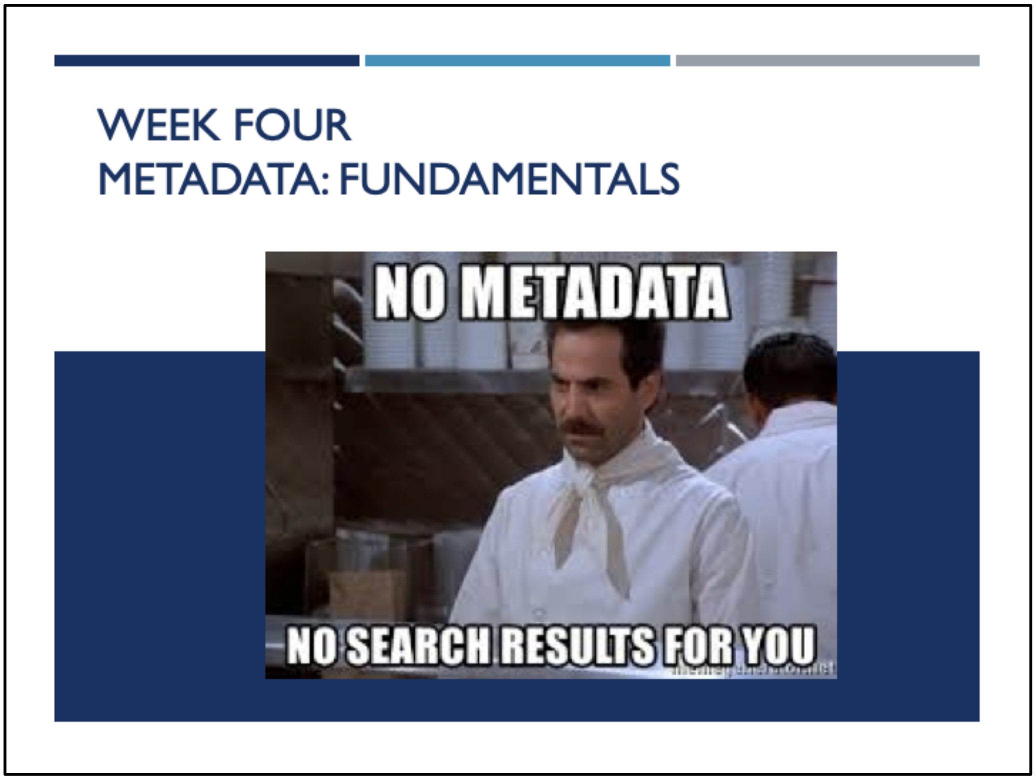
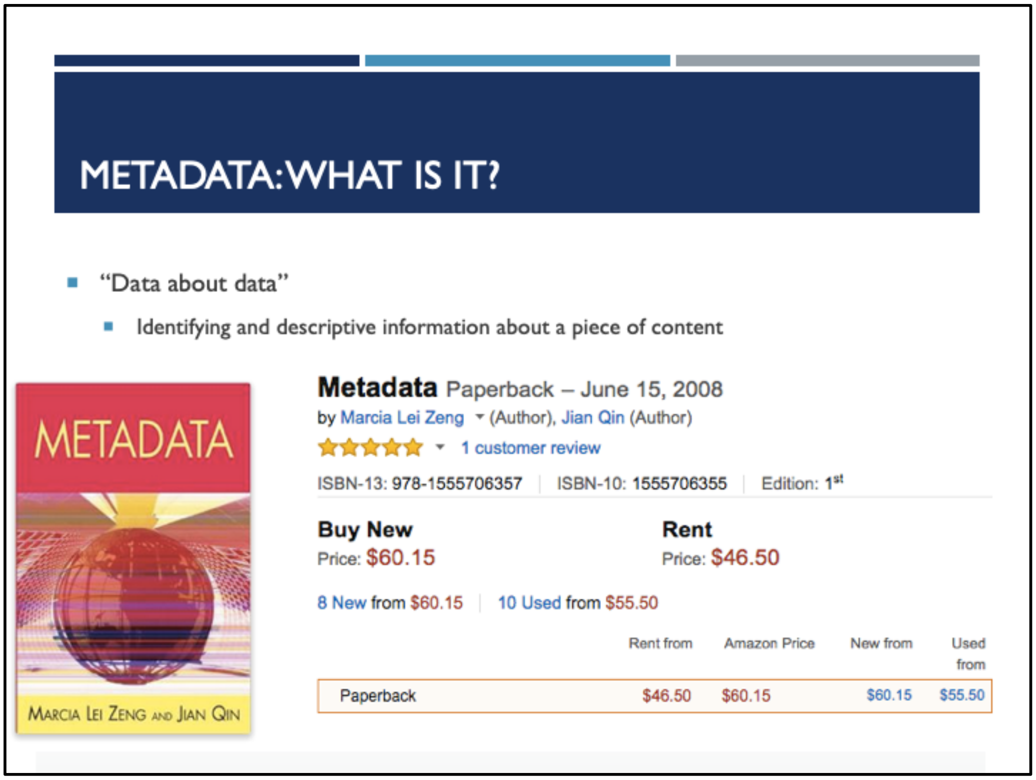


WEEK FOUR
METADATA: FUNDAMENTALS

This week we are starting a new two week unit on metadata, which is kind of like the glue that will hold our digital library together and make things findable.



METADATA: WHAT IS IT?

- "Data about data"
  - Identifying and descriptive information about a piece of content

Metadata Paperback – June 15, 2008
by Marcia Lei Zeng (Author), Jian Qin (Author)
★★★★★ 1 customer review
ISBN-13: 978-1555706357 | ISBN-10: 1555706355 | Edition: 1st

Buy New Price: $60.15
Rent Price: $46.50

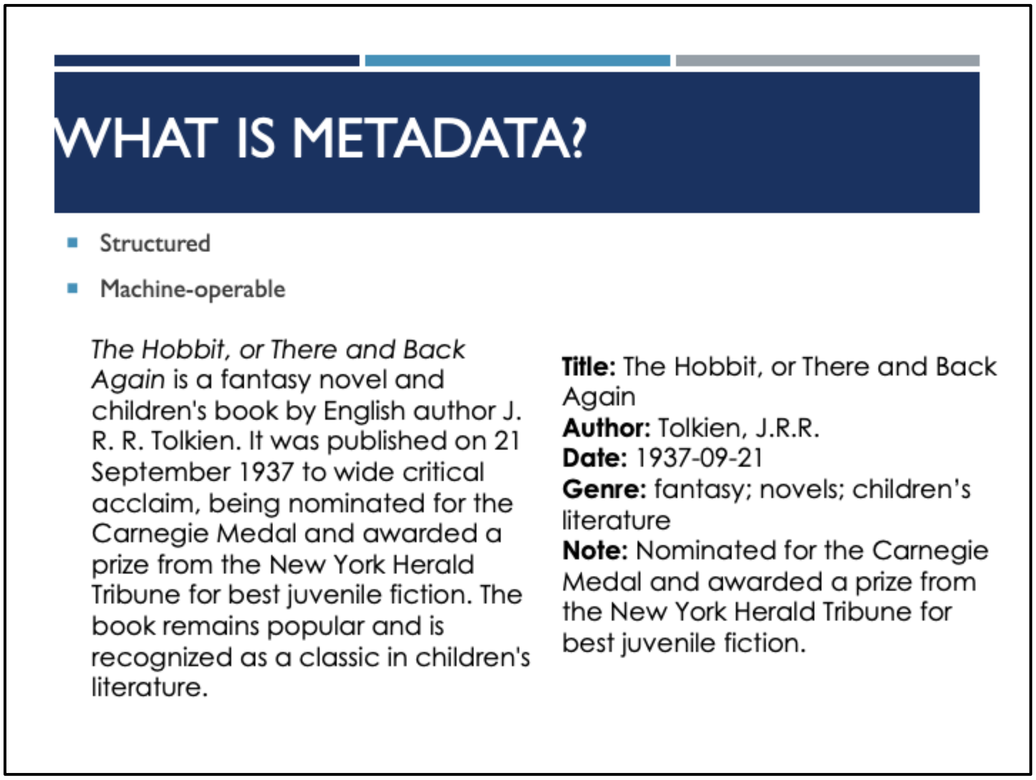
8 New from $60.15 | 10 Used from $55.50

	Rent from	Amazon Price	New from	Used from
Paperback	$46.50	$60.15	$60.15	$55.50

The definition you've probably heard of metadata is "data about data" and to be honest, while that sounds short and pithy, it's still hard to understand.

