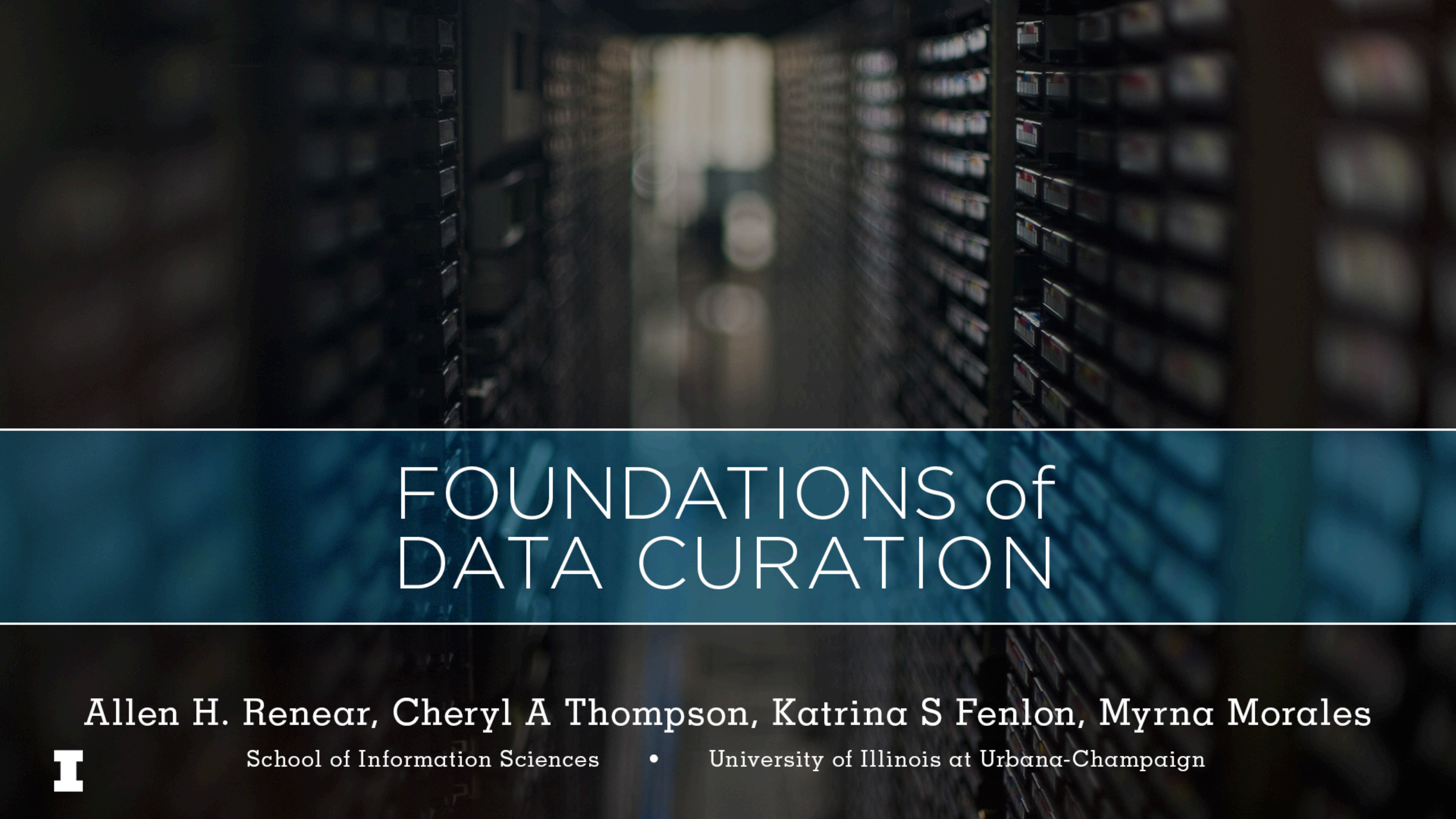
