18.1 Theory

In his book “Libraries of the Future,” J.C.R. Licklider predicted that a world network of automated libraries would together support global information requirements, supported by a hodgepodge of different technologies, since there was no integrated theory (Licklider 1965). After more than 40 years, it has become clear that advances in understanding and technology have allowed a theory-based approach to this field that should aid those interested in computer science, information systems, and/or information technology to better grasp the key concepts regarding digital libraries (DLs).

18.1.1 DL Definitions

Though storing and retrieving information with the help of computers dates back at least to the 1950s (Taube 1955), those focused on what has emerged as the field of DLs did not select that phrase until about 1991, giving it preference over terms such as electronic library or virtual library.
Accordingly, some have been interested in defining what constitutes a DL as a step in clarifying its particular place in the digital world. Three illustrative examples of such definitions follow:

1. “A digital library is an assemblage of digital computing, storage and communications machinery together with the content and software needed to reproduce, emulate and extend the services provided by conventional libraries, based on paper and other material means of collecting, cataloging, finding and disseminating information” (Gladney et al. 1994).

2. “A set of electronic resources and associated technical capabilities for creating, searching and using information. In this sense they are an extension and enhancement of information storage and retrieval systems that manipulate digital data in any medium (text, images, sounds; static or dynamic images) and exist in distributed networks. The content of digital libraries includes data, metadata that describe various aspects of the data (e.g., representation, creator, owner, reproduction rights) and metadata that consist of links or relationships to other data or metadata, whether internal or external to the digital library” (Borgman 1996).

3. “Digital libraries are a focused collection of digital objects, including text, video and audio along with methods for access and retrieval and for selection, organization and maintenance of the collection” (Witten and Bainbridge 2002).

18.1.2 Digital Library Curriculum

By 2005, many DLs were in widespread use, providing increasingly popular means of collecting and finding information. It became clear that educational and training support was needed for the emerging community of digital librarians. Toward this goal, a team at Virginia Tech and the University of North Carolina at Chapel Hill collaborated to develop a curriculum, founded in theory and practice, that could be used to teach about DLs in library schools, information science departments, or computer science programs (Pomerantz et al. 2006). Figure 18.1 gives one perspective on how some of the modules that have been developed could be used in a two course sequence; it also demonstrates many of the key topics associated with DLs. An increasing number (already more than 40) of learning modules, each supporting either self-study, tailored training, or roughly a week of coverage in a course, can be accessed both through Wikiversity and our website (Fox et al. 2012b).

18.1.3 Formal Approaches

In computing, rapid advances often follow the development of a theory that enables a formal approach to practical problems, for example, the use of the relational model for databases (Codd 1970). In the DL field, early efforts ignored the use of formalisms, so the availability and features associated with services were haphazard. Clearly, the community of DL producers can benefit from formal definitions of existing and proposed DLs, along with the services they provide. Formalisms within the field of DLs provide a theoretical foundation. This can save work for designers and implementers, for example, by enabling the reuse of formally and programmatically defined services, and the deployment of existing service modules in the generation of proposed DLs for a specific context. The two major efforts toward a theoretical framework include the 5S formal framework (Goncalves et al. 2004; Fox et al. 2012a) and the DELOS reference model (Candela et al. 2011).

18.1.3.1 5S

The 5S (streams, structures, spaces, scenarios, and societies) framework builds on five precisely defined yet intuitively understandable basic constructs, allowing a high level foundation to be used to describe all key DL concepts (Fox et al. 2012a). Streams encompass sequences of information that represent content. Streams include audio, video, bit streams on disk or transmitted over a network, and character strings that constitute text. Structures represent any type of organization (e.g., a database, taxonomy, thesaurus,
dictionary, ontology, data structure, record, file in a particular format, or any digital object following some type of schema) and can be described using labeled graphs. Spaces are sets of items along with rules and operations valid for each set of items. In addition to the usual 1D, 2D, and 3D spaces, there are vector, probabilistic, and topological spaces. Typical user interfaces are implemented on a 2D space, while many search engines are based on a vector space model. Scenarios are sequences of events that lead to an outcome. All services follow scenarios, as do workflows, procedures, and algorithms. Societies are made up of entities, either human or computing-based, with relationships among them. Thus, they encompass the set of end users, groups engaged in collaboration, social networks, and other aggregates studied in psychology, the social sciences, or fields like law or economics. As can be seen in Figure 18.2, each of the five constructs can be used alone or in combination with the others to define every one of the key constructs we consider to be required parts of even the most minimal DLs.
The 5S formal framework also may be used to encode descriptions of DLs and related concepts. Common services required by a range of DLs were initially defined, for example, indexing, searching, browsing, and annotation (Goncalves et al. 2008). The framework has since been extended to describe concepts found in a number of different domain-specific DLs (Murthy et al. 2007). Such extensions allow for emerging classes of DLs to be formalized, for example, for content-based image retrieval (Murthy et al. 2011) and scientific workflow systems (Leidig et al. 2010). Definitions of new services or aspects to DLs may build on top of existing definitions, and producing formal descriptions of DLs is facilitated through reuse of the existing set of definitions. Additional research connected with 5S has produced tools to aid in developing DLs, involving quality evaluations (5SQual) (Moreira et al. 2007), a description language (5SL) (Goncalves and Fox 2002), and graphical representations (5SGraph) (Zhu 2002). The 5S framework has been used in multiple DLs including Networked Digital Library of Theses and Dissertations (NDLTD), electronic tools and ancient near east archives (ETANA), Ensemble, and scientific DLs (Fox 1999; Ravindranathan et al. 2004; Fox et al. 2006, 2010; Leidig et al. 2011).

