



NSDL EduPak Roadmap - Objectives for 2010

January 19, 2010

NSDL Technical Network Services

Carl Lagoze, PI, Cornell University

Tamara Sumner, co-PI, University of Colorado and Executive Director, Digital Learning Sciences (DLS) at University Corporation for Atmospheric Research (UCAR)

Mike Wright, co-PI, Technical Director, Digital Learning Sciences (DLS), UCAR and Technical Network Services, NSDL

Karon Kelly, Director, Digital Learning Sciences (DLS), UCAR, and Technical Network Services, NSDL

EduPak Roadmap – Objectives for 2010

NSDL Technical Network Services, January 2010

Purpose of this Document. The purpose of this document is to outline planned changes and additions to this established NSDL package that NSDL Technical Network Service will be working on in 2010. This updated version reflects feedback and comments received on a draft Roadmap that was circulated, presented and discussed at the 2009 NSDL Annual Meeting and a subsequent online commentary period.

Audience of this Document. The audience for this Roadmap is NSDL stakeholders. This includes: (1) current and prospective NSDL grantees and the NSDL Resource Center, (2) key project partners, (3) users, including educational institutions and organizations interested in using the software package or the services this package provides (as deployed as infrastructure under nsdl.org) in their own sites and products, and (4) NSF personnel.

How to Comment on this Roadmap. While the official request for comments period is over, we always welcome suggestions and comments. Please contact TNS directly (<http://nsdl.org/about/contactus/>) or post your ideas in the discussion forum associated with this document on the NSDL Community site at <https://www.nsdlnetwork.org/>.

Background on EduPak.

The NSDL since its inception has maintained a central service at NSDL.org to discover education resources on STEM subjects. Information about educational resources has been created and collected in the form of metadata by many collection builders and providers including NSDL funded projects and NSDL Pathways. An early implementation of the service utilized a relational database into which metadata records (normalized to a framework influenced by Dublin Core denoted as nsdl_dc) were ingested, and search provided over the normalized records – effectively a union catalog. This database was known as the Metadata Repository (MR). Metadata was shared with the MR through use of the OAI-PMH protocol. The data model was considered metadata centric in following the union catalog approach where norms are to have a single, authoritative, metadata record about a resource. As the NSDL grew with different collection builders providing different metadata about the same resource, the MR approach began to have limitations.

The MR architecture was later replaced with a digital repository solution based on the open-source Fedora platform. Information (metadata) about resources, and relationships between these, are modeled as objects in a data model developed for NSDL, and the repository accessed through a web-services API. The repository is known as the NSDL Data Repository (NDR) (and associated NDR-API). Initially, metadata was still shared with the NDR using OAI-PMH, and then applications were developed, or repurposed, to work directly with the NDR via the NDR-API (e.g. Expert Voices blog environment based on word press). Search continued to be provided over the normalized metadata (nsdl_dc format). The key premise of the NDR data model is the idea of being resource-centric, i.e. keying off resources as

the primary item and allowing multiple collections and metadata to be associated with a unique resource, and managing the collection associations through objects identifying aggregations and metadata providers. Starting from a resource object (that holds the URL to the resource), an application such as the NSDL search can look for all the related metadata and collections for inclusion.

The prevailing view of the content in the NSDL has remained as collections of metadata provided and managed by an array of collection builders and maintainers. In order to manage collections directly in the NDR (rather than as harvested sets), the NSDL Collection System (NCS) was added to allow collection builders and catalogers to manage their own collections, in their own frameworks, in the NDR. The NCS takes care of the mapping to the internal NDR data model and normalization to nsdl_dc in the background (for use by the NSDL search service at NSDL.org). The Digital Discovery System (DDS - a search component accessed through a REST-based web-service API) was added to provide a customizable search solution that can work with different metadata frameworks thus providing the ability to embed that search in different portals or applications.

As this architecture has matured with the operational NSDL, the underlying software systems that provide the repository (NDR), collection management (NCS) and custom search (DDS) have been released as an open-source solution bundle called EduPak.

2010 EduPak Roadmap

In response to the NSF Cyberlearning report and subsequent interest by NSF, there is a shift in focus toward a platform perspective where “learning applications” build on a platform. In the case of the core NSDL infrastructure at nsdl.org, the platform is a deployment of EduPak and related services such as OAI harvesting.

For the purposes of this roadmap, learning applications are considered to be applications that are focused on supporting the day-to-day practices of teachers and learners. Such applications are oriented with a curriculum, or project, and have a need to bring in related materials and resources, allow for rearranging, commenting on, tagging for the particular learning situation, and sharing those with other educators or students. Some specific examples of past and present learning applications in NSDL are the Instructional Architect (for teachers to create and share projects), the Digital IdeaKeeper (geared for student use), and the Curriculum Customization Service (built to support teachers in planning and organizing around the core curriculum and instructional goals of a school district). The Instructional Architect (IA) is now hosted at nsdl.org, however the IA software is aging and the IA PI would like to see this migrate to the NSDL platform. The Digital IdeaKeeper was an early NSDL funded project that ended some years ago and the software system is no longer supported. To bring the value of that project back to use would be to build on the NSDL platform.

