Overview

Donatella Castelli

CNR-ISTI

Before we talk about ~~SOA~~ DLs and where we want to go

- Would be nice if we had consensus on what DLs are!
  - Is it sufficiently different from other types of information systems? If a DL is “X”, what is not a DL?

Duane Nickull
Senior Technical Evangelist
Adobe Systems, Inc.
http://www.oasis-open.org/

What is a Reference Model?

- A reference model is an **abstract framework** for understanding significant relationships among the entities of some environment, and for the development of consistent standards or specifications supporting that environment

- A reference model is based on **a small number of unifying concepts** and may be used as a basis for education and explaining standards to a non-specialist

- A reference model **is not directly tied to any standards, technologies or other concrete implementation details**, but it does seek to provide a common semantics that can be used unambiguously across and between different implementations

The World of Digital Libraries: DL

1. Digital Library

A (potentially virtual) organization that comprehensively collects, manages, and preserves for the long term rich **digital content** and offers to its **user** communities specialized **functionality** on that content, of measurable **quality**, and according to prescribed **policies**

The World of Digital Libraries: DLS

2. Digital Library System

A software system that is based on a (potentially distributed) **architecture** and provides all functionality that is required by a particular Digital Library. When operating in a Digital Library environment, users interact with the corresponding Digital Library System

The World of Digital Libraries: DLMS

3. Digital Library Management System

A generic software system that incorporates all functionality that is considered foundational for Digital Libraries and provides the appropriate software infrastructure to both produce a basic Digital Library System and integrate additional software offering more refined, specialized, or advanced functionality.

An intrinsic part of a DLMS is a “Digital Library Administrative Tool” that is used to choose the appropriate subset of its functionality, e.g., through relevant parameters of its components, and then (potentially (semi-)automatically) install, deploy, and (re)configure a Digital Library System.

DLMS Types

- **Extensible Digital Library System**

  A complete Digital Library System that is fully operational with respect to basic/ foundational functionality required. It is based on an open software architecture, so that further software components can be incorporated on top of the ones already there with ease.