18.1.3.2 DELOS

The DELOS digital library reference model aims to provide a foundation for describing, understanding, and evolving the field of DLs. The model describes the DL universe in terms of digital library management systems (DLMS), digital library systems (DLS), and DLs. A DL instance is a practically running system that is perceived and used by end users. A DLS is the software implementation of the required functionality of a class of DLs. A DLMS addresses a conceptual problem, provides a software solution, and implements a DL approach as given in formal definitions of architecture, content, functionality, policy, quality, and users. The reference model has been repeatedly updated to provide concept maps, definitions, and conformance evaluation checklists (Candela et al. 2011).
We believe that mappings could be defined between concepts in the reference model and the 5S framework, so either approach might be employed, depending on the situation at hand. However, we prefer (1) a bottom-up approach to theory development, with everything precisely defined; (2) starting from a minimalist perspective; and (3) growing the framework incrementally as needed. Hence, we focus on 5S and use it in the rest of this chapter.

18.2 Data, Information, and Knowledge

Content is an essential part of a DL. The computing-related fields generally view content from varied perspectives based on decomposition into three levels of abstraction or complexity: data, information, and knowledge. This is particularly useful since it encourages use of tailored services inside a DL, such as database management systems, search engines (sometimes based on information retrieval toolkits), and knowledge management systems (e.g., for managing ontologies). Yet, DLs also have simplified work with content by shifting focus to a different set of abstractions, discussed in the subsequent sections.

18.2.1 Digital Objects

In a DL, the simplest abstraction for an element of content is to consider it as a digital object. Clearly, with the help of computers, much content is created in a digital form, like this chapter. Nevertheless, there are myriad objects in the real world that are not in digital form, such as physical books, paintings, statues, buildings, chemical or biological samples, and mathematical constructs. Fortunately, building on practices in linguistics and information science, we can discuss such objects if they are assigned a name, identifier, descriptor, and/or surrogate (e.g., a photograph or image prepared by a scanner). Thus, we can simplify the discussion of the content of a DL if we assume that anything of interest has some associated digital representation, allowing us to focus solely on such digital objects.

18.2.1.1 Atomic/Complex

There are many types of digital objects that can be stored in a DL. Some basic types include text, image, audio, and video. Each of these types has different formats and structures, for example, text formats include web pages, PDF, XML, or any file that contains a stream of characters. Digital objects also can be categorized based on the complexity of the structure of the object as either atomic or complex. Atomic objects are the basic unit of storage in a DL. All the basic types can be considered atomic objects. Basic types cannot be decomposed into smaller objects. Complex objects are objects that are composed of an aggregation of atomic objects into one unit with a certain structure that records the relationships between the different components (Kozievitch et al. 2011). Complex objects also can be constructed using parts of several other objects. Extending this notion further, we refer to the concept of extracting a part of an object (e.g., a sub-document) and treating it as a whole unit by itself, together with connecting together a set of such units, as superimposed information (Murthy et al. 2004). This approach provides a foundation for any type of annotation system and leads to a class of DLs that is particularly useful for a wide variety of scholarly activities (Murthy 2011).

18.2.1.2 Streams, Structures, and Spaces

These three constructs were briefly introduced earlier in Section 18.1.3.1. Their utility in describing the content in DLs needs further elaboration, however.

Streams are sequences of objects, for example, bits, bytes, characters, numbers, images, audio samples, and clicks. Thus, streams can be constructed from objects with different granularity levels. For example, a text is based on characters, while a book is based on pages. Streams may be static, as when a sequence of characters makes up a name or an abstract. Or, they may be dynamic, as in the stream of data that comes from a sensor or surveillance camera, or that is recorded into a log. In part to support all such dynamic streams, data stream management systems (Arasu et al. 2004) have been developed.
Structures also are essential in DLs. Understanding data structures, related algorithms, and their use (Shaffer 1997) is an essential basis for working with structures in DLs. This also aids the use of special languages used to describe structures, such as schema for relational databases, XML (Sperberg-McQueen and Thompson 2000) files/databases, or resource description framework (RDF) (Brickley and Guha 2004) (triple) stores. Of particular importance are record structures, commonly used for metadata (see Section 18.2.5). DLs support other important types of structures, such as classification systems, taxonomies, dictionaries, ontologies, databases, and hyperbases. Furthermore, when combined with streams, we have structured streams, which lead to constructs like documents, protocols (e.g., the organization of the streams connecting clients and servers), and files (that generally have some format or type, which specifies how the bytes are organized to carry meaning).

Spaces support representations that can help in organizing, retrieving, and presenting the content in a DL. One broad class relates to user interfaces that often adopt some spatial metaphor and connect some abstract spatial view with a concrete spatial instantiation (e.g., a window system). Another broad class relates to geographic information systems, often with time added as another dimension, supporting maps, timelines, global positioning system (GPS) data, mobile applications, studies of earth/atmosphere/astronomy, and combinations like mashups. A third broad class relates to multidimensional representations using various features that leads to particular implementations using vector or probability spaces, suitable for indexing, searching (especially approximate, nearest neighbor, cluster based, or range), retrieval, and ranking. An interesting example is an index that often is essential to ensure efficient operations of DLs; such an index generally combines streams of characters (for entries in the index), structures (to allow rapid response for key operations), and spaces (as in the vector space model).

18.2.1.3 Media

It is common for DLs to handle text documents with embedded images. But to address the broad needs of users around the world, DLs must support all types of content, including multimedia and hypermedia (multimedia with links), with sufficient quality of service, through speedy connections.