So let's start by thinking about metadata in a really generic way: metadata is identifying and descriptive information about some actual thing. So the traditional bibliographic record, in MARC, is metadata – it's information about a book. In this record from Netflix, it's structured information about this movie. This is metadata from Amazon.

So really, that's what it all boils down to: a record of information about some thing. Metadata has been a part of our libraries forever through bibliographic records. By the end of this lecture, hopefully we'll understand how some of the things we can do with information search and retrieval now make metadata much more powerful than it was in just the analog or physical format of card catalogs.



WHAT IS METADATA?

- Structured
- Machine-operable

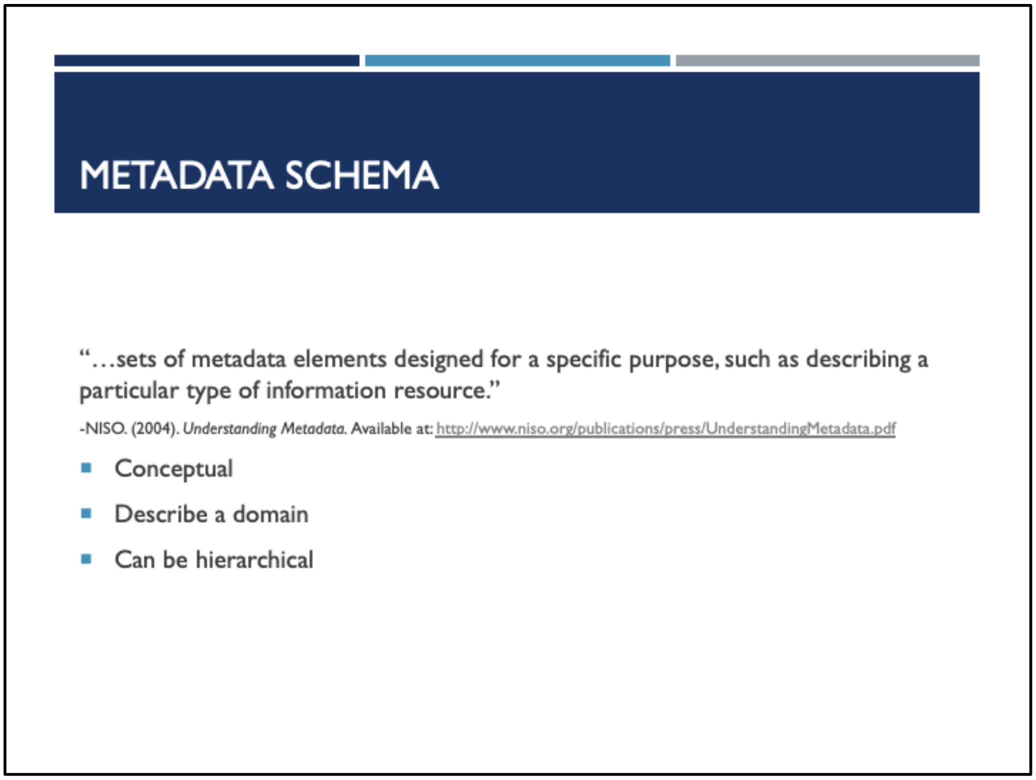
The Hobbit, or There and Back Again is a fantasy novel and children's book by English author J. R. R. Tolkien. It was published on 21 September 1937 to wide critical acclaim, being nominated for the Carnegie Medal and awarded a prize from the New York Herald Tribune for best juvenile fiction. The book remains popular and is recognized as a classic in children's literature.

Title: The Hobbit, or There and Back Again
Author: Tolkien, J.R.R.
Date: 1937-09-21
Genre: fantasy; novels; children's literature
Note: Nominated for the Carnegie Medal and awarded a prize from the New York Herald Tribune for best juvenile fiction.

One of the main features of metadata is that it is structured. So that blob of text about the Hobbit on the left contains the same information as that structured data on the right, but we'd be more likely to think of the data on the right as metadata. The structure comes from the fact that each piece of information is labeled as such, the title is labeled as a title. The author as an author.

That structure in the data, allows it to be machine-operable…because it labels individual bits of information for what they are, you can automate the handling and understanding of it. You could extract out all the titles into a separate list, for example, because they are all labeled with that string of letters: t.i.t.l.e.

The structure is what makes any kind of text metadata rather than just unstructured text.



METADATA SCHEMA

"…sets of metadata elements designed for a specific purpose, such as describing a particular type of information resource."

-NISO. (2004). *Understanding Metadata*. Available at: http://www.niso.org/publications/press/UnderstandingMetadata.pdf

- Conceptual
- Describe a domain
- Can be hierarchical

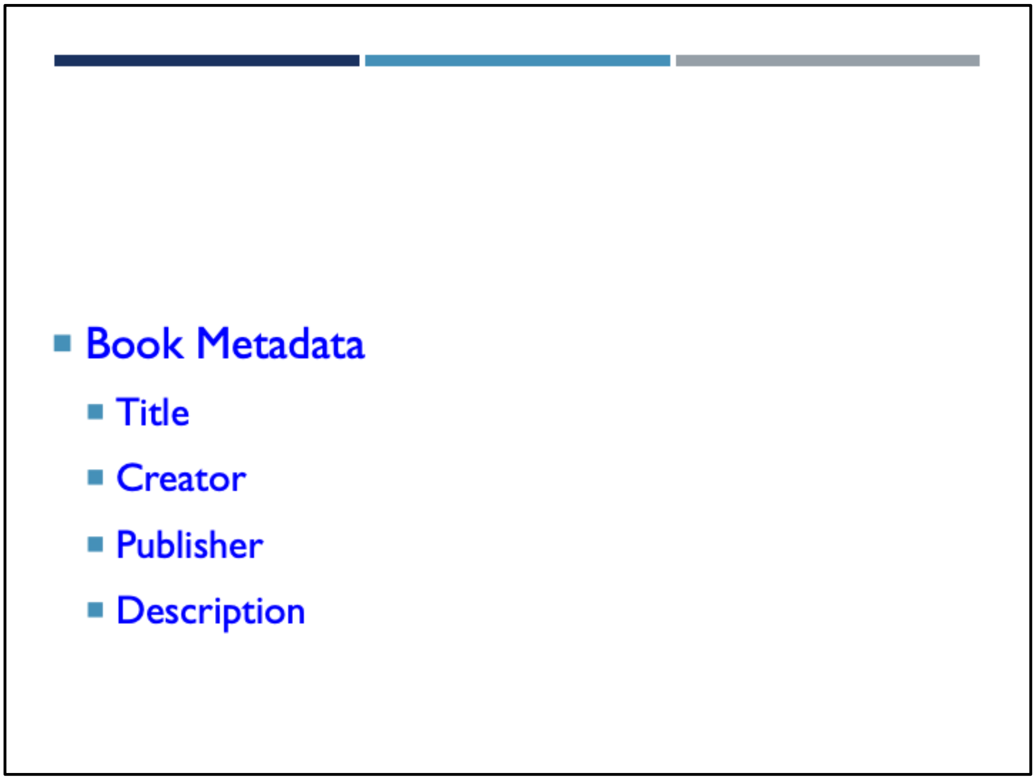
Metadata standards are described as part of **schemas** or **schemes**.

