LSAL QuickTip: Local Names

Deciding on a Local Name system for identifiers is an important aspect of an organization's readiness for the ADL-Registry. This QuickTip summarizes the two basic methods for creation & assignment of local names.

Overview

In the ADL Registry (ADL-R), everything that you intend to register (e.g., content, metadata, repositories) is identified by a handle. Handles have two parts:

Part 1: Naming Authority	Part 2: Local Name
• Unique within the Handle System • Identical for each handle created by your organization (e.g., "100.123") • Assigned to your organization by the ADL-R	• Unique within a Naming Authority • Assigned to everything that you create (e.g., "local1234") • Chosen by your organization

The complete handle is a combination of the naming authority and the local name; in the example above, the handle would be 100.123/local1234. The Handle System guarantees the uniqueness of local names because it does not allow duplication when a handle is registered. However, your organization must decide how it will create local names.

Local names, or identifiers, are generally created by:

- Defining semantically meaningful local names; or
- Creating arbitrary local names.

Options for Identifiers

An identifier has a semantically meaningful name when it conveys some meaning about the object to which it is assigned. For example, the identifier "100.123/chapter-1-page-1" is likely about the first page of the first chapter of a book or course.

An "arbitrary" identifier conveys no meaning. For example, "100.123/1897a90gh998b87" is a random string of characters that carries no meaning about the object that it identifies.

Semantically Meaningful Names	Arbitrary Names
• Are short-cuts for humans to know something about the object by looking at the identifier itself	• Are generated by a machine based on a meaningless numbering system or a series of random characters

• May speed up tasks that require a human to interact with identifiers
  o Creating hyperlinks by hand
  o Looking at files in a directory
	• Are difficult for a human to distinguish, use, and verify
	• Encourage using a repository and searching metadata to locate objects

• Can be problematic when different groups assign different meaning to the same identifier o Unique language or cultural uses	• Used in automated systems that don't require users to see the identifiers • Forces users to focus on the object itself or its metadata

hdl:1870/8F7816AE761A4BDD91D6AF8F385D3F9F

o Individual business practices	for a more methodical and thorough workflow
• Can be challenging when the objects they reference are reused in a different context	• Do not cause confusion when an object is reused in different contexts

As a best practice, you should never attempt to mix semantically meaningful names with arbitrary names within the same naming authority.

Creating Consistent Semantically Meaningful Identifiers

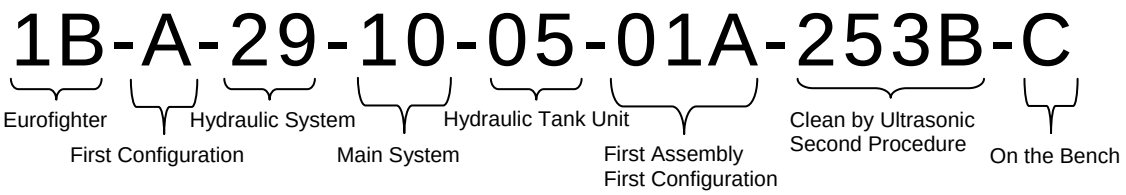
There are several possibilities for creating semantically meaningful local names in an identifier. Choose a system that allows objects to be reused in different contexts without changing the identifier. This may require your organization to define a standard taxonomy that everyone will use to create their identifiers. For example, your local name could be based on:
topic – sub-topic – version – date created. Using this naming taxonomy, your identifier might be *sanitation-hand-washing-illustrated-v1r0-2005-05-29*.

The S1000D Example

The S1000D specification, used widely in the aviation industry, identifies objects using a Data Module Code (DMC). A DMC is composed of two principal parts:

- Identifier of the physical item being described, according to a specific taxonomy and using a central repository to ensure that each identifier is unique.
- Identifier for the information type of the text (to represent actions such as “operating” or “storing”).

An example of a DNC is “1B-A-29-10-05-01A-253B-C”. Someone working with this identifier would recognize it as



Creating Consistent Arbitrary Identifiers

There are two recommended ways to create an arbitrary identifier (1) distributing identifiers from a central source or (2) creating random sequences of numbers and letters. These methods are not mutually exclusive – a central source could hold pre-generated random number strings that are handed out on demand.

Central Source	Random Number Generators (GUIDs)
• Prevents duplication of identifiers	• Duplication of identifiers is possible, though the possibility is low
• Typically begins with a number and then distributes identifiers sequentially as they are requested (e.g., 1000, 1001, 1002, …)	• Typically are very large and represented as 32 character strings (like the footer in this document)

Resource List

S1000D, http://www.s1000d.org/
Sample GUID Generator, http://services.lsal.cmu.edu/guidservice/guidservice.asmx

hdl:1870/8F7816AE761A4BDD91D6AF8F385D3F9F