Multimedia has become an effective information sharing resource. DLs should provide direct support, including content-based searching, summarization, annotation, and specialized linking, as well as adequate storage (amounting to petabytes), for video, speech, audio, images, graphics, and animations. This is important, since web users expect services like those found with Flickr, Pandora, Shutterstock, YouTube, and Webmusic.IN. Sources of multimedia include libraries, museums, archives, universities, and government agencies, as well as individuals comfortable with uploading from cameras, smartphones, and camcorders.

Specialized DLs also support oral histories, recordings of dying languages (and translations), reuse of video segments, searching for similar images/videos, geo-coding, analysis of health-related images and data, support for bioinformatics, management of chemical information, access to tables and figures in documents, and preservation of virtual environments. Since formats keep changing in specialty areas, long-term preservation of such content is a particular challenge.

18.2.2 Collections

In a DL, a collection is generally an assembled, cataloged, and easily accessed digital information aggregation, whose resource data are usually provided through underlying database and storage structures (Unsworth 2000). Within a collection, storage preserves many kinds of digital information (such as documents and media files), which have been cataloged (see Section 18.2.5) and indexed for users' easy access. Thus, a collection is a set of digital objects, which may be atomic or complex, which may represent a variety of media and content types, and which should be accessible, such as by browsing (if in the form of a hyperbase) and searching (through an index and catalog). Within a collection, the format and class of digital objects generally is consistent in structure and representation. As an example, a book collection generally has associated descriptive information for each item in the collection, such as “title,” “publication information,” “material type,” “call no,” “location,” “status,” and “ISBN.”
Digital Libraries

A DL may be made up of one or several collections. Minimal support for DLs that manage multiple collections involves supporting each one independently. Integration challenges must be addressed if a unified view is to be afforded across the collections. This can be further complicated when the DL provides portal services while storing key information for digital objects that are distributed (Fox et al. 2010).

18.2.3 Linking and Linked Data

As can be seen in Figure 18.2, a DL includes a hypertext collection, or, more generally, if there is multimedia content, a hyperbase. This links objects in the collection together and facilitates browsing. Additional capabilities may be provided, depending on the nature of the hyperbase and on related services; we can understand this more clearly in the context of the World Wide Web (WWW).

The WWW is a huge source of data that is linked into what is called a web graph. The nodes in the graph are web pages, while the edges (arcs) are the hyperlinks between those pages. The WWW is considered a dynamically linked graph, where the graph is rapidly increasing in size and the links are continuously being created and changed. Hyperlinks connect web pages for purposes of navigation, connectivity, and reuse. The relationships between linked web pages are unspecified, and their nature can only be guessed by studying the anchoring texts.

Hyperbases can be much richer and supported by more powerful services. For example, links can be bidirectional, and stored in a database rather than in objects themselves. Furthermore, links can connect either sub-documents or documents, so the nature of each of the origin and destination can be anywhere in the range from highly specific to very general. When the hyperbase includes content fitting with the RDF (W3C 2004b), a node in the hyperbase can be as small as a string that names something, while when multimedia is included, a node can refer to a large video. Since the WWW uses general addresses, a node can even be a website or database.

With linked data, the granularity often focuses on the small side. In PlanetMath (Krowne 2005), there is automatic linking from many of the words in each encyclopedic entry, so the number of links is very high, and connectivity is strong (Krowne 2003). Linking web pages based on the semantic concepts and relations between them is considered in the Semantic Web (Berners-Lee et al. 2001) approach. Examples of works with extensive linking include Wikipedia, DBpedia, and ontologies.

18.2.4 Semantic Digital Library

Incorporating semantics (machine-oriented representations of meaning) in DLs can be done at different levels and using different formats. Semantics can help enhance the organization of data, aiding interoperability, facilitating integration of different collections, enhancing and enriching the services provided, and providing a domain model for each collection managed by the DL. In keeping with the recommendations of the World Wide Web Consortium (W3C) and fitting with plans for the Semantic Web, at least two types of representations should be supported. The first is RDF (W3C 2004a,b), typically managed with some type of triplestore, as is used in the Cyberinfrastructure for Network Science (CINET) project (see Section 18.4.1). The second involves ontologies, as represented by the web ontology language (OWL) (W3C 2004a) or other schemes. One of the goals of the Ensemble system (see Section 18.5.3) (Ensemble 2009) is to use ontologies to aid access by stakeholders. Thus, instructors in business, teaching about databases, can make use of connections in the ontology to identify database-oriented educational resources in an Ensemble collection developed by computer scientists.

18.2.5 Metadata and Catalogs

Metadata provides information about one or more aspects of the digital objects stored in the DL, such as: how the data are created, the purpose of the data, time and date of creation, creator or author of data,
location on a computer network where the data resides, and standards enforced on the data. Many current DLs at least include content for each of the 15 specified metadata elements that constitute the Dublin Core metadata standard, which was developed to enable easy description of web content (Dublin-Core-Community 2002).

Metadata are valuable for information retrieval. If digital objects themselves cannot be accessed and indexed, the metadata may be the only information usable to enable exploration. Even when full text is available, often the subject description and other elements in the metadata description can be used to enhance retrieval. Accordingly, the choice of metadata scheme can affect how well the system can satisfy the user’s information needs and greatly affects its long-term ability to maintain its digital objects.

Metadata records are stored in a catalog, so that every digital object has at least one metadata record in the catalog (or more if several metadata formats are supported). In some cases, the catalog is called a union catalog, when digital objects come from a variety of sources. A representative example is the union catalog of the NDLTD (2012).

