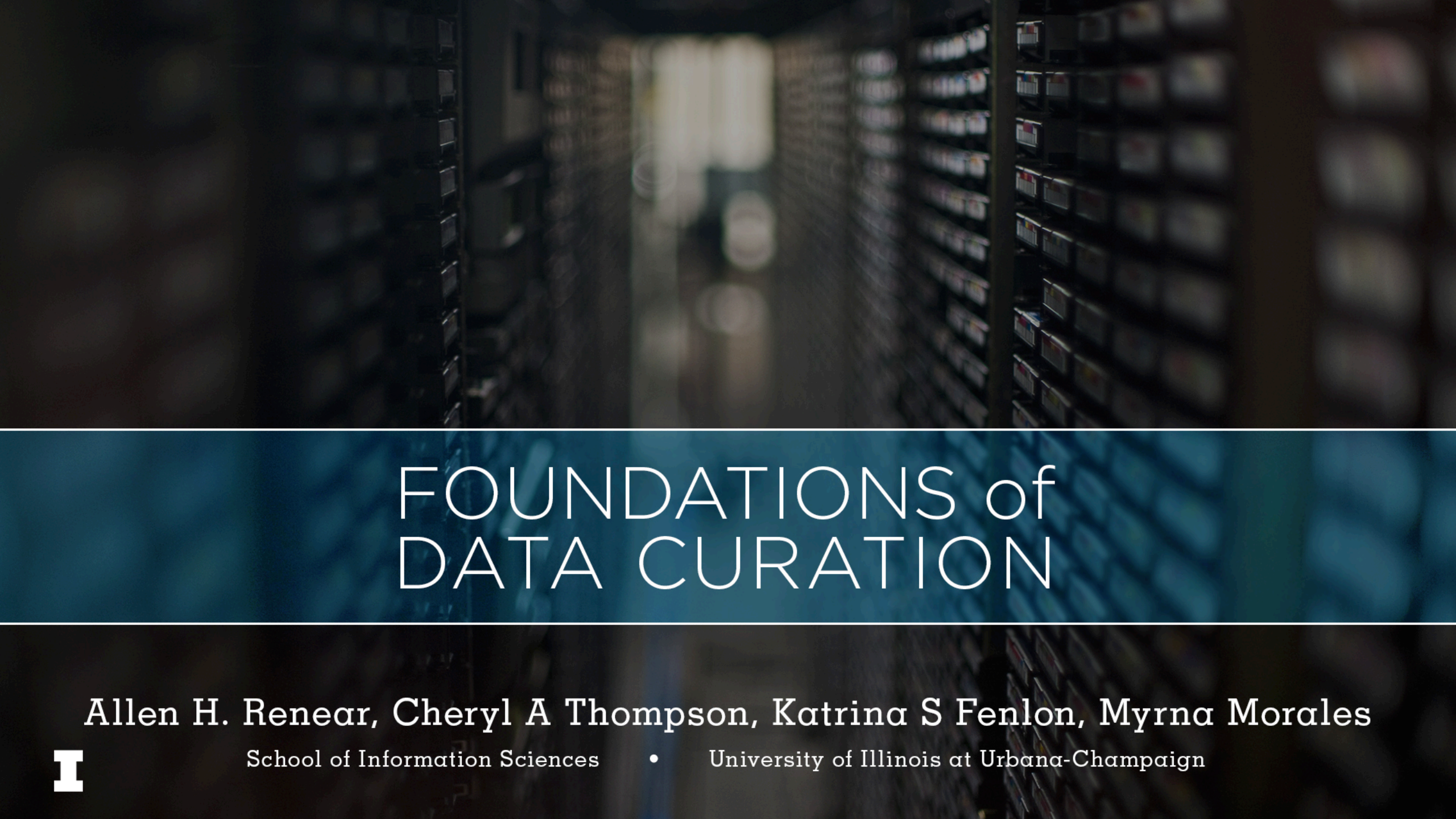