These are sets of labels, or **elements**, that are use for a specific purpose like describing a book, or a film, or the characteristics of a digital file, so and so forth.

Metadata schemas are primarily conceptual, meaning they don't have to be expressed in any particular computer language, but are really just about definitions of elements.

They describe a particular domain or subject…although sometimes the domain can be really broad

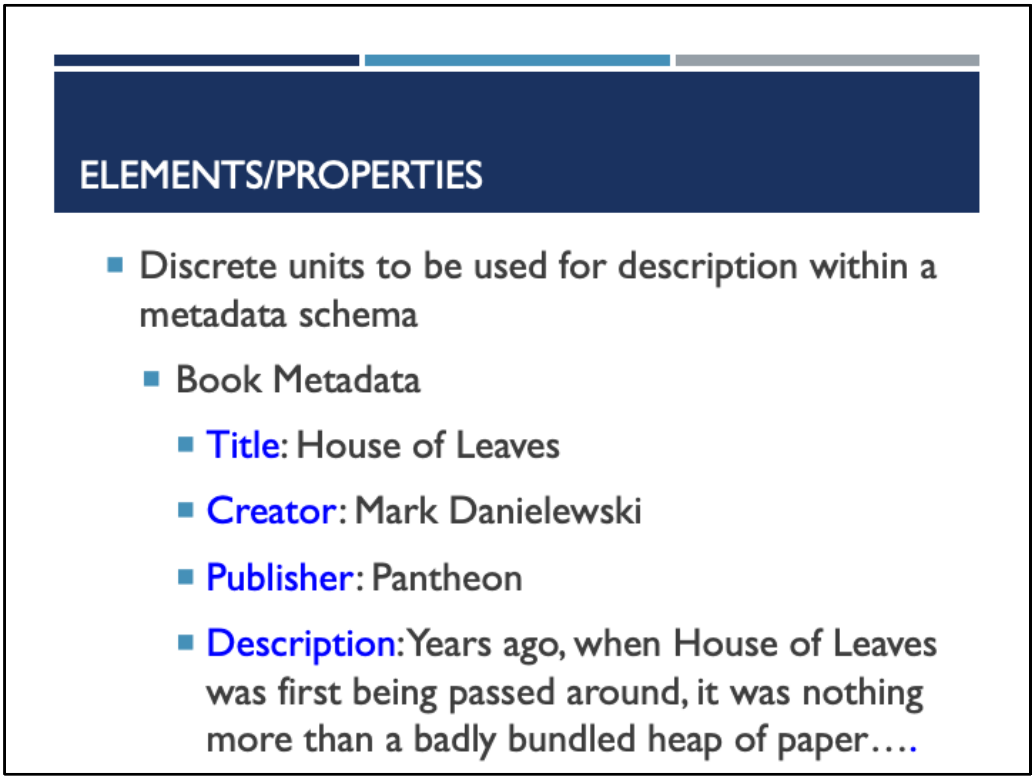
And lastly they can be hierarchical, although they don't have to be. So that means elements within a schema can be split into subgroups, all the members of which inherit certain characteristics. Let's look at some examples to understand this better.



- Book Metadata
  - Title
  - Creator
  - Publisher
  - Description

So, this is an example of a metadata schema. It’s a group of elements, show here in blue, that could be used to describe a book. It’s as simple as that really. For example, I could make up another schema that was clothing metadata and it would have color, fabric, and size…it’s just a set of characteristics that you would use to describe a type of thing.

But usually when people talk about schemas they are talking about sets of elements that have been created by a formalized group or organization (think Library of Congress, or the Getty Research Institute) and then are codified and published as a standard. We’ll talk about some of those specific schemas a bit later.

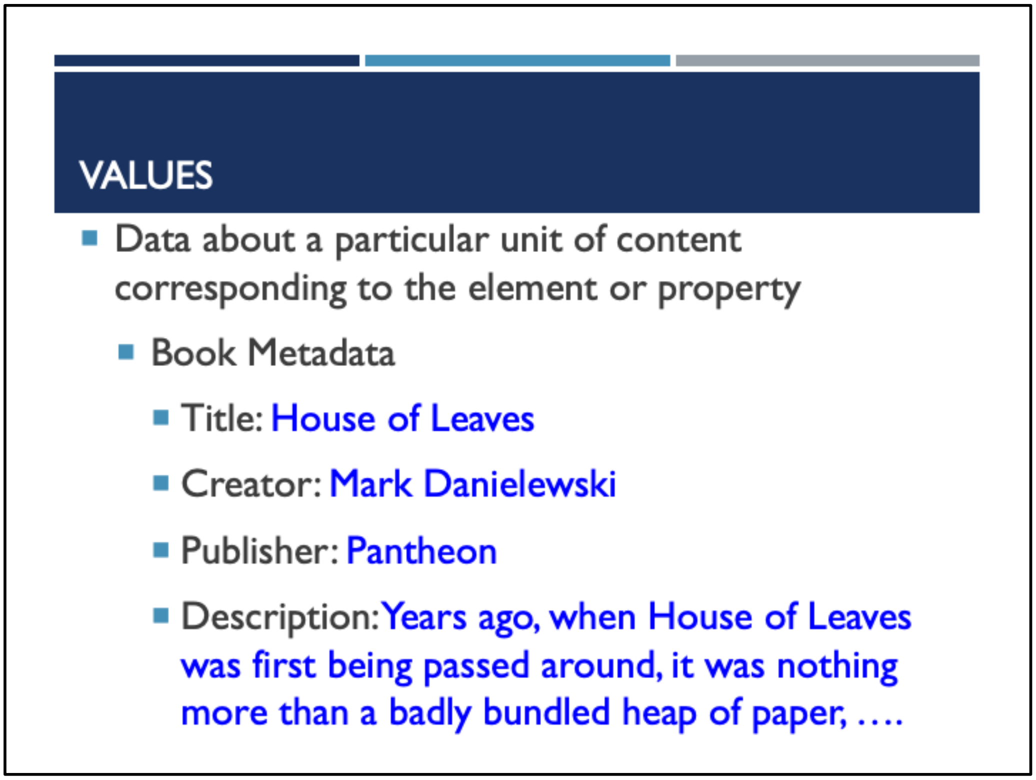


ELEMENTS/PROPERTIES

- Discrete units to be used for description within a metadata schema
  - Book Metadata
    - Title: House of Leaves
    - Creator: Mark Danielewski
    - Publisher: Pantheon
    - Description: Years ago, when House of Leaves was first being passed around, it was nothing more than a badly bundled heap of paper….

The discrete units of description within the schema are called **elements**, as I mentioned earlier, or sometimes they are also called **properties**. Those two words are generally speaking pretty interchangeable.