### 18.2.6 Repositories

Repositories are a leading way of collecting, organizing, archiving, preserving, and providing access to intellectual properties. One important role is to have collections of metadata.

#### 18.2.6.1 Institutional Repository

Institutions producing scholarly work often collect, store, and provide access to intellectual properties generated within the institution. These repositories often allow universal access to resources including theses and dissertations, courseware, and technical reports. Services of these repositories include cataloging, organizing, indexing, browsing, and searching. One of the main goals of these repositories is to archive and preserve the contents while providing long-term access to the material.

A number of software packages are available to help institutions set up and maintain digital repositories. DSpace, introduced by the DSpace Foundation (DSpace 2008), is an open-source software package used by a number of institutional repositories, in part, since it is easy to install and deploy. Other systems include Fedora (Staples et al. 2003), ePrints (EPrints.org 2002), and various open-source content management systems (CMSs).

#### 18.2.6.2 Open Archives Initiative

Interoperability between different repositories is important for linking and cataloging resources across these repositories. Many DLs participate in the Open Archives Initiative (OAI) in order to ensure interoperability (Van de Sompel and Lagoze 2000). The OAI Protocol for Metadata Harvesting (PMH) is a standard for gathering metadata across DLs. Under OAI-PMH, actors fall into one of two roles, the data provider and the service provider. A data provider has a collection of metadata that it exposes through the protocol. A service provider then can provide any of a number of services (e.g., searching and browsing) from its local collection that is built through a series of requests for metadata (using PMH), sent to data providers. A number of open-source software packages such as jOAI (Weatherley 2012) and the collection workflow integration system (CWIS) (Internet Scout Project 2012) are available to allow repositories to expose their metadata following PMH.

The OAI also has aided interoperability through another initiative, OAI-ORE (Lagoze and Van de Sompel 2007), which focuses on object reuse and exchange. This operates in the context of the Semantic Web to enable interoperability involving groupings of information items (see Section 18.2.4). It addresses compound objects, such as those described using RDF, so those aggregations can be transferred between repositories, and used in each location.
18.2.7 Life Cycle

Content management involves more than the storage and retrieval of content. Policies and procedures are required to support content throughout its life cycle. A traditional life cycle describes stages like generation, management, use, change, and deletion (see Figure 18.3). Long-term aspects of content management include curation, archiving, and preservation. Curation involves deciding what content to maintain, under what conditions, and for how long. The following subsections discuss further the subsequent stages of archiving and presentation.

18.2.7.1 Archives

Archiving is the process of preparing, representing, and storing content to be accessed later. Key issues to consider include storage media, format representations, and access tools. A large number of physical and digital archives exist in a broad range of domains. Many countries have national archives, and there are special graduate programs for archivists.

18.2.7.2 Preservation

Preservation is the process of ensuring that the format, meaning, and accessibility of content are not lost over time. Various preservation operations may be performed on collections, for example, reformatting, copying, converting, migrating, and transforming. Physical media and content representations are changed periodically to keep up with the progression of technology. If the meaning and integrity

FIGURE 18.3 Life cycle quality on outer ring.
of content are not preserved, then content may be lost or may become inaccessible. There are a host of external threats to digital collections (Rosenthal et al. 2005). These include media failure, hardware failure, software failure, network problems, media and hardware obsolescence, software obsolescence, human error, natural disaster, attack by hackers, lack of funding to continue maintenance, and the demise of the hosting institution. Technology obsolescence complicates preservation services as computers and software are updated frequently and lead to the inability to access data. Technology fragility leads to inaccessible data when byte streams are changed or corrupted. Content should be examined periodically to mitigate some of these factors and ensure that bits have not changed, for example, using MD5 checksums. Duplication and replication techniques also are used to ensure an acceptable copy is available if a given copy is corrupted or lost, for example, using lots of copies keep stuff safe (LOCKSS) (Reich and Rosenthal 2001). Accessibility is provided through emulation, simulation, and translation from older software and operating systems.

The full information life cycle shown in Figure 18.3 includes the phases discussed earlier, along with others. For each phase, as can be seen in the outer ring, it is important to ensure that quality is maintained, so users interested in information at every stage have suitable services that address their information needs. This leads to a broader discussion of users, or, in 5S terms, societies, in the next section.

18.3 Societies

18.3.1 Users, Roles, Collaborators, and Communities

A DL serves a wide range of audiences including content authors, contributors, reviewer, editors, publishers, managers, content users, viewers, and readers. Individual users can be grouped into different categories based on their interest and use of the DL. Users with similar interests may form groups and communities within a DL. These groups can include reviewers, authors, editors, and members of organizations.

The need for access control on DL content can lead to systems supporting various roles for users. For example, while some users can be given the “Edit” privilege (i.e., an advanced role), other users may only have viewing privilege (i.e., a general role). A user may be assigned one or multiple roles based on factors such as ownership, activity level, and membership status.

User interactions such as tagging and ratings within a DL create a collaborative environment where community-added information enhances the value of existing content.

18.3.2 Social Networks

Social networks are becoming essential in different domains where collaboration is deemed important. Existing tools make it easier for DL users to share content on their preferred social networks (e.g., Twitter, G+, and Facebook). Systems used to manage DLs also are allowing users to create social networks within a DL. The existence of social networks can be leveraged to enhance the quality of services like rating and recommending (Ensemble 2009).

18.3.3 Intellectual Property

Societies follow a range of legal, social, economic, political, and other rules and regulations—these all connect to DLs in the 5S framework. One of the key areas associated with DLs that cuts across a number of these society-related issues deals with how content relates to its creators, disseminators, and users. Intellectual property (IP) concerns focus particularly on the origins of content, the ownership of economic interests, who has access (and under what conditions), and how all this is managed.

