Identifiers and Repositories

hussein suleman

uct cs honours 2005

Features of (Good) Identifiers

- Must uniquely name digital objects and/or metadata.
  - e.g., ISBN numbers
- Must be unique within global/local domain.
  - e.g., Email addresses
- May have scheme associated to indicate source or naming convention.
  - e.g., webpage URLs – http://something …

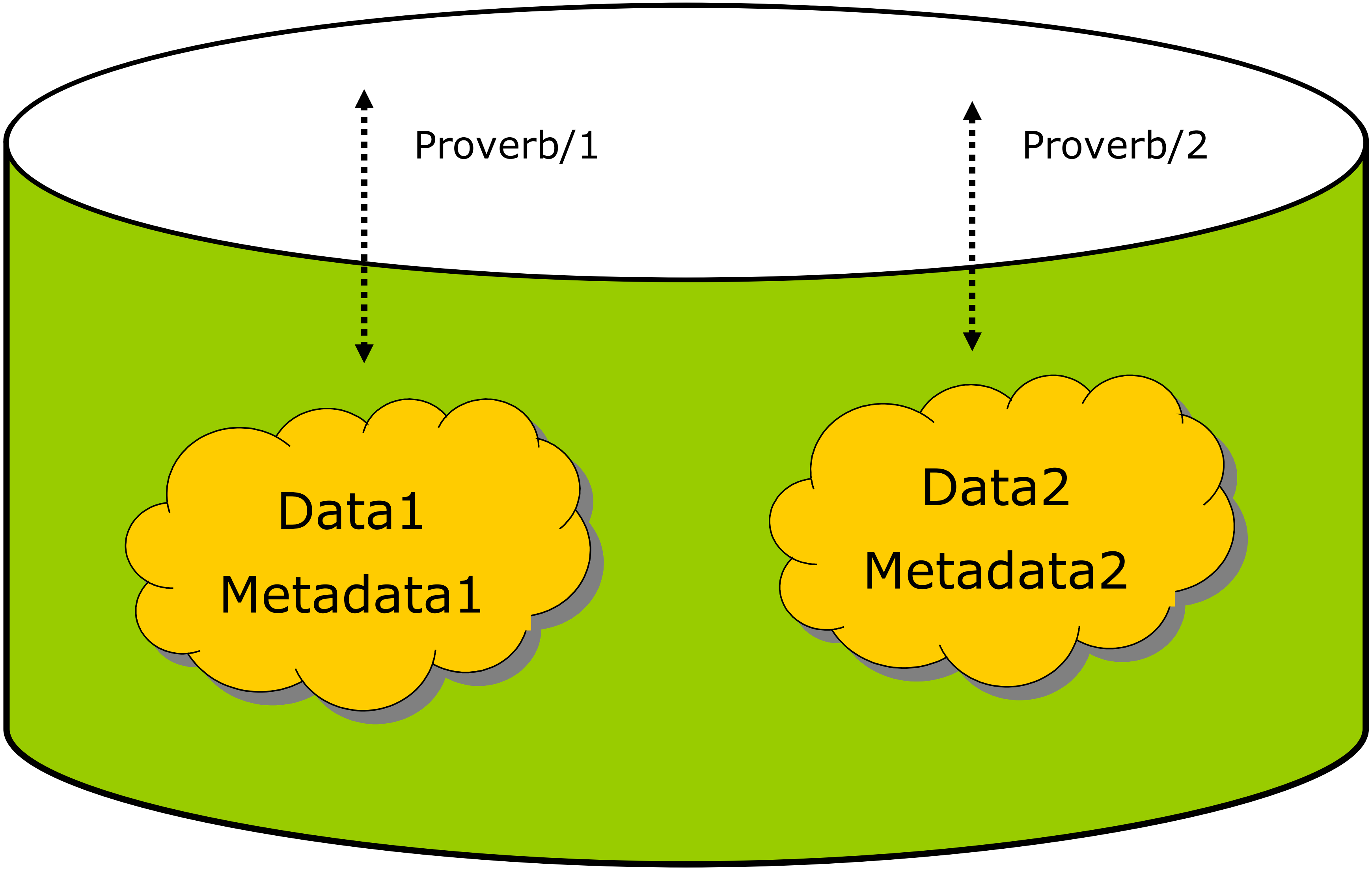
URLs, URNs and URIs

- Universal Resource Locator (URL) = location-specific e.g.,
  - `http://www.husseinsspace.com/pictures/200404napp/04napp9.jpg`
- Universal Resource Name (URN) = location-independent e.g.,
  - `http://purl.org/net/hussein/pictures/200404napp/04napp9.jpg`
- Universal Resource Identifier (URI) = generic identifier e.g.,
  - `oai:hspics:200404napp9`

Example Repository 1

Identifier	Metadata	Data
Proverb/1	Source: Africa Type: Proverb	When an old man dies, a library burns down
Proverb/2	Author: Albert Einstein Type: Quotation	Imagination is more important than knowledge