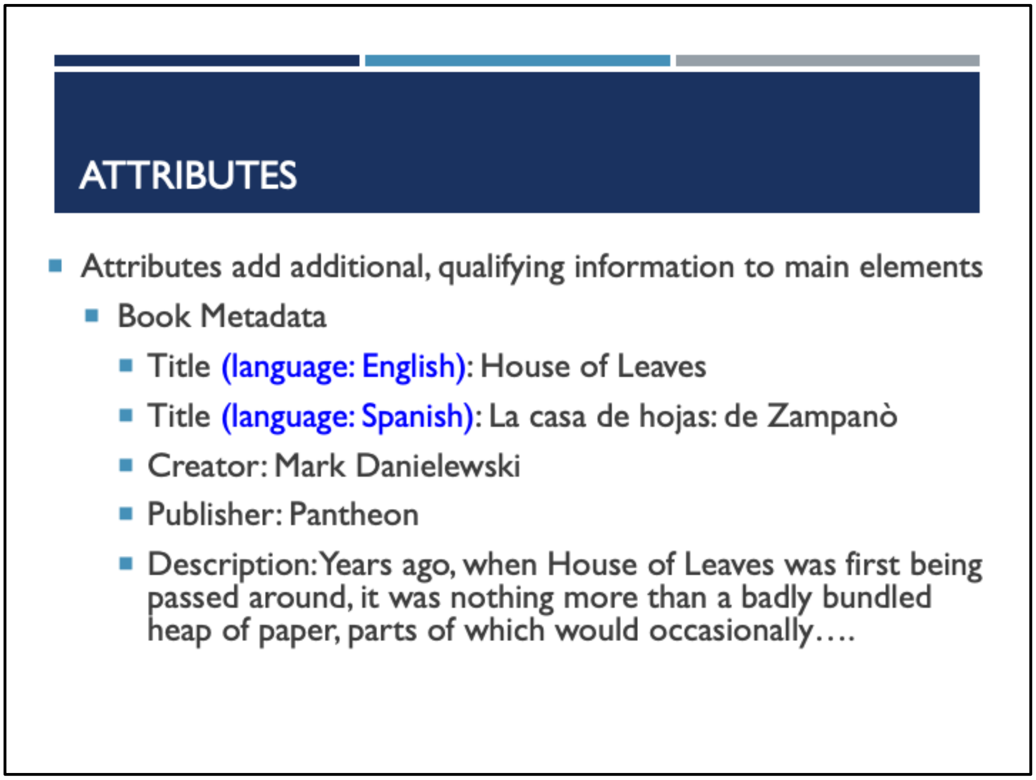
So here the elements are in blue, the discrete units of information we can describe like title, creator, description etc.



VALUES

- Data about a particular unit of content corresponding to the element or property
  - Book Metadata
    - Title: House of Leaves
    - Creator: Mark Danielewski
    - Publisher: Pantheon
    - Description: Years ago, when House of Leaves was first being passed around, it was nothing more than a badly bundled heap of paper, ….

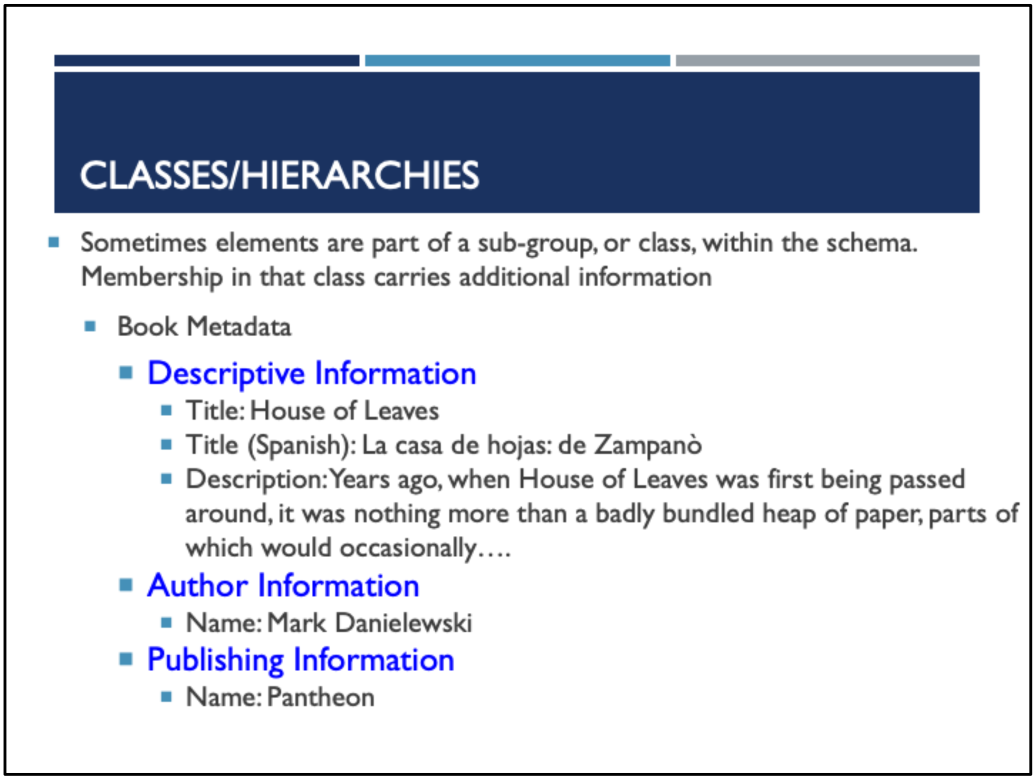
Values is the term applied to any descriptive data you express using the schema. So in blue here, the specific information about this book, House of Leaves, is in blue. Those are the metadata values for this particular record,



ATTRIBUTES

- Attributes add additional, qualifying information to main elements
  - Book Metadata
    - Title (language: English): House of Leaves
    - Title (language: Spanish): La casa de hojas: de Zampanò
    - Creator: Mark Danielewski
    - Publisher: Pantheon
    - Description: Years ago, when House of Leaves was first being passed around, it was nothing more than a badly bundled heap of paper, parts of which would occasionally….

Attributes are like qualifiers for elements. They distinguish between elements of the same type and add information. So continuing with our book metadata, example the attributes here would be the language and the language English and Spanish would be the values of that attribute for those two elements.

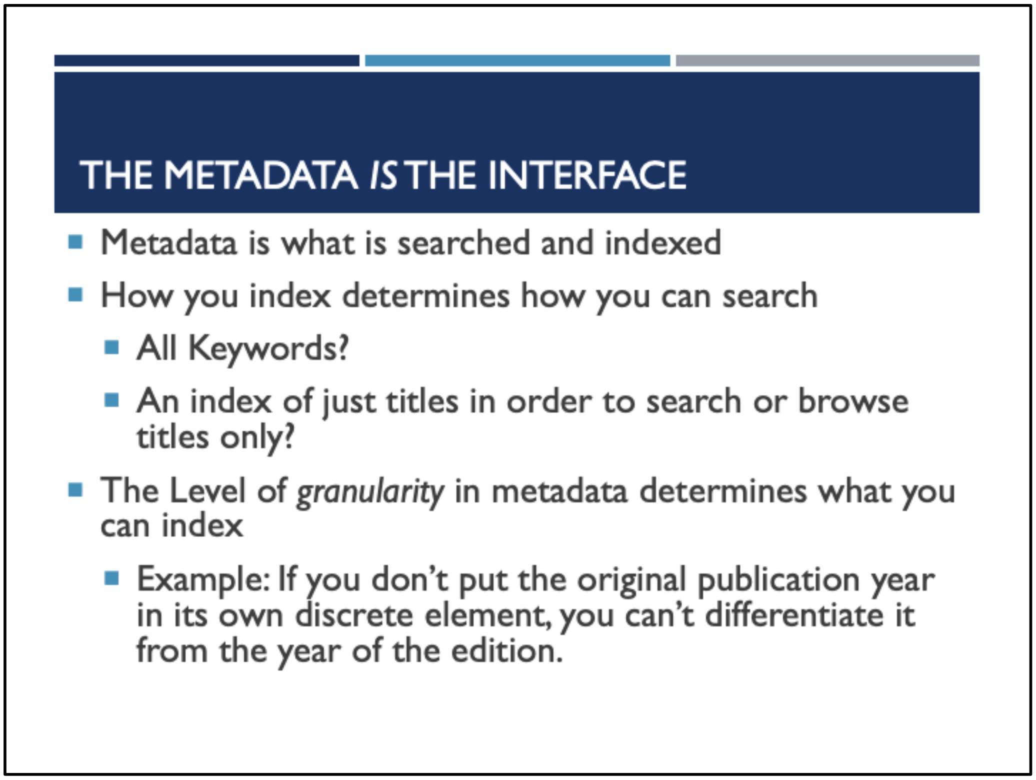


CLASSES/HIERARCHIES

- Sometimes elements are part of a sub-group, or class, within the schema. Membership in that class carries additional information
  - Book Metadata
    - Descriptive Information
      - Title: House of Leaves
      - Title (Spanish): La casa de hojas: de Zampanò
      - Description: Years ago, when House of Leaves was first being passed around, it was nothing more than a badly bundled heap of paper, parts of which would occasionally….
    - Author Information
      - Name: Mark Danielewski
    - Publishing Information
      - Name: Pantheon

Finally, I mentioned that schemas could be hierarchical. Some schemas employ the use of **classes**, which are kind of like subgroupings of related properties. They also allow you to reuse a property in two places. So you could have say a class for author with a name property (and maybe some other properties not shown here like date of birth/death, stuff like that) and then a publisher class that also uses the name property.