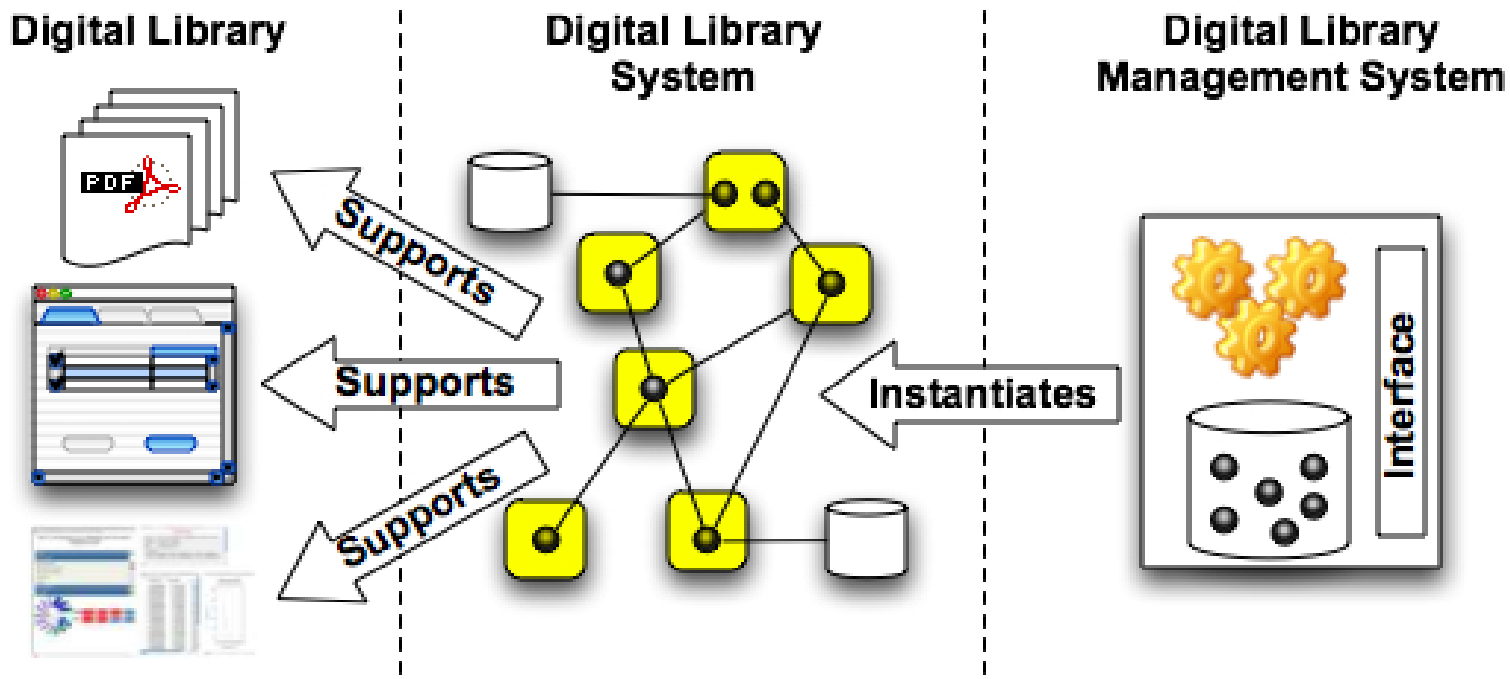
- **Digital Library System Warehouse**

  A collection of software components that cover all basic/foundational functionality required and a set of tools that can be used to combine these components in arbitrary ways (in Lego-like fashion) to create Digital Library Systems offering diverse sets of functionality.

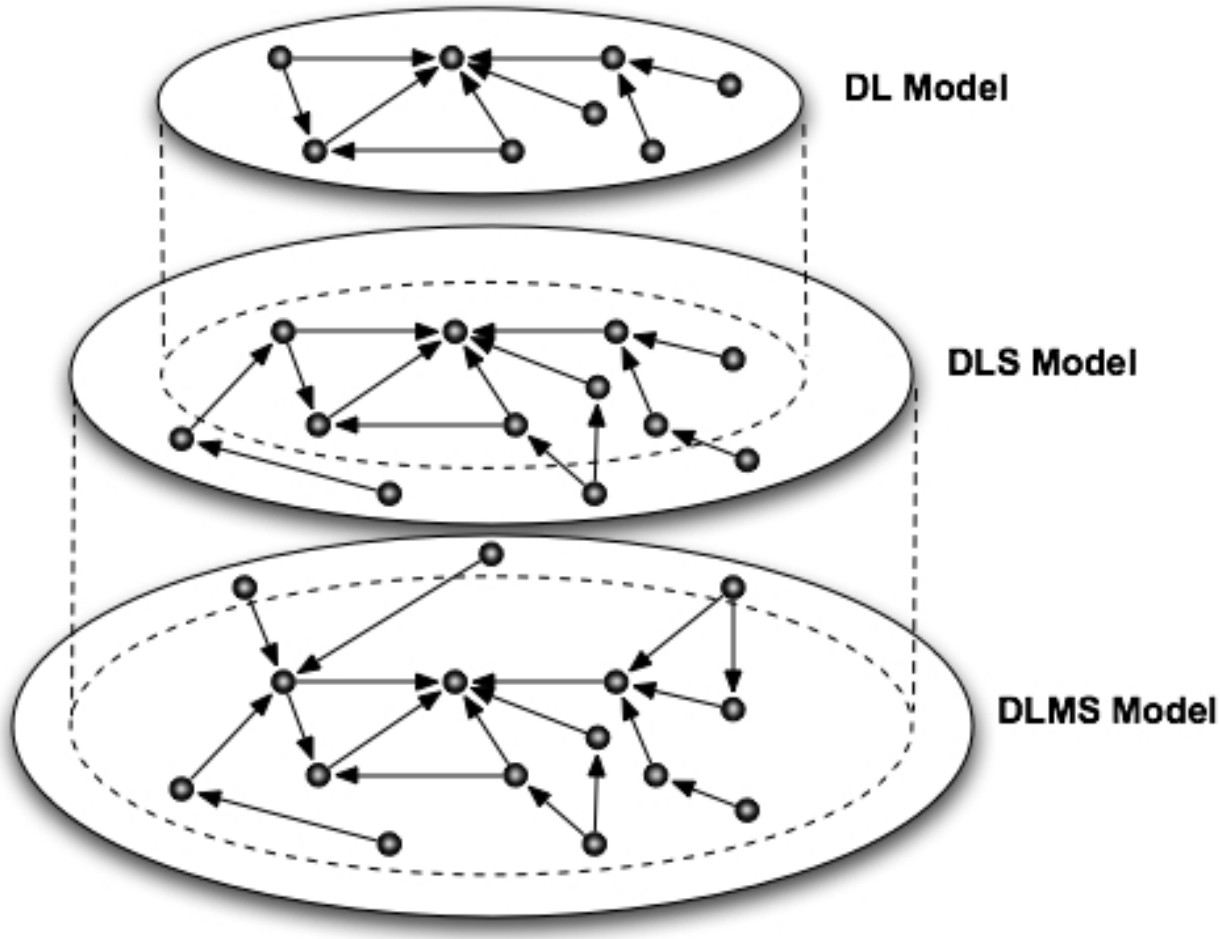
- **Digital Library System Generator**

  A highly parameterized software system that encapsulates templates for a broad range of functionality, which includes all basic/foundational as well as any advanced functionality that has been deemed appropriate. Through an initialization session, the appropriate parameters are set and tuned; at the end of that session, the Digital Library System software is generated.