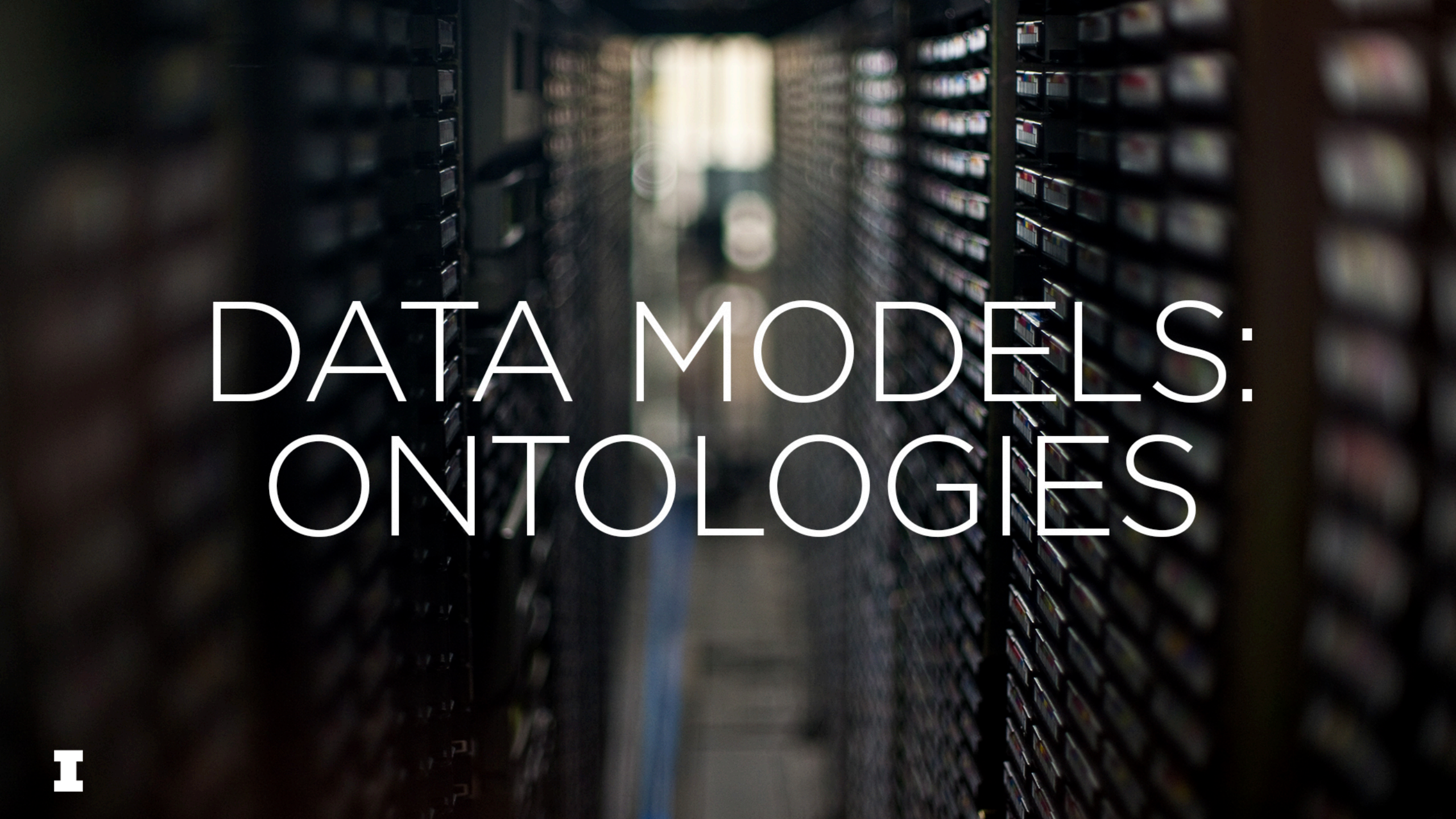