Adding hierarchy or classes allows you to make more nuanced data and allows you to re-use classes for many records. So you can have a single publisher class instance and point to it from all the records for books that they publish.



THE METADATA *IS* THE INTERFACE

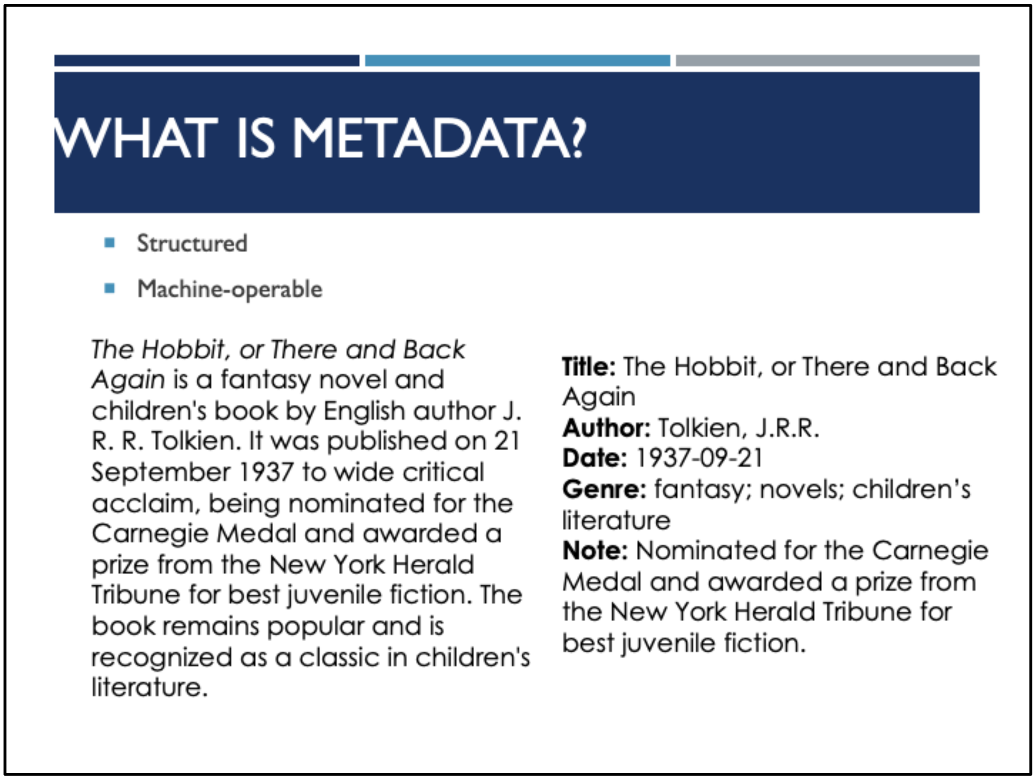
- Metadata is what is searched and indexed
- How you index determines how you can search
  - All Keywords?
  - An index of just titles in order to search or browse titles only?
- The Level of *granularity* in metadata determines what you can index
  - Example: If you don't put the original publication year in its own discrete element, you can't differentiate it from the year of the edition.

So now that we know a little bit about what metadata is, we'll talk about WHY it's important.

There is a paper that I didn't assign this year, but it is pretty good, that is called "The metadata IS the interface" and I think that's a great way of saying it.

The metadata is what is searched and indexed in your repository. The level of description you use, in turn determines the level of indexing you can do, which determines what you can search, as well as what you can display. So if your metadata doesn't have enough **granularity**, which is the term for how detailed the metadata is – how many things are placed in their own elements vs. maybe just put in a long descriptive paragraph -- you can't distinguish between things in that data for browsing or display.

Think back to that earlier slide where we showed the structured and unstructured data.



WHAT IS METADATA?

- Structured
- Machine-operable

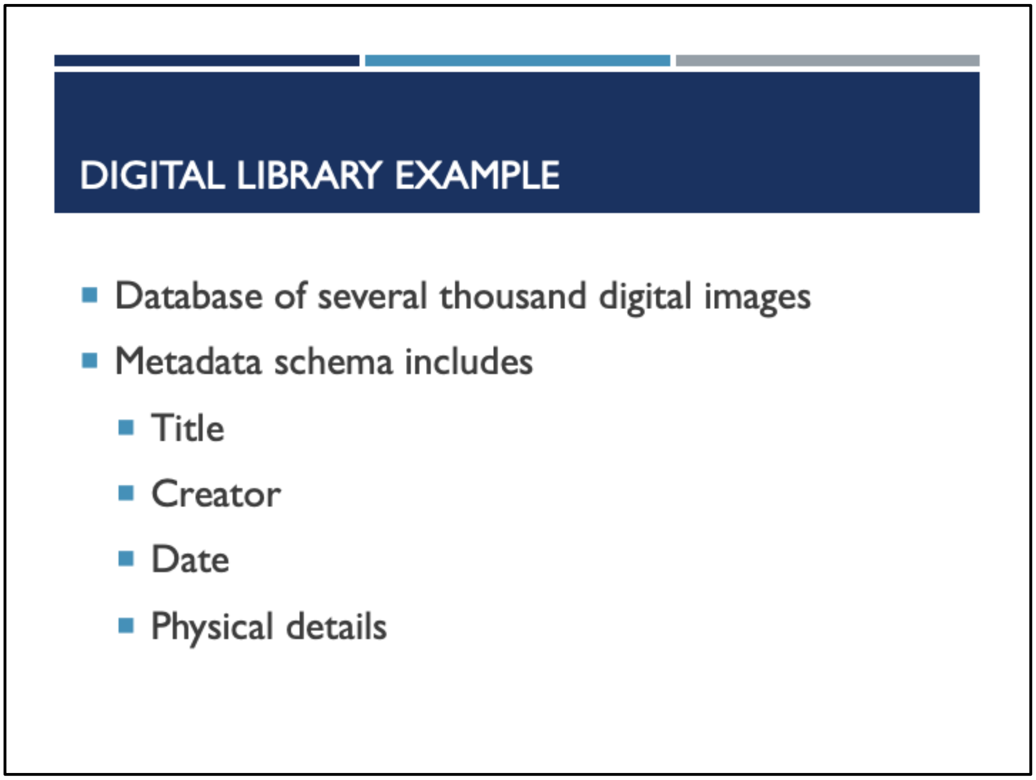
The Hobbit, or There and Back Again is a fantasy novel and children's book by English author J. R. R. Tolkien. It was published on 21 September 1937 to wide critical acclaim, being nominated for the Carnegie Medal and awarded a prize from the New York Herald Tribune for best juvenile fiction. The book remains popular and is recognized as a classic in children's literature.

Title: The Hobbit, or There and Back Again
Author: Tolkien, J.R.R.
Date: 1937-09-21
Genre: fantasy; novels; children's literature
Note: Nominated for the Carnegie Medal and awarded a prize from the New York Herald Tribune for best juvenile fiction.

The structured data on the right can be indexed, because it has been labeled. I can put the title in a table with all the other titles, the author in a table with all the other authors.

When someone searches and finds this record, then I can display the author name with the label "author" and then maybe find other records with the same author name.

In the unstructured text you would need human intelligence to determine where the title and author are.