© 2014 by Taylor & Francis Group, LLC
18.3.3.1 Open Access

Some DLs provide access to their content online, free of charge, and free of most copyright and licensing restrictions. Such libraries may be called open-access DLs. These are common in the case of institutional repositories and of other sites that are part of the OAI. However, sometimes access must be controlled, as is explained in the following section.

18.3.3.2 Digital Right Management, Security, and Privacy

Security is important to consider when designing certain types of DLs. Security weaknesses in DLs can be exploited to access confidential information, result in loss of integrity of the data stored, or cause the whole DL system to become unavailable for use. This can have a damaging effect on the trust afforded to publishers or other content providers, can cause embarrassment or even economic loss to DL owners, and can lead to pain and suffering or other serious problems if urgently needed information is unavailable (Fox and ElSherbiny 2011).

There are many security requirements to consider because of the variety of different actors working with a DL. Each of these actors has different security needs, which are sometimes in conflict; this can make the security architecture of a DL more complex (Chowdhury and Chowdhury 2003). For example, a DL content provider might be concerned with protecting IP rights and the terms of use of content, while a DL user might be concerned with reliable access to content stored in the DL.

Some content stored in a DL may not be provided for free. Many content providers are concerned with protecting their IP rights. They may employ digital rights management (DRM), which refers to the protection of content from logical security attacks and the addressing of issues relating to IP rights and authenticity. DRM provides protection by encrypting content and associating it with a digital license (Tyrväinen 2005). The license identifies the users allowed to view the content, lists the content of the product, and states the rights the users have to the resource, in a computer readable format, using a digital rights expression language or extensible rights markup language that also describes constraints and conditions.

18.4 Systems

Building upon the discussion of formal approaches in Section 18.1.3, we can explain how systems fit with DLs. According to the DELOS reference model (Candela et al. 2011), this can be considered in levels (see Section 18.1.3.2). Further details are given in the following subsections, beginning with a broader discussion to provide context.

18.4.1 Cyberinfrastructure

Cyberinfrastructure (CI) refers to a broad area of software and systems deployment to aid societies with computing solutions including automation and provision of desired services (Atkins et al. 2003; NSF 2007). DLs generally fit into this either as standalone or embedded systems. A standalone DL can be utilized to support information satisfaction in a web portal, provide an underlying storage system, perform logical control and organizational functions, and conduct domain-specific tasks. The AlgoViz case study discussed in Section 18.5.2 explains such a standalone DL with its own portal and management system. On the other hand, a DL can operate at the middleware layer, embedded in a larger environment, for example, to provide a variety of data management and complex services to support scientific tasks. This makes sense, since eScience systems are complex and contain a variety of computing components. In general, CI often involves user interfaces, communication protocols, distributed high-performance computing resources, instrumentation, software applications, repositories, content collections, and multiple societies and user types. The CINET case study in Section 18.5.6 explains about a DL that is fully integrated within such a larger CI.
18.4.2 Architecture

In the software world, the structural aspects of a system are referred to as architecture, which in the case of DLs can be quite varied (Fox et al. 2005), including whether the system is a monolithic entity or an interconnection of components or whether it is centralized or distributed. Varying approaches can be employed, such as using agents (Birmingham 1995) or having a service-oriented architecture (Petinot et al. 2004).

DLs from multiple institutions and domains can be combined into a single federated system (Leazer et al. 2000), even when constituent DLs span multiple organizations with related but often conflicting purposes, goals, content, formats, collections, users, and requirements. As such, integration and interoperability is needed between federated DLs. These actually are easier if harvesting replaces federated search (Suleman and Fox 2003). This use of harvesting was one of the initial contributions of the OAI, discussed earlier in Section 18.2.6.2. Thus, harvesting is used in the National Science Digital Library (NSDL, see Section 18.5.4), which connects multiple DLs serving a variety of science-related domains.

18.4.3 Building DLs

In general, the different phases that go into building a DL are (Dawson 2004)

- Planning
- Requirements gathering and analysis
- Standards and policy making
- Collection development
- Architecture design
- User interfaces
- Metadata collection and enhancement
- Classification and indexing
- Implementation of services
- Access management
- Security/authenticity
- Preservation
- User studies
- Evaluation

Many DLs have been built from scratch, or from general components, as we did with the Computing and Information Technology Interactive Digital Educational Library (CITIDEL 2001), Multiple Access Retrieval of Information with ANnotations (MARIAN) (France et al. 2002), and Ensemble (2009). Such custom efforts are important in large-scale DLs catering to a very specific audience. More common, however, especially in recent years, is to make use of readily available software products, toolkits, and generic systems like Greenstone (Witten et al. 2000) for developing DLs. Many such solutions, often open-source, now exist for building DLs. The most widely used ones are presented in the following text.

The Greenstone DL software was born out of the New Zealand Digital Library Project at the University of Waikato and is one of the most extensively used open-source multilingual software packages for building DLs (Witten et al. 2001, 2009). It is a comprehensive system for constructing and presenting collections of millions of documents including text, images, audio, and video. The runtime system of the software has two parts: the receptionist, which takes requests from clients, and the collection server, which is the communication channel between the receptionist and the database. Collections can be built by specifying the Uniform Resource Locators (URLs) to be crawled. The user interface can be customized by the user.

