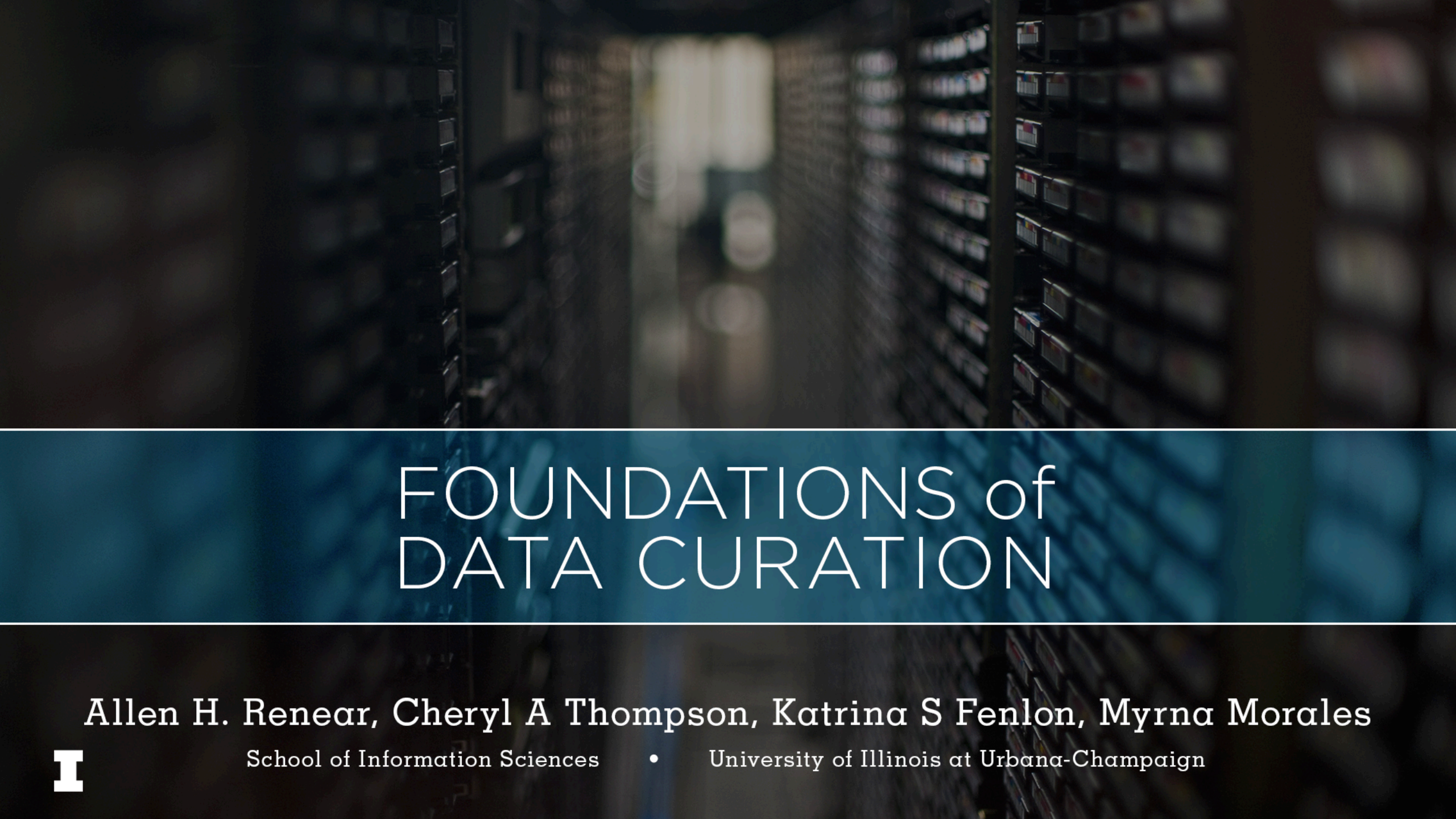
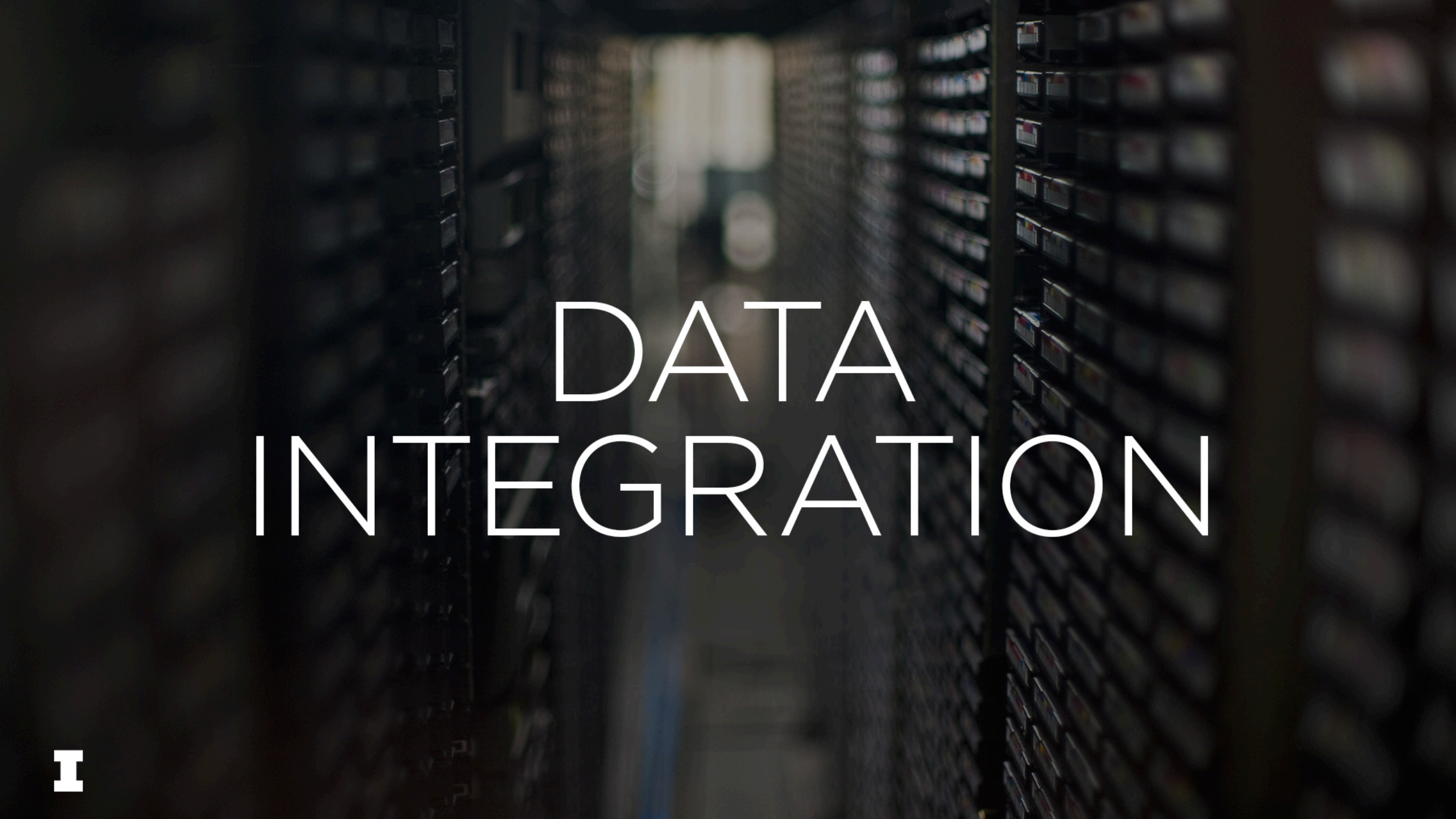