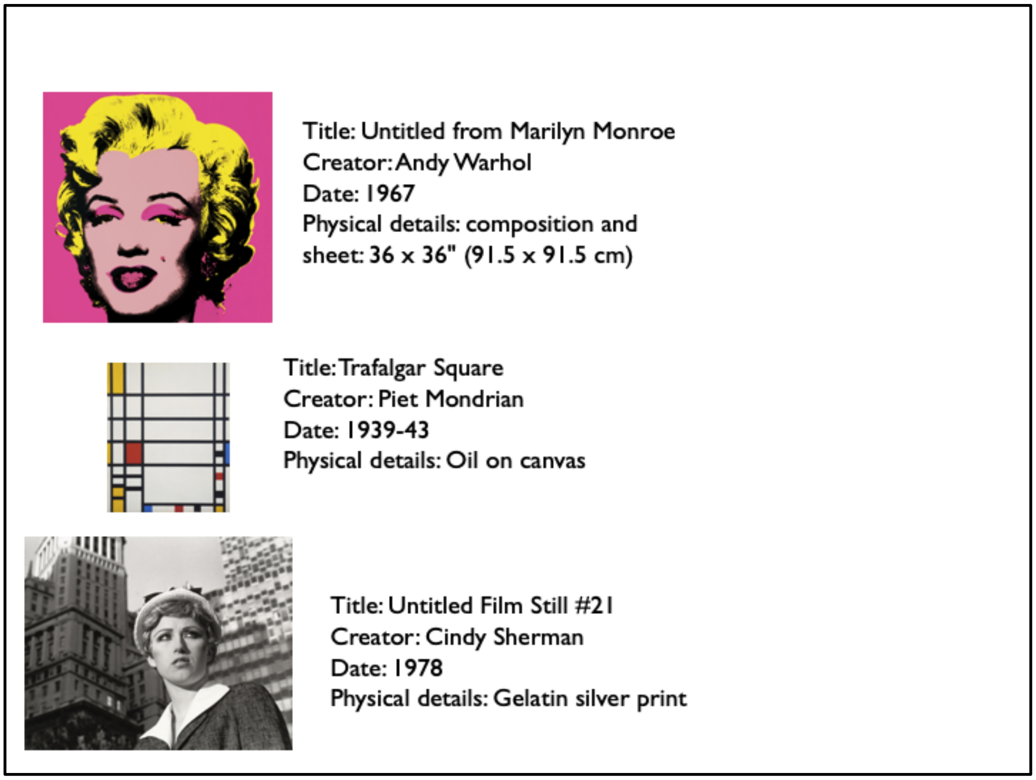


DIGITAL LIBRARY EXAMPLE

- Database of several thousand digital images
- Metadata schema includes
  - Title
  - Creator
  - Date
  - Physical details

As another example of this, let’s imagine we are managing a database of several thousand images.

We use a metadata schema that allows us to enter the title, creator, date, and some physical details.

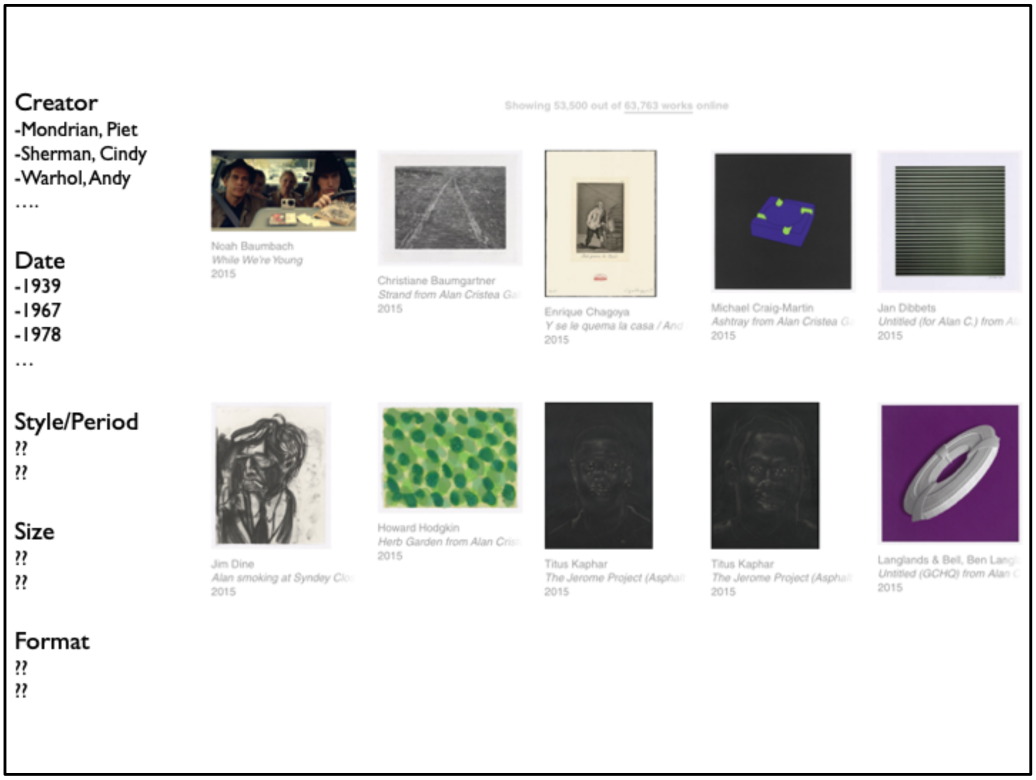


Title: Untitled from Marilyn Monroe
Creator: Andy Warhol
Date: 1967
Physical details: composition and sheet: 36 x 36" (91.5 x 91.5 cm)

Title: Trafalgar Square
Creator: Piet Mondrian
Date: 1939-43
Physical details: Oil on canvas

Title: Untitled Film Still #21
Creator: Cindy Sherman
Date: 1978
Physical details: Gelatin silver print

So here’s what three potential images might have as metadata



Creator
-Mondrian, Piet
-Sherman, Cindy
-Warhol, Andy
….

Date
-1939
-1967
-1978
…

Style/Period
??
??

Size
??
??

Format
??
??

Showing 53,500 out of 63,763 works online

Noah Baumbach
While We're Young
2015

Christiane Baumgartner
Strand from Alan Cristea Ga
2015

Enrique Chagoya
Y se le quema la casa / And
2015

Michael Craig-Martin
Ashtray from Alan Cristea G
2015

Jan Dibbets
Untitled (for Alan C.) from Al
2015

Jim Dine
Alan smoking at Syndey Clo
2015

Howard Hodgkin
Herb Garden from Alan Cris
2015

Titus Kaphar
The Jerome Project (Asphal
2015

Titus Kaphar
The Jerome Project (Asphal
2015

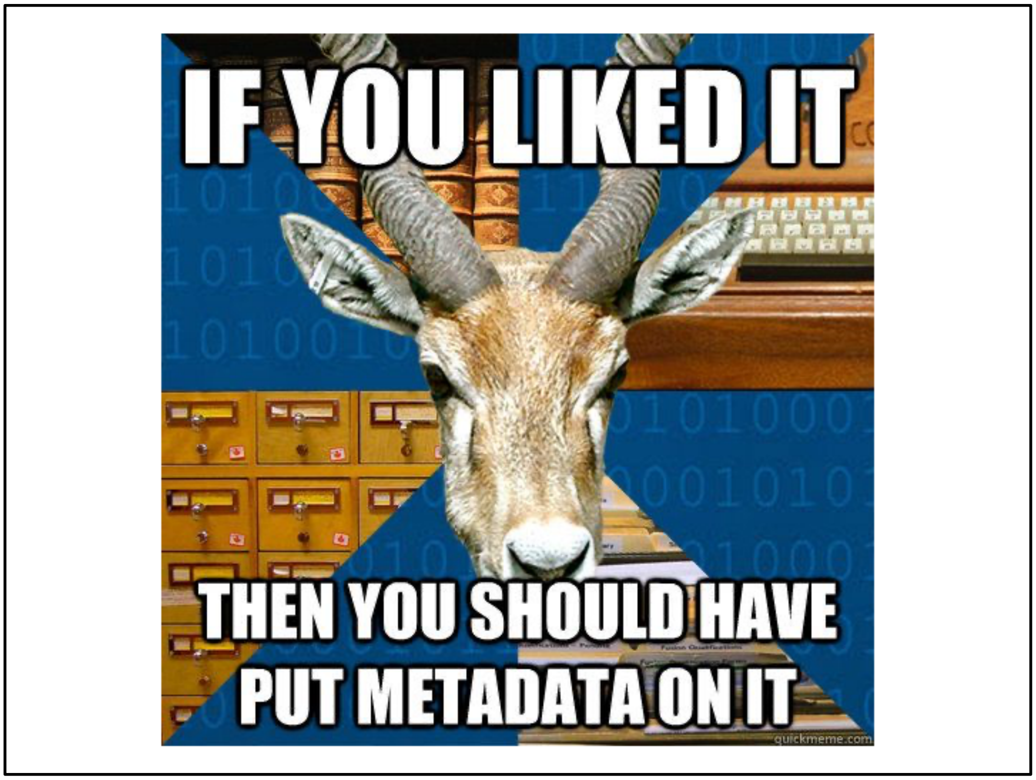
Langlands & Bell, Ben Langl
Untitled (GCHQ) from Alan C
2015