Conceptual Model 1

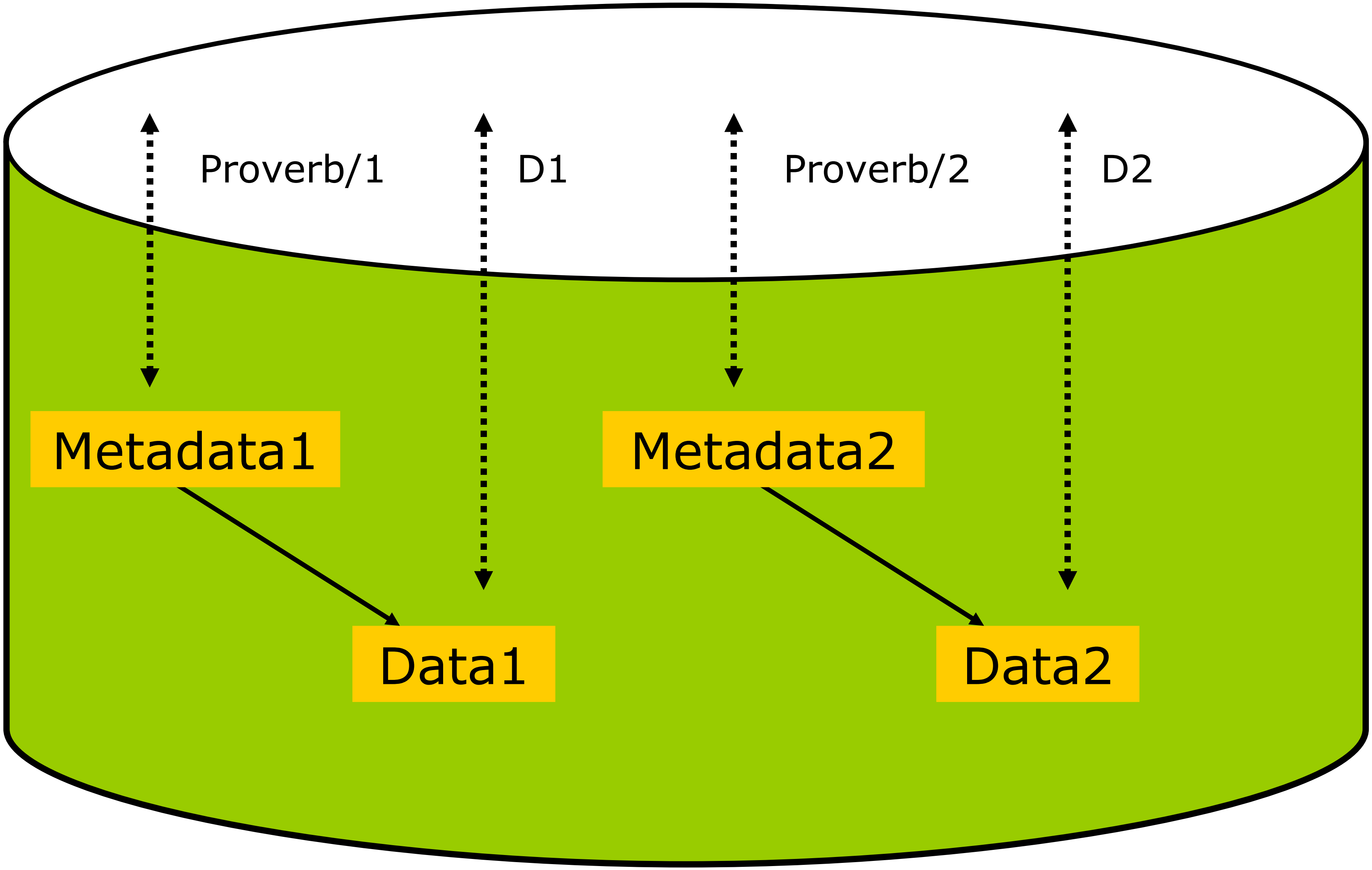


Example Repository 2

Identifier	Metadata
Proverb/1	Source: Africa Type: Proverb Identifier: D1
Proverb/2	Author: Albert Einstein Type: Quotation Identifier: D2

Identifier	Data
D1	When an old man dies, a library burns down
D2	Imagination is more important than knowledge