FOUNDATIONS of DATA CURATION

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

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


DATA MODELS:
ONTOLOGIES

1

THE PROBLEM:
CONNECTING DATA TO INFORMATION

The Problem: connecting data to *information*

Tables and trees abstract away from storage and processing
and let us focus on the data itself

But just as there are many ways to store the same data,
there are also many ways to use tables and trees to *express the same information*

This results problems similar but different than we saw with tables and trees

Setting up the problem

Imagine that you you need to manage some data.

You might use XML trees, or you might use relations, or you might use RDF triples.

Or you might start with XML trees, then in a few years use relations, and then later use RDF triples.

But all through these changes your domain of interest remains pretty much the same:
the same kinds of things, the same kinds of properties, the same kinds of relationships.

Moreover schemas for a domain are frequently revised even within a single type of logical data model

But again, even as schema details and structures are updated, the domain remains the same

So: different data model types and different schemas within a data model type

But through all this just one domain of kinds of objects, relationships, properties

The Problem

So, different data models types and different schemas within a data model

But just one domain of kinds of objects, relationships, properties

So obviously our understanding of that domain is of course extremely important.

In a single domain that understanding:

Guides initial schema development

Is constant across schema variations and revisions (within a model type)

Is constant across schemas from different model types

But where is this understanding of the domain articulated? And how is it connected to these logical schemas?

Too often only in the memory of programmers and other staff.

Even when prose documentation of the domain features exists that documentation rarely provides the formal precision and completeness needed to be useful; nor the mathematical constructs needed for computer processing.

How to solve this problem

To address this problem

we need an independent neutral way to describe the domain of interest.

This description cannot be tied to any particular logical model
Or to implementation variations of different specific schemas within a logical model.

It must in fact abstract away from logical models and schema variations
(just as those logical models themselves abstracted away from storage methods).

This description should be similarly precise, complete, and mathematical.

And it must be possible to map these descriptions to schemas at the logical level.

You look skeptical

It is natural to feel that this is going too far

After all, when we look at a schema, we feel that in fact it is describing the domain

But this is because of all the information we bring to bear in interpreting that schema

And since we do this so naturally and routinely we tend to forget just how technical and thin the formal theory of the relation model is

Let’s look at the issue from a different perspective:

What we think we see . . .
vs what is really there

Take a look at this relation; what do you see? What is it saying to you?

WorkID	Author	Title	FirstPublished
W58425	M42425	Moby Dick	1851
W85246	H24246	The Scarlet Letter	1850
W55427	H24246	Fanshawe	1828

What we intend to do here is record here, and what you see, *information*, propositions we believe to be true.

e.g., (1) *W58425, a work, is titled "Moby Dick" and was published in 1851.*

But given the formalities of the relational model the relation's explicit semantics produces, at best, this proposition:

(2) *There is a tuple where the primary key **WorkID** has the value'W5825' and the **title** attribute has as its value the string "Moby Dick."*

The Missing Link: *Information*

WorkID	Author	Title	First Published
W58425	M42425	Moby Dick	1851
W85246	H24246	The Scarlet Letter	1850
W55427	H24246	Fanshawe	1828

Yes, *you* can indeed infer from that relation that

(1) *W58425, a work, is titled "Moby Dick" and was published in 1851.*

But that is an *inference* humans make.

We can make such inferences from the fact that in a particular tuple certain attributes have certain values.

But the relational model does not support this inference.

And we make the inference using background information: English meanings of column headings and values*, knowledge of the domain, common sense, conversations with the DBA, bits of ancient documentation, etc.

This makes the *meaning*, the *semantics* of the relation unreliable and inaccessible to computer processing.

*Note that in the formal relational model the column headings are not attributes in the ordinary sense (properties), they are domain names, strings mapped to sets of allowed values. And there is no sense, in pure relational theory, in which the value of an attribute/value pair is *asserted of*, *said about*, or *describes* the thing identified by the corresponding primary key. That is all added interpretation, even when intended by the database designer. And in many cases that particular interpretation would be false or senseless (e.g. tables for M:M relationships).

The heart of the problem

We lack a shared framework
that could explicitly and formally map the relevant features in the domain of interest
to the relation or tree schemas for data about those things

Such a framework could be used to guide relation and tree schema development and revision
and to identify the common domain features reflected in different relation and tree schemas.

The framework would also provide relation and tree schemas with a *semantics*
and as a consequence their instances, relations and trees, would have meaning and assert propositions

Without that we really don't know, formally, what a relation (or an XML document) is telling us

Yes, it's in our head, but that not good enough

We need yet *another level of abstraction*

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.