Now, when we design the interface, let's we want to add facets for narrowing down search results. Those facets are just an index of what was in metadata fields. So we can easily make an index of creators and dates.

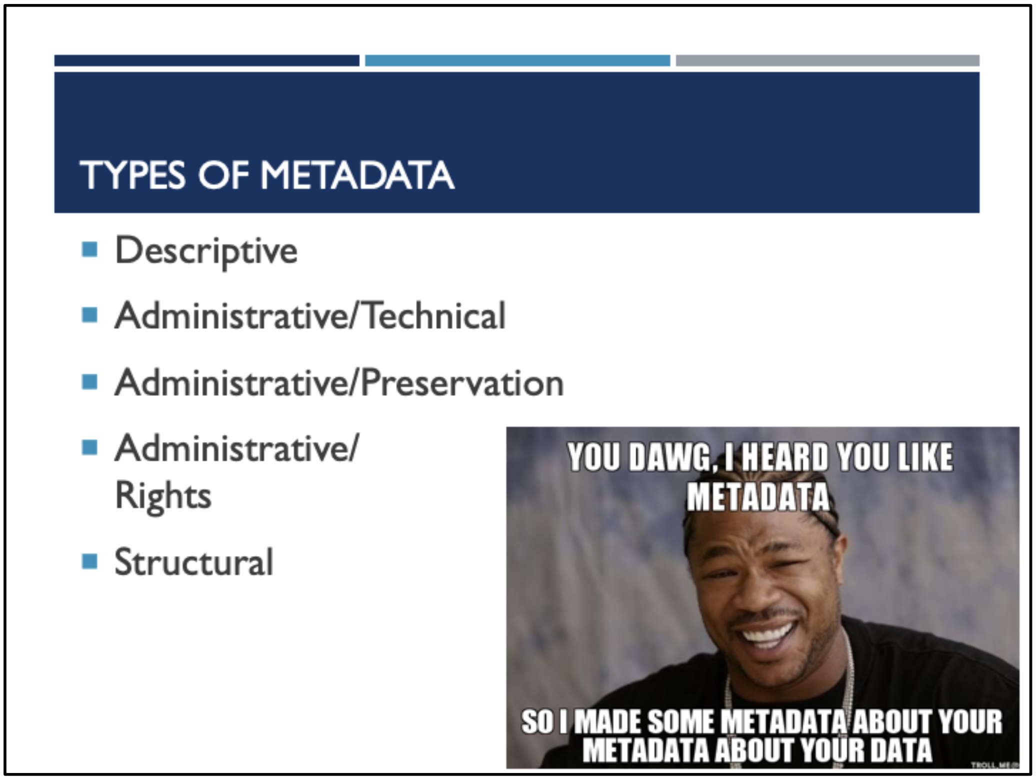
But we couldn't do an index of styles or artistic movements because that's not in the metadata. We might have had details about the size and format of the paintings (let's look back at the last slide to check)….

..but they are smashed into the same field so we couldn't do a facet for each. Unless you wrote a program that could parse through the value of those physical description fields and look for specific phrases like "oil on canvas" or look for indication of size (like abbreviations for feet or inches), there is no way to index those separately. And that wouldn't be easy to do.

The point here is that decisions about metadata made early on in the process have far-reaching ramifications



So long story short, channeling Beyonce, if you liked it then you should have put metadata on it.



TYPES OF METADATA

- Descriptive
- Administrative/Technical
- Administrative/Preservation
- Administrative/ Rights
- Structural

YOU DAWG, I HEARD YOU LIKE METADATA
SO I MADE SOME METADATA ABOUT YOUR METADATA ABOUT YOUR DATA
TROLL.ME

Thinking about metadata from a broader perspective now, most schemas are created with a specific purpose. They perform a specific function. Generally, they fall into one of these types:

Descriptive: describing the content

Administrative/Technical: describing the format of the material, typically the digital format characteristics

Administrative/Preservation: the kind of details needed to preserve the material, typically again very technical details about the creation of the digitized object

Administrative/Rights: information about the intellectual proper status of the material

The reason I've put "administrative" in all of these, is that sometimes you will see parts or all of these functions combined in a schema and termed "administrative" metadata

Finally, **Structural Metadata** describes how a multi-part item is composed. If you remember last week we talked about lots of different wrapper files, those are all a kind of structural metadata. The wrapper is a single record that encapsulates all the relevant metadata records in different schemas or points to the data files needed for the use of the digital object.

MAJOR SCHEMAS

MODS

PREMIS

EAD

TEI

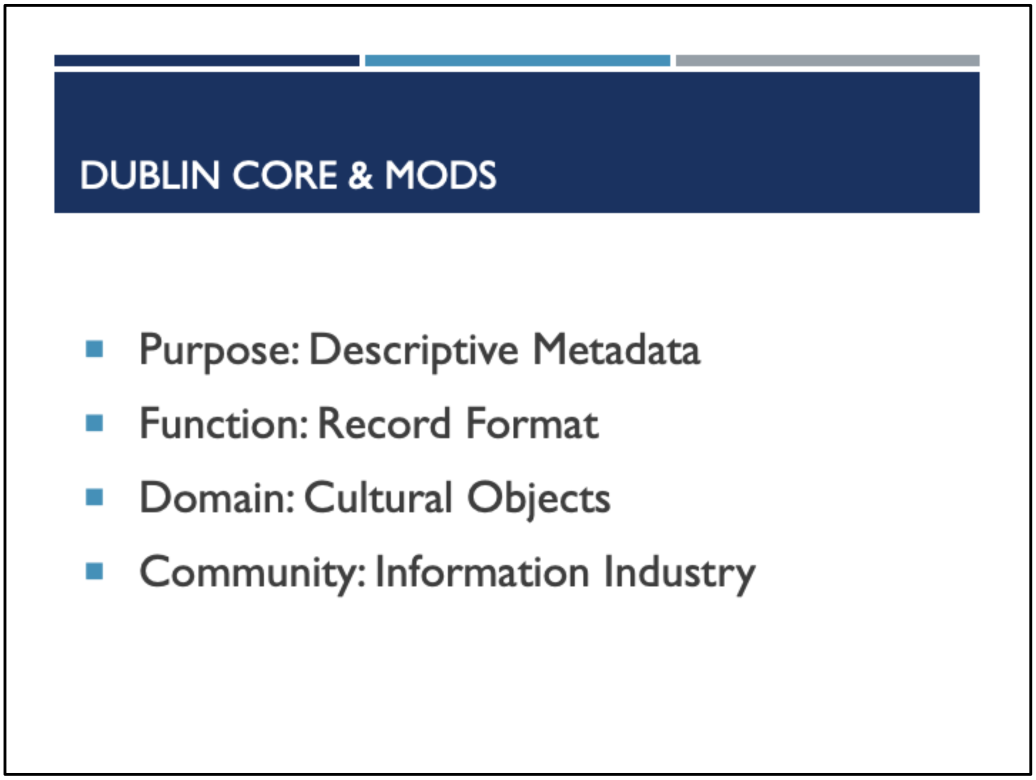
TEXT ENCODING INITIATIVE

Dublin Core®

METS Metadata Encoding & Transmission Standard

Official Web Site

There are lots of standardized schemas that are very commonly used in libraries, museums, and archives which have been developed by professional groups and organizations. Most of the ones here are used for different purposes. We won’t talk about all of them this week, but will use two of them as examples.