Conceptual Model 2



Proverb/1

D1

Proverb/2

D2

Metadata1

Data1

Metadata2

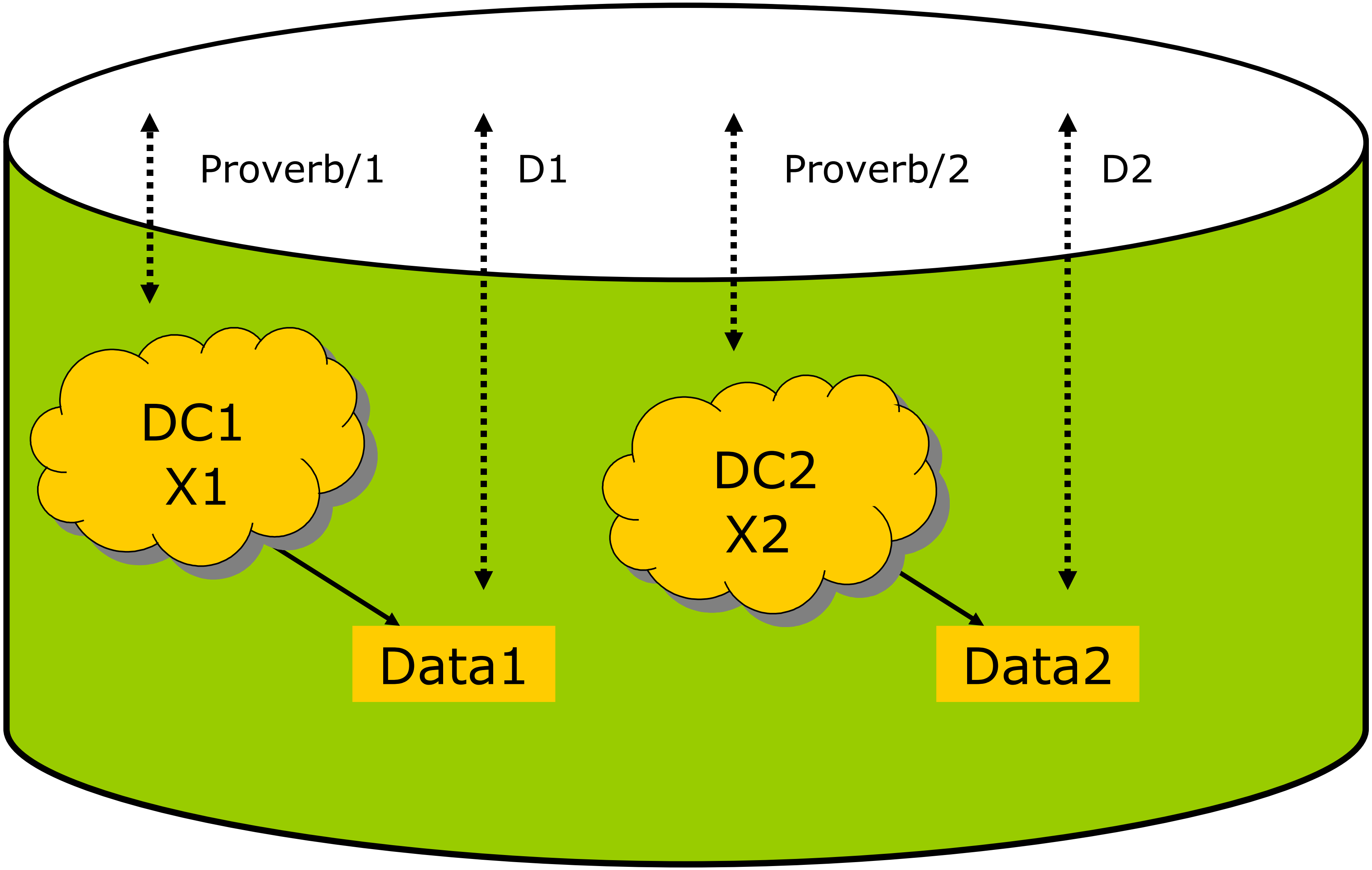
Data2

Example Repository 3

Identifier	MetadataX	DC
Proverb/1	Id: P1 Data: D1	Source: Africa Type: Proverb Identifier: D1
Proverb/2	Id: P2 Data: D2	Author: Albert Einstein Type: Quotation Identifier: D2

Identifier	Data
D1	When an old man dies, a library burns down
D2	Imagination is more important than knowledge

Conceptual Model 3



Repository Storage Models

- Generalisation of database.
- Collection of metadata records.
  - in XML or other flat files
  - in database BLOBs
  - in columns of database tables
  - embedded in digital objects
- Abstract interface to data collection.
  - no concept of how (meta)data is stored.

Repository Access Protocol (RAP)

- A repository can be defined as a network-accessible server.
- RAP specifies a simple interface to access and manage digital objects in a repository.
- RAP is an abstract model, with concrete implementations in the Dienst, OpenDLib, OAI and ODL projects.
- This is usually referred to as the "Kahn/Wilensky architecture".
    - does Kahn ring any bells?

RAP Operations

- ACCESS_DO
  - Return a manifestation (dissemination) of a digital object based on its identifier and a specification of what service is being requested.
- DEPOSIT_DO
  - Submit a digital object to the repository, assigning or specifying an identifier for it.
- ACCESS_REF
  - List services and their access mechanisms for the repository.