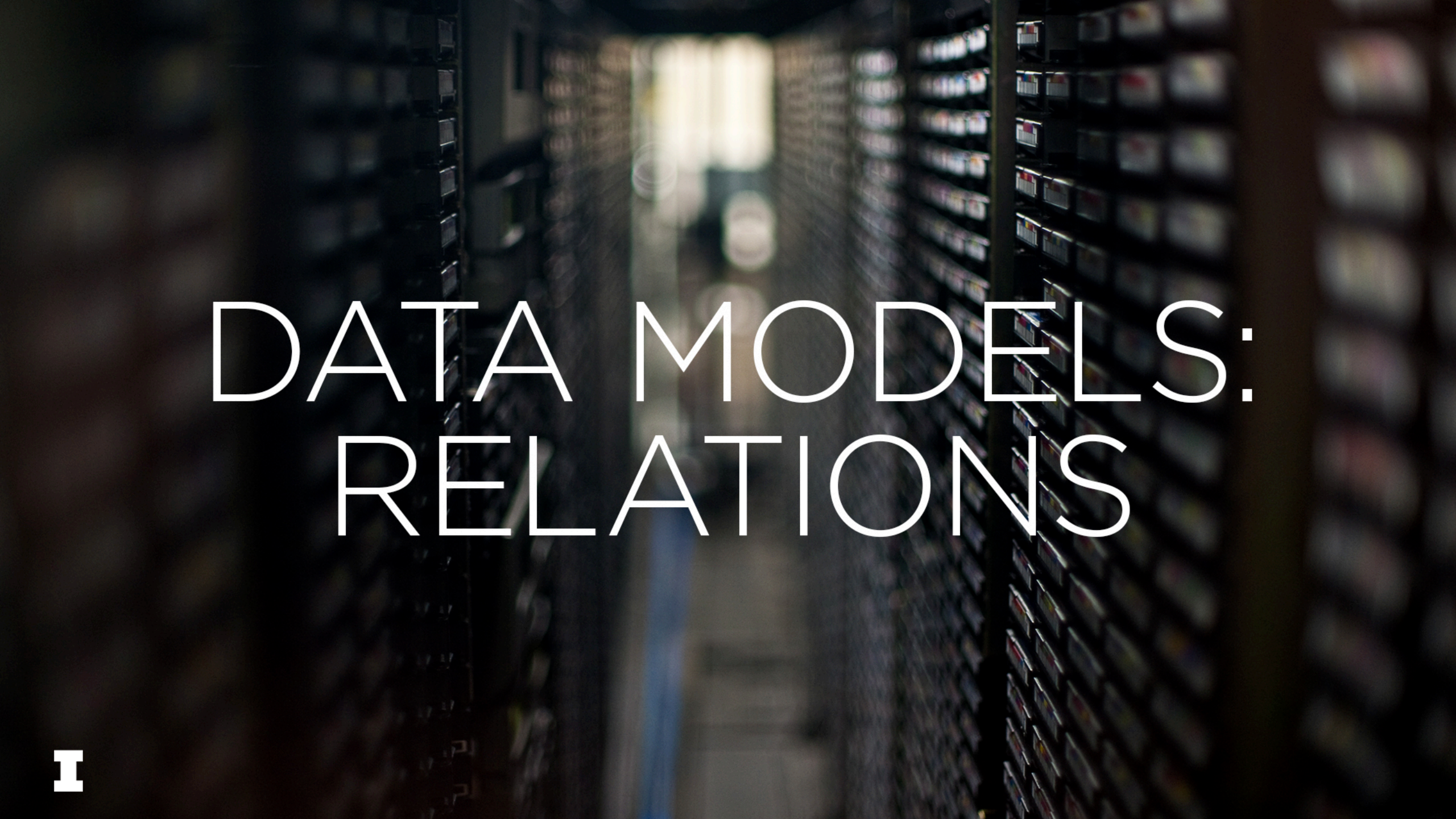