Other widely used software for DL construction include Fedora Commons and DSpace, both of which are managed by a nonprofit organization called DuraSpace. DSpace is a “turnkey institutional
repository” that is used for creating open-access collections of all types of media. It is used most commonly in academic institutions. Fedora Commons is a software platform for managing, storing, and accessing digital content. Due to its modularity, it can easily be integrated with other database systems. DSpace, being an out-of-the-box solution, is easy to use in small repositories, while Fedora Commons is a more robust suite of services for an advanced, completely customizable repository (Bagdanov et al. 2009).

18.4.4 Services

DLs carry out many scenarios, and, thus, provide varied services to societies. We discuss this broadly in the 5S context in (Fox et al. 2012a), including providing a comprehensive taxonomy of services and describing which services we believe should be built into any system that is called a DL. The key minimal services are discussed later in Sections 18.4.4.1, 18.4.4.2.1, and 18.4.4.2.2; the following discussion also sets the context for understanding additional DL services. Some services, like indexing (see Section 18.4.4.1), relate to having a suitable DL infrastructure, while others address societal needs (see Section 18.4.4.2).

18.4.4.1 Indexing

DLs contain a vast amount of data and information. Indexing mechanisms allow users to search the information space efficiently. Most DLs offer search services based on text indexing, which parses the sentences to build an index of words. Search engine technologies such as Lucene are widely used to repeatedly index and search through text files. Apache Solr, an open-source search platform, uses Lucene as its search engine and adds additional features including hit highlighting, faceted search, and dynamic clustering. Indexing of multimedia content such as images and video also is incorporated in some DLs (Wei 2002; Orio 2008) to allow content-based information retrieval.

18.4.4.2 User Needs

Societies, in general, and users, in particular, have need for information. Accordingly, DLs provide a range of services to help users satisfy their needs; key among those are the ones discussed in the following subsections.

18.4.4.2.1 Search

Searching is one of the most common activities of DL users. DLs offer a capability to search content at varied levels of object granularity. Generally, two types of searching can be supported: full-text search and metadata search. The searching process consists of two steps, sometimes integrated: retrieving and ranking. Supported by the indexing service (see Section 18.4.4.1), the retrieving step accesses the indexing data structure using the query terms entered by the user. Documents that (approximately) match the query terms are retrieved and given as output to the ranking process. The purpose of the ranking process is to find the best order of the documents for the user.

Faceted search is a powerful feature that provides a combination of searching and browsing (see subsequent sections), where the searching process is constrained using one or more of the attributes of the objects.

18.4.4.2.2 Browse

A browsing service provides another means for users to explore information in DLs. Browsing is based on the logical or physical organization of objects in the DL. Often, a rich organization like a hyperbase (resulting from linking texts or multimedia, as in hypertext and hypermedia) supports browsing, but simpler structures, like lists of dates or author names, also are widely used.

Browsing and searching can be considered as dual operations (Fox et al. 2012a). One can have the same results of searching using browsing and also the same results of browsing using searching. In the first case, query terms are used to determine the nodes in the navigation tree (graph) that
match them. The navigation path can be built by traversing the navigation tree backward from the
matched nodes to the root of the tree. In the second case, one issues multiple queries that produce
the same results as those from following the navigational path during browsing. Each query corresponds to
one step in the navigational path that restricts more results than those of the previous query.

18.4.4.2.3 Visualize
Visualizing provides the DL users with the results of their requests, like searching and analyzing,
through an interactive representation. The representation of the results depends on the kind of the
operations producing the results, the context where the results are presented, and the type of data.
Search results can be represented as a list or as a grid of items (Fox et al. 1993). Analysis results can be
represented by graphs, charts, timelines, or maps.

18.4.4.2.4 Personalize
Personalization of information and services within a DL can be achieved in many different forms.
Services can be personalized to meet the specific needs of a group or community of users. Notification,
subscription, and tagging are some examples of personalized services for individual users. Along with
services, information can be personalized based on user profile, interest, and activity within the DL
(Neuhold et al. 2003), for example, in content recommender systems.

18.4.5 Integration
Within DLs, integration is required to bring people together and get coordinating processes to work.
Content, storage structures, file formats, metadata schemas, policies, and software are often diverse
within collaborations and federated systems. The integration of databases, content collections, and
metadata structures remains a difficult task. One approach, used in DLs, involves accessing diverse
underlying collections through Application Programming Interfaces (APIs) to conduct search over each
system. DLs also provide a means of implementing a standard set of services and user interfaces that
allow for integrated discovery and management in non-standardized collections. The goal of integra-
tion is to make the differences between systems, content, and services to appear seamless to end users,
despite differences and conflicts in underlying components.

18.4.6 Interoperability
DLs can be considered to be information management systems that deal with different kinds of digital
objects and provide different kinds of services that facilitate the manipulation of these objects and the
extraction of information from them. Thus, there are many collections of objects in either digital form
or real life that can be accessed from a DL through one interface. Accordingly, DLs should support
interoperability, to overcome the differences and difficulties of interfacing such sources of information.
Collections of differing objects vary regarding organization (structure) and metadata. These syntactic-level
differences sometimes aid when dealing with even more complicated semantic differences, which users
expect will be addressed in a comprehensive interoperability solution. Two such types of interoperability
that should be supported in DLs include data interoperability and services interoperability. Data interop-
erability focuses on providing access to different collections with different structures. One of the approaches
developed for supporting data interoperability is through OAI (see Section 18.2.6.2).

18.4.7 Quality
DLs manage different types of objects and support services for different types of users. Assessing, eval-
uating, and improving the performance of a DL often is based on identifying the different concepts
included in a DL and the dimensions of quality that are associated with these concepts. Table 18.1 shows
the main concepts that are needed to build a DL management system and what quality aspects we measure with respect to each concept. For example, for each digital object in the DL, we need to consider the relevance of the object to the user needs, whether the object can be preserved, how similar the object is to other objects, etc.