FOUNDATIONS of DATA CURATION

Allen H. Renear, Cheryl A Thompson, Katrina S Fenlon, Myrna Morales

School of Information Sciences • University of Illinois at Urbana-Champaign


DATA
INTEGRATION

2

MANAGING HETEROGENEITY

Managing Heterogeneity

Managing encoding heterogeneity
Managing syntax heterogeneity
Managing model heterogeneity
Managing representational heterogeneity
Managing semantic heterogeneity
Managing processing heterogeneity
Managing policy heterogeneity

Data integration:

> ". . . combining data residing in difference sources and providing users with a unified view. . . [Lenzerini 2002]

Kinds of heterogeneity (again)

*Relatively easy
**Often difficult
***Usually very difficult

Encoding heterogeneity *

Different mappings from bitstreams into bytes, characters, numbers, or other logical units

Syntax heterogeneity *

Different data description languages for the same model type: e.g. RDF/XML vs N3

Model heterogeneity **

Different model type; e.g., relations vs entities/relationships

Representational heterogeneity **

Different modeling choices within a model type; e.g. relationships vs entities.

Semantic heterogeneity ***

Different conceptualization of similar domain features

Processing heterogeneity **

e.g. different maintenance and update regimes

Policy heterogeneity **

e.g. different privacy and security rules, varying ownership and licensing, etc.