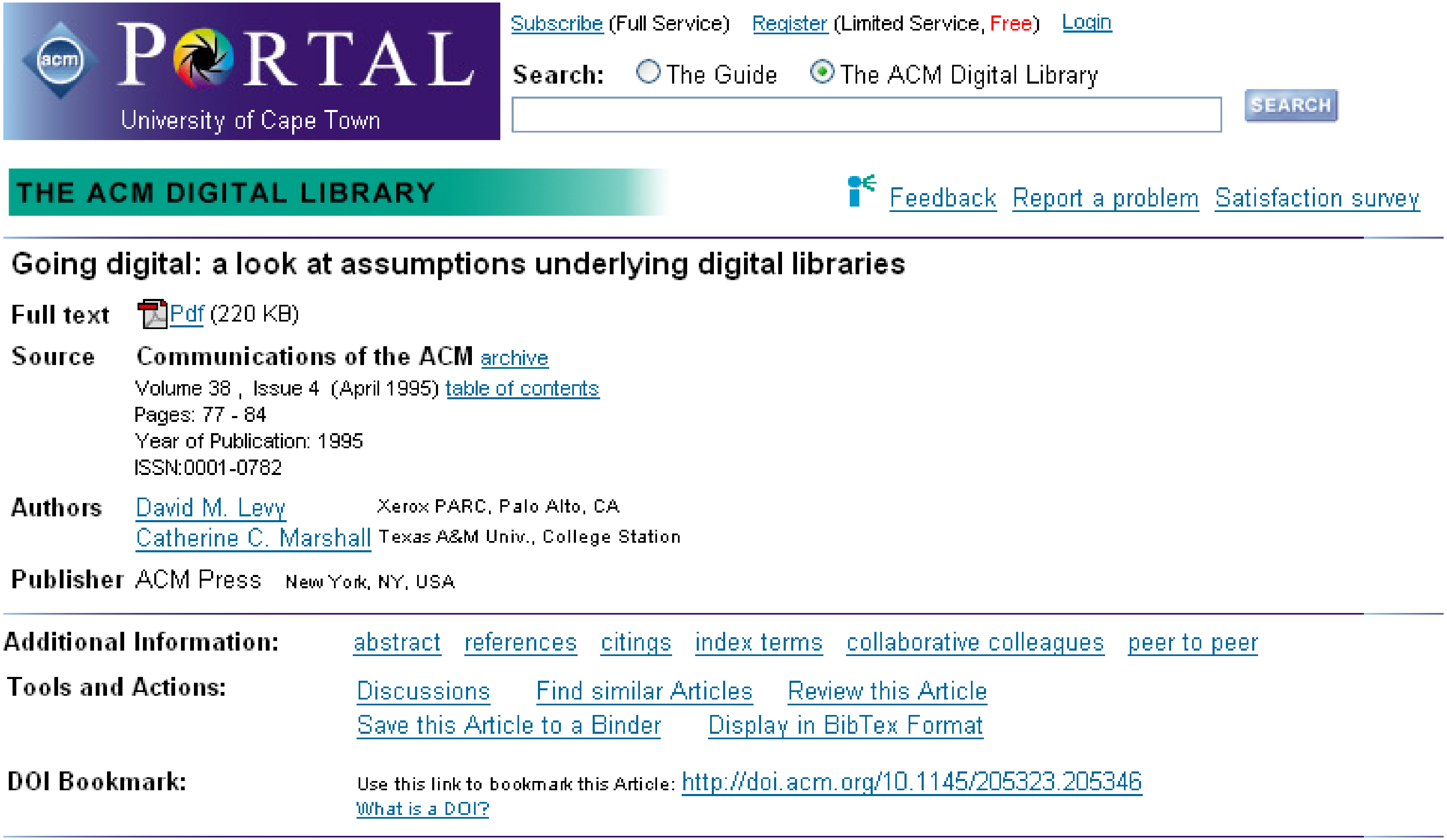
RAP: Naming of Digital Objects

- Each digital object must have a location-independent name (handle), made up of a repository identifier and a local name.
  - Example:
    - berkeley.cs/csd-93-712
    - where berkeley.cs is the repository and csd-93-712 refers to a technical report.
- Handles are resolved by a handle server to redirect a service provider to a repository containing an object identified only by its location-independent handle.

Handle Servers

- A handle server stores the association between handles and physical locations of objects.
- Handle servers follow a DNS model:
  - they are distributed and replicated
  - there are global and local servers
  - handles may be cached locally after being resolved to minimise resolution traffic
  - management of servers/handles requires an authority system for management, accountability, delegation, etc.