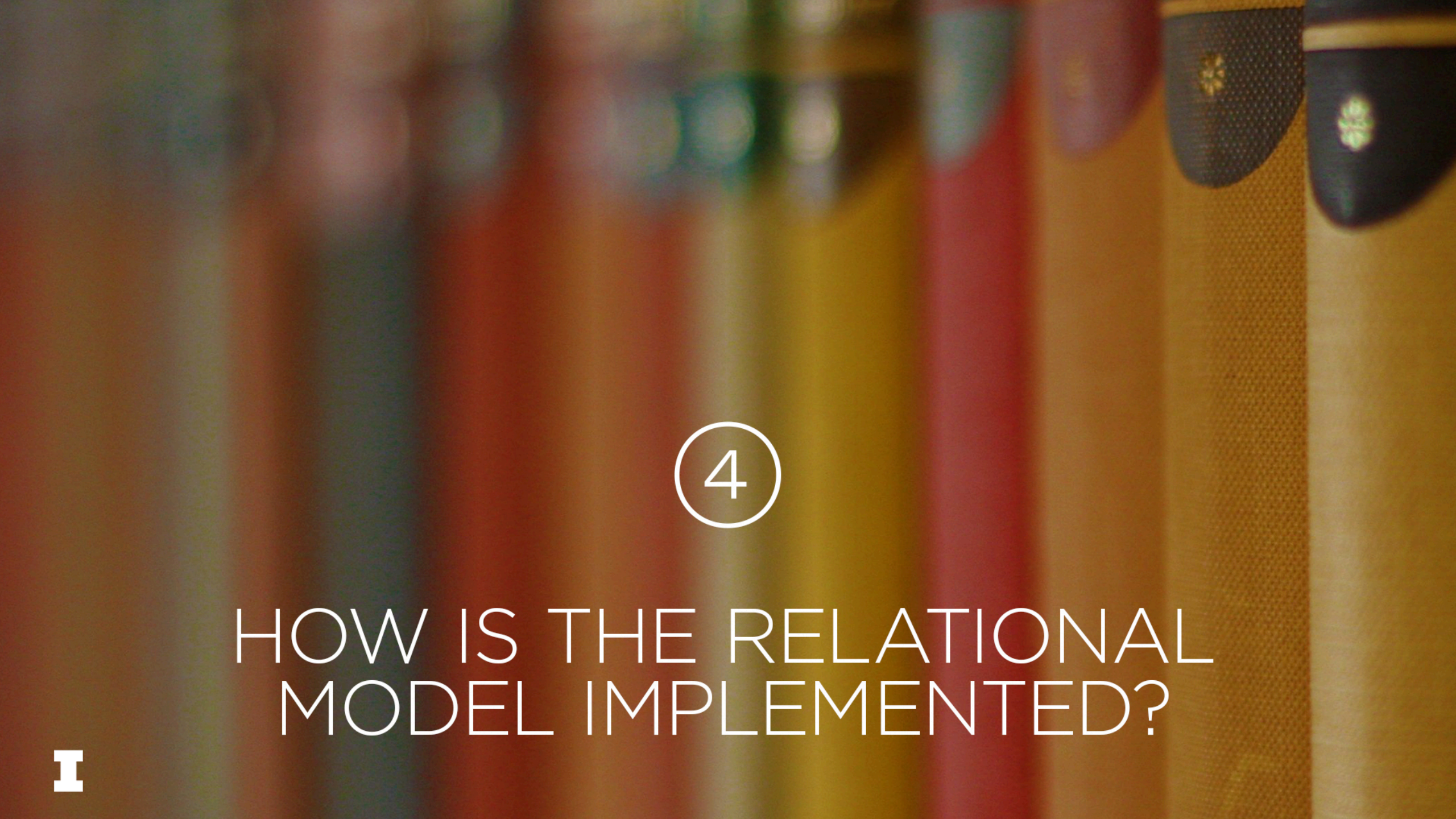
FOUNDATIONS of DATA CURATION

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

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



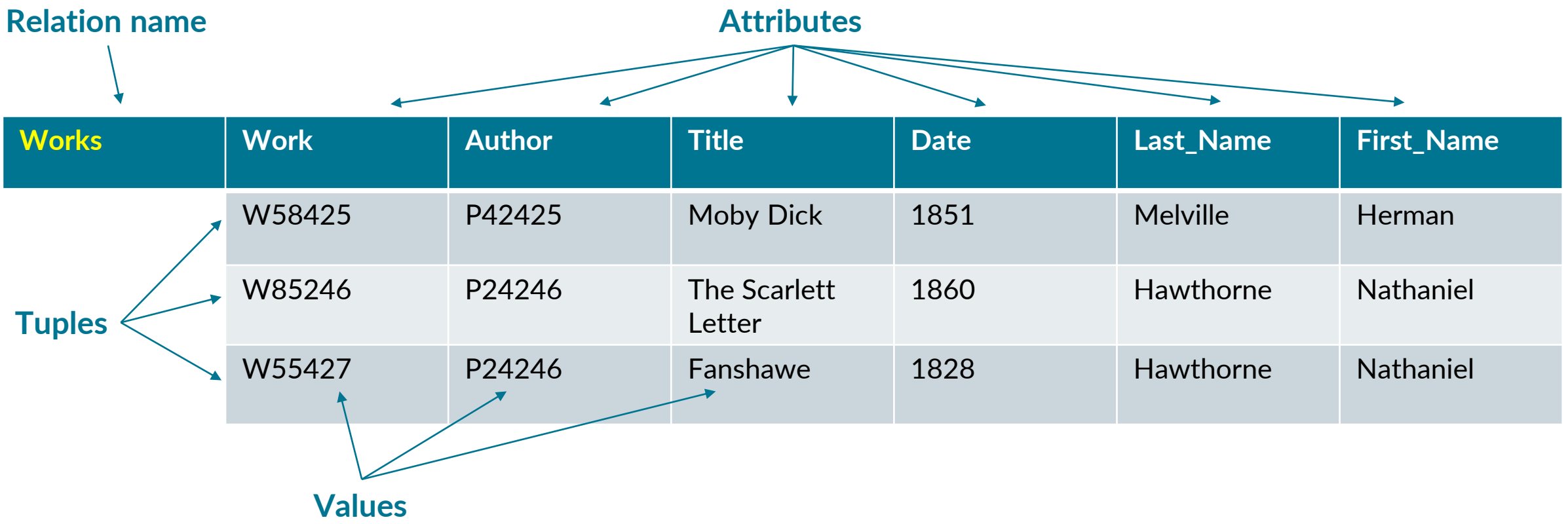
DATA MODELS: RELATIONS


4
HOW IS THE RELATIONAL MODEL IMPLEMENTED?

The Relational Model In More Detail

- Terminology
- Schemas
- Normalization
- Constraints
- Query languages

Relation Terminology



Schemas (generally)

Schema: a general term for a specification of how data is or will be organized
they may also specify: vocabulary, syntax, data types, attributes, value ranges, etc.

Schemas are written in a *schema language*.
Most schemas can themselves be processed by software.

Schemas can be used to:

- configure access and retrieval
- map between levels of abstraction
- support validattion
- create structured interfaces for input
- support inferencing and analysis
- support format conversions
- support documentation

Schemas for relations

A simple table schema (or *relation schema*):

```
AuthorTable (authorID, last, first)
```

A simple relational database schema:

```
{
AuthorTable (authorID, last, first)
WorkTable   (workID, authorID, title, date)
}
```


Normalization and Functional Dependencies

Functional dependency:

Suppose that whenever two tuples agree on *Author*, they will also agree on *Last_Name*

Work	Author	Title	Date	Last_Name	First_Name
W58425	P42425	Moby Dick	1851	Melville	Herman
W85246	P24246	The Scarlett Letter	1860	Hawthorne	Nathaniel
W55427	P24246	Fanshawe	1828	Hawthorne	Nathaniel

Normalization

Works	***Work***	*Author*	Title	Date
	W58425	P42425	Moby Dick	1851
	W85246	P24246	The Scarlett Letter	1860
	W55427	P24246	Fanshawe	1828