Adapted from Bertram Ludäscher, Kai Lin, Shawn Bowers, Efrat Jaeger-Frank, Boyan Brodaric, Chaitan Baru, "Managing scientific data: From data integration to scientific workflows", *Geoinformatics: Data to Knowledge, 2006*; and Amit Sheth. "Changing Focus on Interoperability in Information Systems: From System, Syntax, Structure to Semantics". In M. Goodchild, M. Egenhofer, R. Fegeas,and C. Kottman, editors, *Interoperating Geographic Information Systems* Kluwer, 1998.

Managing encoding heterogeneity

Variations in character encodings (or other logical atoms), byte interpretation, bit stream management, magnetic tape formatting etc. have created many problems.

- How many bits in the octet are coding? Is the byte to be read left to right or right to left? Are there control bytes in the bit stream? What are line end and carriage return characters?
- If sets of logical atoms are not identical there is no complete 1:1 conversion and a variety of problematic resolutions -- such as conflation, unification, omission, and named references -- must be considered:

  And if the nature of the logical atoms may be unclear is that (“ü”) a diaeresis or umlaut?

But thanks to common tools and conventions (particularly Unicode) encoding is less of an issue now.

To be sure the mapping from bit streams to characters is extremely complex.

See: *Unicode Character Encoding Model*, Ken Whistler & Mark Davis (Unicode Technical Report #17)

But global implementation of the Unicode standard is robust and complete.

Syntax Heterogeneity

Converting from one syntax to another (within the same general data model) is a very common data curation activity

- For instance, from SGML to XML or JSON or from RDF/XML to N3 or turtle.

Schemas also may need to be converted from one schema language to another

- For instance from XML/DTD schema language to (W3C) XML/Schema or XML/RelaxNG.

Syntax conversions can be challenging, but often can be accomplished without loss of information, and for common conversions there exist effective tools.

There can be problems though, particularly at the schema level. For example,

- XML/Schema has more data typing options than XML
- SGML/DTDs have grammar constraints difficult to implement in XML/DTDs

Model Heterogeneity

The same information can be expressed using completely different *types* of data models (e.g. relations vs trees, vs ontologies).

Conversion from one data model to another is often needed, for data integration, preservation, exchange, to use particular tools and applications.

Conversions such as these are particularly important in data curation

- Exporting XML files from a relational data base
- Using a relational database to store XML documents
- Converting relations to RDF graphs, and RDF graphs to relations

Methods for these conversions are exist, but they can be challenging to implement and may not provide the same functionality, data typing, or constraints

In addition they may not be “natural”: imagine storing an XHTML web page as relations!!

Most importantly: a new schema, for an equivalent but different data model, will need to be developed.

Representational Heterogeneity

Here are two classic problems for representational heterogeneity:

For XML: element or attribute?

`<chapter>` or **`<div type=chapter>`** ?

For ontologies: entity or relationship?

The pros and cons of expressively equivalent modeling choices can be difficult to evaluate.

Semantic Heterogeneity

This is the most challenging heterogeneity. We take in up in the next video.

Processing Heterogeneity

Some examples

Modification differences (instance level)

- One dataset is updated weekly, another monthly, another when observations change
  Related: lack of atomicity
- One dataset is undergoes integrity and validation weekly, another after every update
- One dataset documents provenance of modifications, another does not

These are particularly problems for federated datasets, but also for derived combined datasets, especially when they are routinely and perhaps automatically derived (ETL).

Modeling differences

Datasets may undergo schema changes without notification or coordination.

Metadata differences

Datasets may undergo schema changes without notification or coordination.

Policy Heterogeneity

Information may be subject to different restrictions related to ownership, privacy, libel, security, disclosure, etc.

- As a dataset may have originated in one country, be about the citizens of another, be owned by a third, be stored in a fourth, and be accessed by users in many others legal and regulatory requirements may be complex, even inconsistent.

  And another to be integrated with the first, may have…etc.

- **Data derived by analysis** from input data from multiple sources adds additional complexity.

(Imagine: a deduction from data from multiple sources)

FOUNDATIONS OF DATA CURATION (IS531)

Allen H. Renear, Cheryl A Thompson, Katrina S Fenlon, Myrna Morales
School of Information Sciences
University of Illinois at Urbana-Champaign

Includes material adapted from work by Carole Palmer, Melissa Cragin, David Dubin, Karen Wickett, Bertram Ludaescher, Ruth Duerr and Simone Sacchi.

Comments and corrections to: renear@illinois.edu.