A Three-Tier Framework



Models containment



Main Roles of Actors

- **DL End-users**

  Exploit the DL functionality for providing, consuming, and managing the DL information domain

- **DL Designers**

  Exploit their knowledge of the application semantic domain to define, customize, and maintain the Digital Library so that it is aligned with the information and functional needs of its end-users. To perform this task, they interact with the DLMS providing functional and content configuration parameters.

Main Roles of Actors (cont.)

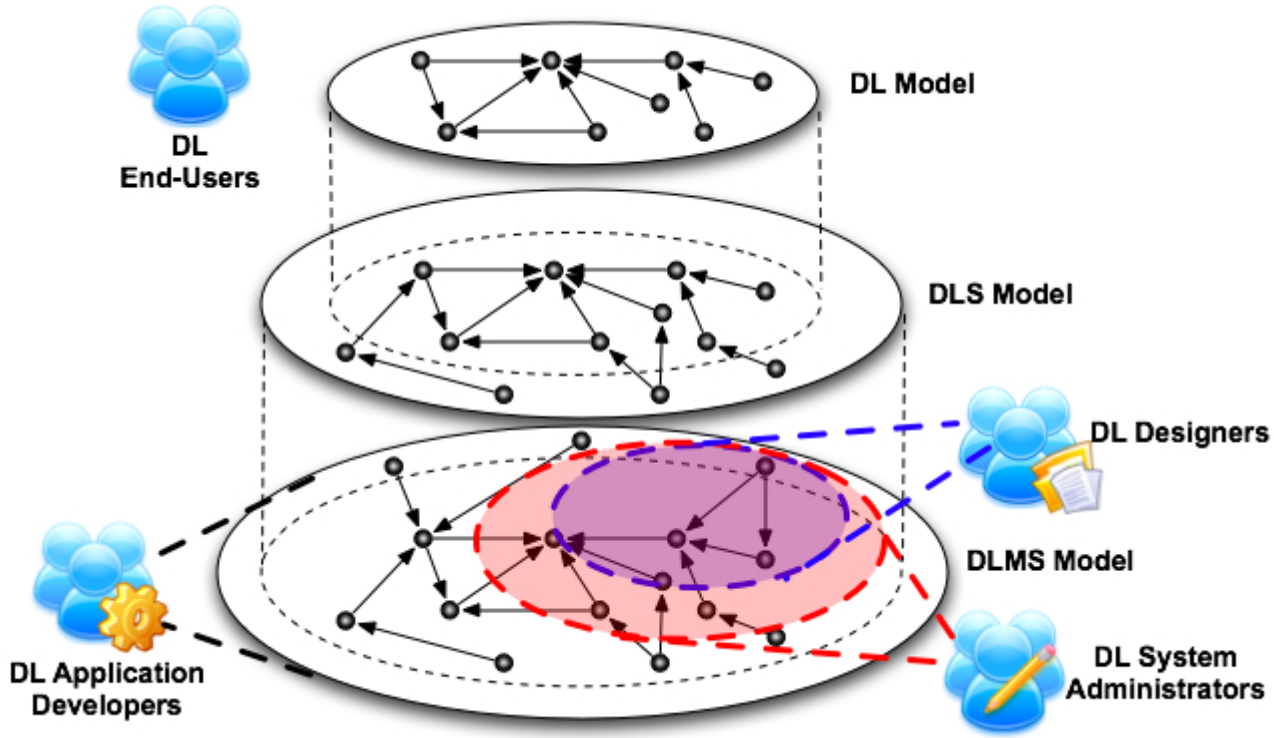
- **DL System administrator**

  Select the software components necessary to create the DLS that would serve the required DL and decide where to deploy them. They interact with the DLMS by providing architectural configuration parameters, like the selected software components, the hosting nodes, the components allocation, etc.

- **DL application developers**

  Develop the software components of DLMSs and DLSs, realizing the necessary functionality.

Role specific models



Main Concepts