In each of these examples, users of the applications can include library resources, tag them, create notes, upload documents to a personal (or project) space and share those with others. There are similar examples of personal portfolios in other NSDL projects (e.g. TeachersDomain). Other projects have expressed their need to be able to add supplemental information to collections, e.g. the AAAS services project to evaluate resource quality and alignment. The work outlined in this roadmap will provide the

means to consider all these cases, and to allow work toward the deployment of personalization services in NSDL by allowing applications to build on the NSDL platform. To meet the goals, we will provide:

- Ability to annotate other records – i.e. insert extra information that supports metadata already in the NDR as distinct from adding to a resource.
- Better support of the collection management process through the NDR-API such that applications don't have to know low-level details about how collections are represented in the NDR and to reduce the number of "calls" to the NDR-API an application must make.

Provide a simpler API to NDR and add annotations

Recommendations:

- Begin development of a Collections API to the NDR that aligns with collection management practice
- Include annotations as part of the new Collections API (extending the data model)

Work to be performed: TNS staff will develop and test a Collections API to provide an interface with the NDR that abstracts how the NDR data model is implemented in the NSDL deployment case. The NSDL deployment case is that underpinning the core library at nsdl.org. This API will be developed to satisfy the use cases of the current tools managing collections – the NCS, DDS, OAI and Web Feed Ingest (WFI) processes, and the processes supporting the current NSDL search service. The current NDR-API would remain in place to allow for cases where applications may need to work directly with the base objects.

As part of the API development, the concept of annotations will be included as specifically the concept of allowing an arbitrary metadata to reference other metadata in any arbitrary collection in the NDR. This will satisfy the current annotation use cases from some learning applications (the NSDL Curriculum Service and Instructional Architect) and from other projects who want to add these type of annotations (e.g. newly funded AAAS services project to look at quality and alignment). Support for annotations, and their management through the Collection API will be tested against the CCS and IA cases, and the AAAS services work to begin with. As other NSDL use cases are identified, they will serve to provide further test cases.

When: The Collections API specification will be released by March 1, 2010, so it is available for reference in the 2010 NSDL RFP process. Implementation will proceed through early 2010. This will allow for other applications to be updated and subsequent roll out to production by August, 2010.

Stakeholders impacted: NSDL users, NSDL grantees who want to integrate with NDR, NSDL grantees with annotation needs, past NSDL grantees (moving past systems onto the platform)

Rationale: While EduPak is considered a general solution for building a digital library, it must be configured for any particular deployment case. The primary deployment case is the NSDL platform underlying nsdl.org. A consequence of that deployment is the way the base NDR data model is configured for the NSDL case. The base model comprises five types of object; Agent, Aggregator, Metadata, MetadataProvider, and Resource. Objects have properties, relationships (to other objects),

and datastreams (essentially buckets for content). In the NSDL deployment case, various property names are reserved in objects (e.g. OAI set information such as set specifications held in the MetadataProvider object), datastreams are reserved (e.g. an nsdl_dc stream in a metadata object) and administrative information about collections is held as a specific collection in the NDR (and managed through an NCS). This information is an overlay of the base NDR data model specific for the NSDL deployment case, and provides a practical example of an EduPak deployment.

Currently, applications that work with the NDR-API must all include the logic to deal with all the facets of the information held thus placing a burden on the developers and applications. The Collections API will reduce this burden and provide an easier path for others to integrate with the NDR. It will also mean application developers and service users will not have to work with the underlying NDR data model implementation. The separation of the collection view will allow adjustment of the data model in the NDR without needing to re-implement applications that work with the NDR through the Collections API.

In developing a Collections API, it will provide the opportunity to include the addition of annotations such that tools using the new API can insert and manage collections of annotations. Annotations are metadata referencing other metadata in any arbitrary collection in the NDR, thus are providing supplemental information in the context of the “annotated” metadata. The provision of annotations will provide a structure to support the development of learning applications that require mechanisms such as tagging.

Risks: While the basic NDR data model and NDR-API are generic, the Collections API would necessarily be tied to the NSDL deployment information model. Mitigating this is the fact the current NDR-API will remain (which allows interaction with the base data model independent of any particular deployment case).

The model continues to maintain a focus on metadata, or collections of metadata, in the NDR by now adding annotations as metadata that reference other metadata. This approach does not make extensive use of features of the basic NDR model such as object properties. Mitigating this again, is that the API can provide a means to go back and adjust the underlying model to make more use of the NDR object model.

Further information:

- <http://wiki.nsd.org/index.php/Community:NDR>
- <http://wiki.nsd.org/index.php/Community:NCS>
- <http://wiki.nsd.org/index.php/Community:DDS>
- <http://wiki.nsd.org/index.php/Community:EduPak>