Information in DLs goes through several steps, starting from adding a raw digital object and ending by presenting the information to satisfy a user need, according to the “Information Life Cycle” (Borgman 1996). We can evaluate a DL by measuring the quality of information in each step of the information life cycle. Figure 18.3 shows the different steps included in the information life cycle in a DL, the services provided in each step, and the quality criteria that are used in evaluating those services.

Efficiency, effectiveness, completeness, relevance, compliance, timeliness, and similarity are examples of such evaluation criteria. For each criterion, one has to specify some measures (quality indicators). For example, efficiency of interactive services in a DL can be evaluated by measuring response time. Likewise, the relevance of the objects in the DL can be evaluated by measuring precision and recall.

## 18.5 Case Studies

There have been thousands of DLs used over the past decades, so understanding the DL field is aided by considering how the concepts explained earlier relate to a representative set of case studies.

### 18.5.1 ETANA

The ETANA-DL was built in support of Near Eastern archaeology (Shen et al. 2008). It involved collection of information from a number of archaeological sites, development of schema for each, mapping the schema into a global schema, harvesting metadata from sites using OAI-PMH, transforming that

<table>
<thead>
<tr>
<th>Digital Library Concept</th>
<th>Dimensions of Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digital object</td>
<td>Accessibility, Pertinence, Preservability, Relevance, Similarity, Significance, Timeliness</td>
</tr>
<tr>
<td>Metadata specification</td>
<td>Accuracy, Completeness, Conformance</td>
</tr>
<tr>
<td>Collection</td>
<td>Completeness, Impact factor</td>
</tr>
<tr>
<td>Catalog</td>
<td>Completeness, Consistency</td>
</tr>
<tr>
<td>Repository</td>
<td>Completeness, Consistency</td>
</tr>
<tr>
<td>Services</td>
<td>Composability, Efficiency, Effectiveness, Extensibility, Reusability, Reliability</td>
</tr>
</tbody>
</table>
metadata according to the schema mapping to be stored in a union metadata catalog, constructing an ontology for the domain with the aid of experts, and providing a broad range of services for archaeologists and others with interest. The browsing, searching, and visualizing were integrated, as was the whole system, based on the 5S framework, into a union DL.

### 18.5.2 AlgoViz

The AlgoViz project (AlgoViz 2011) is one of the primary content contributors to Ensemble. AlgoViz contains a catalog of visualizations that can be used to demonstrate various aspects of different algorithms. AlgoViz also has a bibliography collection related to algorithm visualizations (AVs) and hosts the field report collection—a forum like place where educators share their experience on using any particular AV. AlgoViz implements an OAI-PMH data provider service that maps catalog entry content to tags in the Dublin Core format. The metadata are thereby made available to Ensemble and other potential OAI service providers.

### 18.5.3 Ensemble

Ensemble (Ensemble 2009; Fox et al. 2010), the computing pathway project in the NSDL (see subsequent sections), is a distributed portal that collects and provides access to a wide range of computing education materials while preserving the original collections. It also hosts a community space to support groups of people with different interests related to computing education. Furthermore, there is a “Technologies” section that lists educational tools. Ensemble also shares key metadata with NSDL.

### 18.5.4 NSDL

The NSDL supports learning in Science, Technology, Engineering, and Mathematics (STEM) by providing access to a vast collection of related educational resources (NSDL 2001). Since 2000, the U.S. National Science Foundation has funded many groups in a variety of projects (e.g., Ensemble) to collect and manage quality metadata covering all key STEM areas (NSF 2012). NSDL uses harvesting and federated search, social networks, and a broad range of services, some centralized and some distributed, in one of the largest DL efforts yet undertaken.

### 18.5.5 CTRnet

The Crisis, Tragedy, and Recovery (CTR) network (CTRnet 2011) is a project aimed at building a human and digital network for collecting all information related to CTR events. A DL has been developed for managing the collected data and for providing services for communities that would help them in recovering from these events and for researchers to analyze data and extract useful information.

The CTRnet DL collects all types of data like reports, webpages, images, and videos. In collaboration with the Internet Archive (Internet_Archive 2000), many collections of websites have been prepared for further analysis and display to users. Integration with social media is also supported by collecting tweets related to CTR events. Tweet collections are analyzed for topic identification and opinion leader detection. The URLs in tweets are extracted, expanded, and stored for further use as seeds for archive building using local focused crawlers (around Heritrix) or the Internet Archive crawling service. The CTRnet DL provides communities and stakeholders with services like searching, browsing, visualizing, sharing, and analyzing data related to CTR events. Figure 18.4 shows key related data flows.

To aid in organizing the collected information, an ontology has been developed for the domain that includes all types of disasters and CTR events. The CTR ontology began as a merger of three disaster databases and is iteratively enhanced through the concepts extracted from collections archived and tweets collected using the yourTwapperKeeper (TwapperKeeper 2011) toolkit.
18.5.6 CINET

Educators teach about graphs and networks in multiple academic domains. In computer science, these topics are taught in courses on discrete structures. Graduate network science courses are based on this type of content, supporting education and research. These courses require real-world large graphs, scalable algorithms, and high-performance computing resources. CI including a portal has been developed to provide management for network science content and algorithms, including teaching advanced courses.

The CINET project provides educators, students, and researchers with access to a cloud computing cyber environment. This resource may be utilized to submit, retrieve, simulate, and visualize graph and network resources. It builds on our simulation-supporting digital library (SimDL) (Leidig 2012), which manages graphs, simulation datasets, large-scale analysis algorithms, results, and logging information.