Handle Example



Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ○ The Guide ● The ACM Digital Library

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

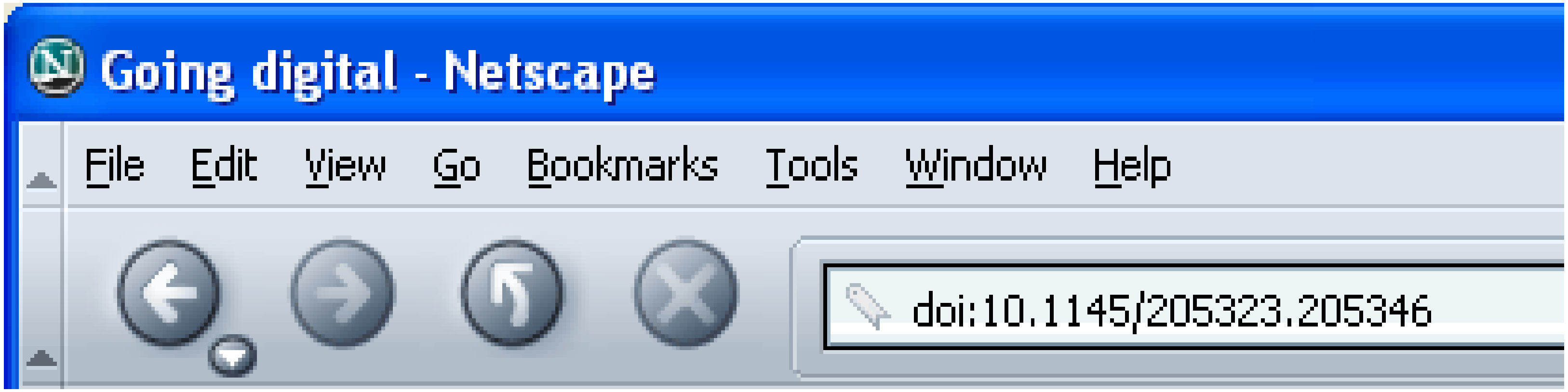
Going digital: a look at assumptions underlying digital libraries

Full text	Pdf (220 KB)
Source	**Communications of the ACM** archive Volume 38 , Issue 4 (April 1995) table of contents Pages: 77 - 84 Year of Publication: 1995 ISSN:0001-0782
Authors	David M. Levy Xerox PARC, Palo Alto, CA Catherine C. Marshall Texas A&M Univ., College Station
Publisher	ACM Press New York, NY, USA

Additional Information:	abstract references citings index terms collaborative colleagues peer to peer
Tools and Actions:	Discussions Find similar Articles Review this Article Save this Article to a Binder Display in BibTex Format
DOI Bookmark:	Use this link to bookmark this Article: http://doi.acm.org/10.1145/205323.205346 What is a DOI?

Digital Object Identifiers (DOIs)

- DOIs are a standardised implementation of the handle concept.
- Handles/DOIs are URIs that refer to digital objects while URLs are URIs that refer to network services.
- Handle/DOI resolution can be performed transparently using a browser plug-in.



doi:10.1145/205323.205346

Other repository models

- FEDORA (Flexible Extensible Digital Object and Repository Architecture) defines a generic interface to manage digital objects at a lower layer in an information system.
  - see http://www.fedora.info/
- SODA (Smart Objects Dumb Archive) packages digital objects into buckets containing the data along with the code to mediate access, display the objects, enforce rights, etc.

References

- Kahn, Robert and Robert Wilensky (1995) “A Framework for Distributed Digital Object Services”, CNRI. Available http://www.cnri.reston.va.us/home/cstr/ar

- Maly, Kurt, Michael L. Nelson and Mohammed Zubair (1999) “Smart Objects, Dumb Archives: A User-Centric, Layered Digital Library Framework”, in D-Lib Magazine, Vol. 5, No. 3, March 1999. Available http://www.dlib.org/dlib/march99/maly/03