People	***Person***	Last_Name	First_Name
	P42425	Melville	Herman
	P24246	Hawthorne	Nathaniel

Keys

Works	*Work*	*Author*	Title	Date
	W58425	P42425	Moby Dick	1851
	W85246	P24246	The Scarlett Letter	1860
	W55427	P24246	Fanshawe	1828

Primary key: Each value for *Work* attribute identifies one work. Each value for *Person* attribute identifies one person.

Foreign key: *Author* references primary key “Person” of “People” table

People	*Person*	Last_Name	First_Name
	P42425	Melville	Herman
	P24246	Hawthorne	Nathaniel

Normalization and data curation

Understanding functional dependencies is important to data curation because they

- allow the use of normalization to reduce redundancies that cause error and inconsistency, as well as degrade efficiency of updates and validation
- allow a developer, user, or analyst to reason about how data may be manipulated or reorganized

Constraints and data curation

Constraints such as key constraints, data types, data ranges, referential integrity, etc. are critical to data curation.

- They help to model the real world, real states of affairs, with greater complexity and expressiveness than relational model alone
- They support validation and consistency
- They reflect what may be *assumed* by users, application developers, storage structures, curators… etc.

Query languages and data curation

The relational model supports the use of well-understood query languages, rather than idiosnycratic language based on unique structures.

This not only supports shared learning, training, documentation, and tools, but ensures that retrieval, views, and calculations have well-defined semantics and will perform as expected.

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.