Educators can use the DL as a repository for class projects, lecture materials, and research involving preexisting content and software. High-value educational content has been added to the repository; the DL includes the key datasets referred to in textbooks within this domain. Figure 18.5 details the system components in the CINET infrastructure. The DL provides core services to store and retrieve network graphs and algorithms. It also provides higher level services related to user interfaces, simulation applications, and computing resources.

This class of DLs provides services not commonly found in full-text collections. Thus, the DL automatically processes computer-generated content, extracts metadata from files, builds indexes of distributed scientific content, and provides search over highly structured numeric files. Infrastructure independent communication brokers are used to send and receive data in the DL and invoke DL services through automated workflow processes. This architectural design allows for generated scientific information to be captured, indexed, managed, curated, and accessed without human intervention. Management of scientific content in this manner provides a standardized management system for later efforts to support interoperability within a domain. Additional simulation-specific services aim to support advanced scientific workflow functions. These services include incentivizing researchers to contribute content, memorizing (i.e., caching) and reusing existing results, scientific workflow provenance.
tracking, curating of large-scale content, and scientific document searching. SimDL and its related services represent an emerging class of DLs applied in the sciences.

### 18.5.7 Publisher DLs

By the late 1980s, it was becoming clear that academic, professional, and commercial publishers were in the midst of an electronic revolution that would directly affect their activities (Fox 1988). This has led to the adoption of DL technologies to cover much of the publishing world. Automated systems are commonly used to receive submissions, handle reviewing, manage editorial work, collect final versions of documents, archive publications, and provide access (often through subscription or per-item payment schemes) (ACM 2000; IEEE-CS 2004).

### 18.5.8 WDL

The World Digital Library (WDL 2012) aims to promote international understanding and cross-cultural awareness by providing free multilingual access to online metadata and images of cultural riches of United Nations for Education, Science and Culture Organization (UNESCO) member states and others that have become partners. The types of items include books, journals, manuscripts, maps, motion pictures, newspapers, prints/photographs, and sound recordings.

In June 2005, Dr. James H. Billington, U.S. Librarian of Congress, proposed the establishment of the WDL to UNESCO. In December 2006, key stakeholders from around the world attended an Experts Meeting sponsored by UNESCO and the Library of Congress, deciding to establish working groups focused on standards and content selection. By October 2007, a prototype of WDL was presented at the UNESCO General Conference. In April 2009, WDL was officially launched and opened to the public.

WDL has an emphasis on quality, with quantity also desired but not at the expense of quality. Regarding the development of WDL, some of the key features and aspects are as follows:

1. Metadata: Each item in WDL is described, indicating its significance as well as what it is. Curators and experts provide context to stimulate the curiosity of students and public interest in cultural heritage. There is consistent coverage of topical (based on the Dewey Decimal System), geographical, and temporal information.

**FIGURE 18.5** CINET system components.
2. Languages: Access to metadata, navigational aids, and supporting content like curator videos is enabled for seven languages: Arabic, Chinese, English, French, Portuguese, Russian, and Spanish.

3. Collaborative networks: Technical and programmatic networks connect partners, stakeholders, and users—all in a spirit of openness. There are contributions, including from libraries and archives, of: content collections, technology, finances, and involvement in working groups.

4. Digitization: The Library of Congress and partners in Brazil, Egypt, Iraq, and Russia have digital conversion centers to produce high-quality digital images, as can be found in the WDL. WDL intends to work with UNESCO to encourage the establishment of additional digital conversion centers around the world.

18.6 Future

The DL field—and its “children” like repositories, CMSs, and web portals—will continue to have broad impact as people around the globe utilize specialized systems to work with particular types of content, providing tailored assistance that goes beyond the general services of search engines. Supporting research and development efforts will continue, sometimes referred to using “DLs,” but also carried out by those whose first loyalty is database processing, information retrieval, knowledge management, human–computer interaction, hypertext, electronic publishing, or other related domains. DLs are the result of synergies informed by all these fields, so conceptual and practical integration and interoperability is perhaps the most important type of DL research required. Establishing a firm formal foundation, supporting transparency across data/information/knowledge collections, developing consensus regarding quality measures and indicators, and shifting toward modular Semantic Web–oriented implementations atop a service-oriented architecture, are among the key challenges now faced. Funding and collaboration are required to address these and other problems explained earlier, so users working in a particular domain can benefit from improved services (e.g., support for collection security, integrated text and data analysis workflows, and enhanced information visualization). The discussion earlier should help you understand this important area, so you can more effectively use DLs and/or can assist in their further unfolding.

Acknowledgment

Some of this material is based upon work supported by the National Science Foundation (NSF) under Grant Nos. CCF-0722259, DUE-9752190, DUE-9752408, DUE-0121679, DUE-0121741, DUE-0136690, DUE-0333531, DUE-0333601, DUE-0435059, DUE-0532825, DUE-0840719, IIS-9905026, IIS-9986089, IIS-0002935, IIS-0080748, IIS-0086227, IIS-0090153, IIS-0122201, IIS-0307867, IIS-0325579, IIS-0535057, IIS-0736055, IIS-0910183, IIS-0916733, ITR-0325579, OCI-0904844, OCI-1032677, and SES-0729441. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

This work also has been partially supported by NIH MIDAS project 2U01GM070694-7, DTRA CNIMS Grant HDTRA1-07-C-0113, and R&D Grant HDTRA1-0901-0017.

References


© 2014 by Taylor & Francis Group, LLC


