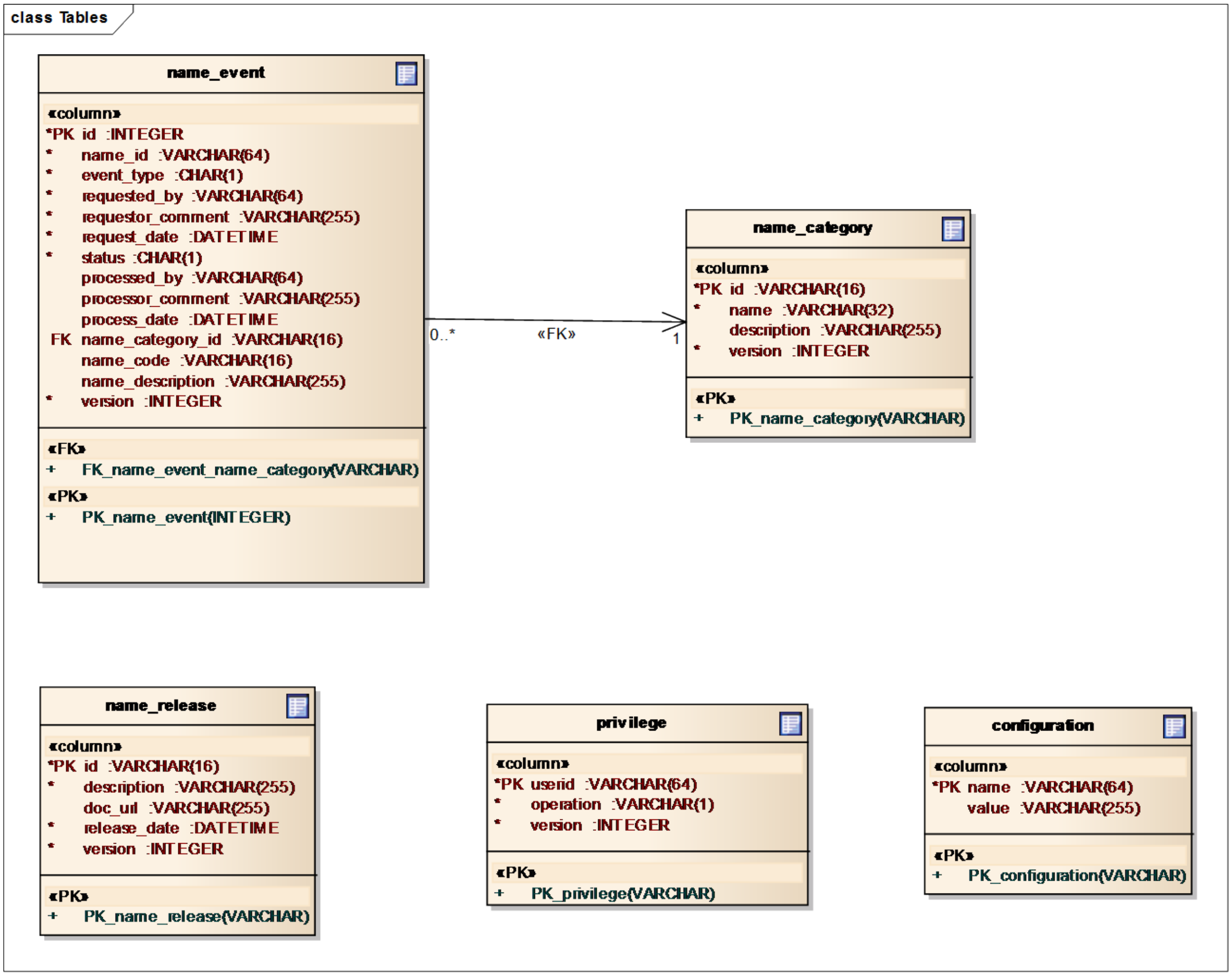


Figure 3 Physical Model

2.2.2 Table Description

All tables have a column named *version* for optimistic concurrency control.

1. name_category: Each row represents a name category

Attribute	Description
id	Unique identifier
name	Category name
description	description

2. name_event: Each row represents a name event

Attribute	Description
id	Unique identifier (for events)

name_id	Unique identifier for names
event_type	i – initiate (Add), m – Modify, d - Delete
requested_by	User id of the requestor
requestor_comment	Comment from requestor
request_date	Date of request
status	a – approved, b – being processed, r – rejected, c - cancelled
processed_by	User id of the person who processed the event/request
processor_comment	Comment from the processor
process_date	Date of process
name_category	Name category. Ex: system, subsystem, device-type
name_code	The actual name. Ex: QH, CAV, AC
name_description	Descriptive name

3. name_release: Each row represents a release

Attribute	Description
Id	Unique identifier
description	Description of the release
doc_url	URL pointing to the released document
release_date	Date of release

4. privilege: Each row represents a user privilege

Attribute	Description
userid	User
operation	E – edit. User with Edit privilege can process events/requests, and publish releases.

5. configuration: Each row represents a configuration property

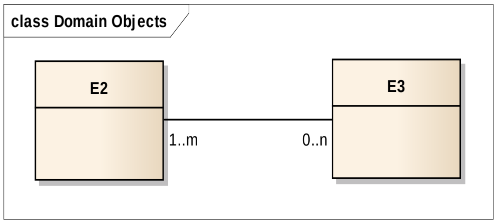
Attribute	Description
Name	Property name
Value	Property value

3 References

1. FRIB Naming System Document, FRIB-T10500-PR-000001, FRIB Portal.

Appendix A – Notation

Notation Example:



1. E2[1..m] - E3[0..n] means:
    a. An instance of E2 can be related to 0 or more instances of E3
    b. An instance of E3 can be related to 1 or more instances of E2
2. E2[1..1] - E3[0..n] means:
    a. An instance of E2 can be related to 0 or more instances of E3
    b. An instance of E3 can be related to exactly one instance of E2
3. E2[3..6] - E3[2..5] means:
    a. An instance of E2 can be related to 2 to 5 instances of E3
    b. An instance of E3 can be related to 3 to 6 instances of E2