DUBLIN CORE & MODS

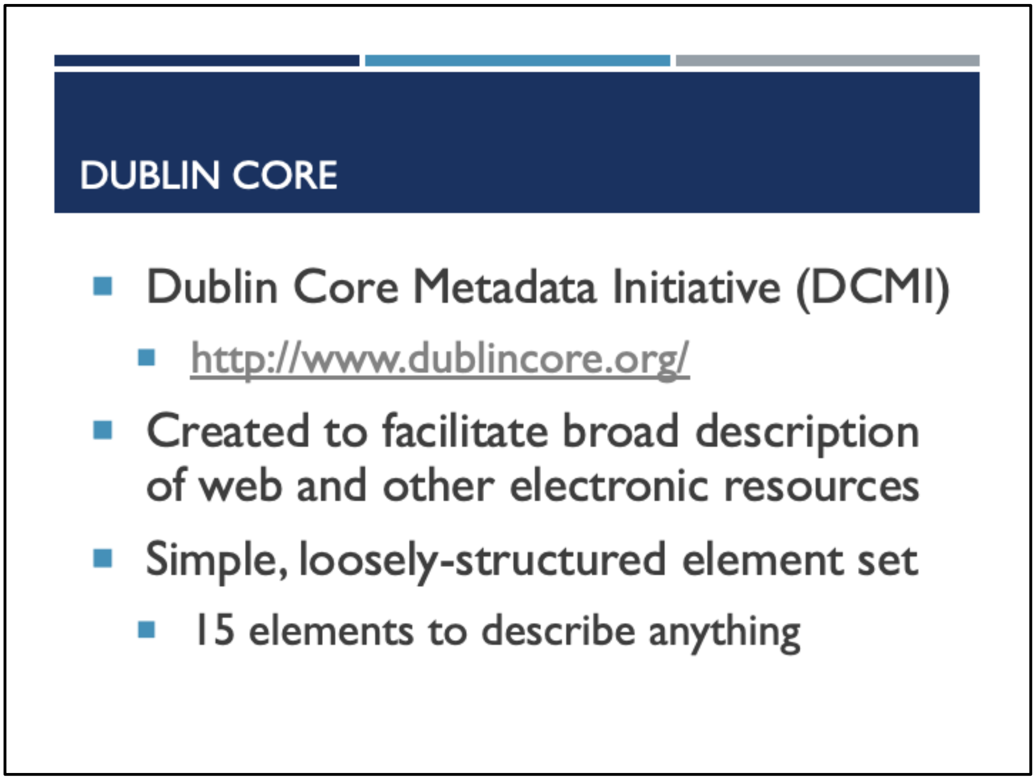
- Purpose: Descriptive Metadata
- Function: Record Format
- Domain: Cultural Objects
- Community: Information Industry

Dublin Core and **MODS** will be the two schemas we will use in our first assignment next week.

The four categories you see here are sometimes used to describe the characteristics of metadata:

Purpose: the type, like we just discussed
Function: this means how it is used, is it used to create a record, or a wrapper? Is it instead a thesaurus or controlled vocabulary which are related concepts
Domain: describes what type of material it is describing, so these are both related to the cultural heritage materials, other types could specifically be about text or a/v or images, or could be about educational materials, or consumer goods, or books, etc
Finally**, Community**: describes who uses the standard, in this case the Information Industry (libraries, archives, museums) as opposed to say retailers, or engineers, or musicologists

Both of these schema are descriptive metadata schema, they are used for creating metadata records about cultural heritage materials (i.e. the kinds of stuff in libraries, archives, and museums)



DUBLIN CORE

- Dublin Core Metadata Initiative (DCMI)
  - http://www.dublincore.org/
- Created to facilitate broad description of web and other electronic resources
- Simple, loosely-structured element set
  - 15 elements to describe anything

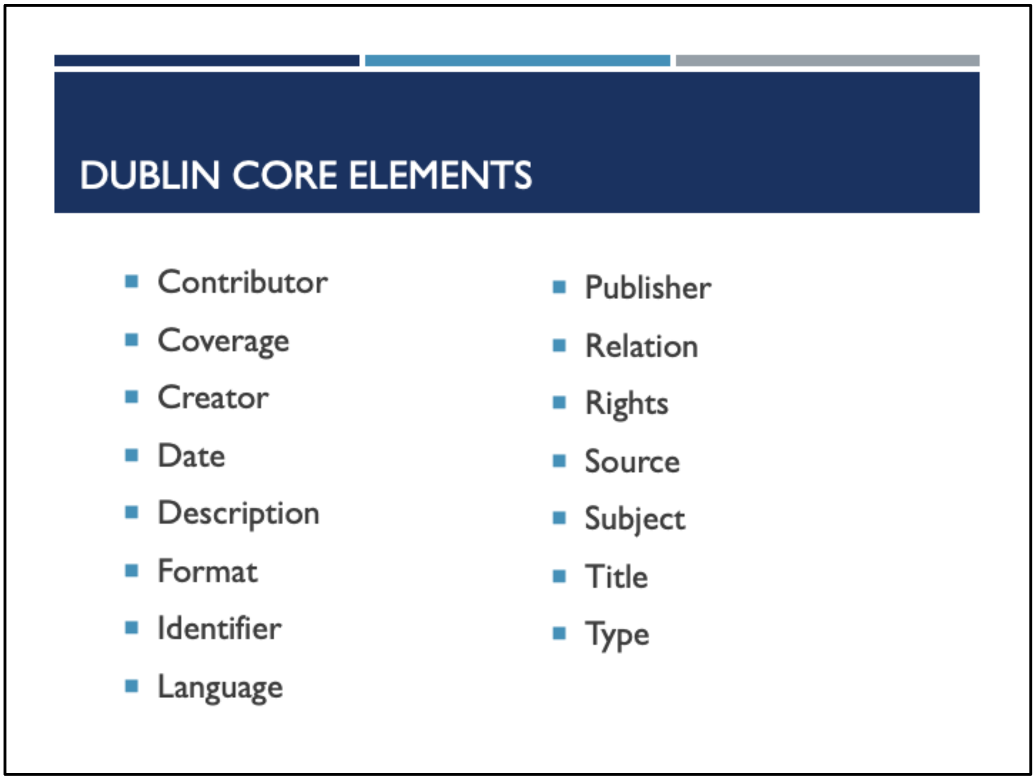
First up is **Dublin Core**.

This was created by a group called the Dublin Core Metadata Initiative, I believe because they met in Dublin, Ohio (which is where OCLC is, although there is no formal relationship there) and that’s how the name was created.

It was created to facilitate description of any and all kinds of objects on the web.

It was also created specifically to be very simple, and very loosely structured

So there are only 15 elements, all are repeatable as many times as you need



DUBLIN CORE ELEMENTS

- Contributor
- Coverage
- Creator
- Date
- Description
- Format
- Identifier
- Language
- Publisher
- Relation
- Rights
- Source
- Subject
- Title
- Type

So, here’s the original Dublin Core metadata set. It was basically an attempt to agree on the basic things you need to describe and identify something in order to get everyone using the same standard.

- Contributor
- Coverage
- Creator
- Date
- Description
- Format
- Identifier
- Language

- Publisher

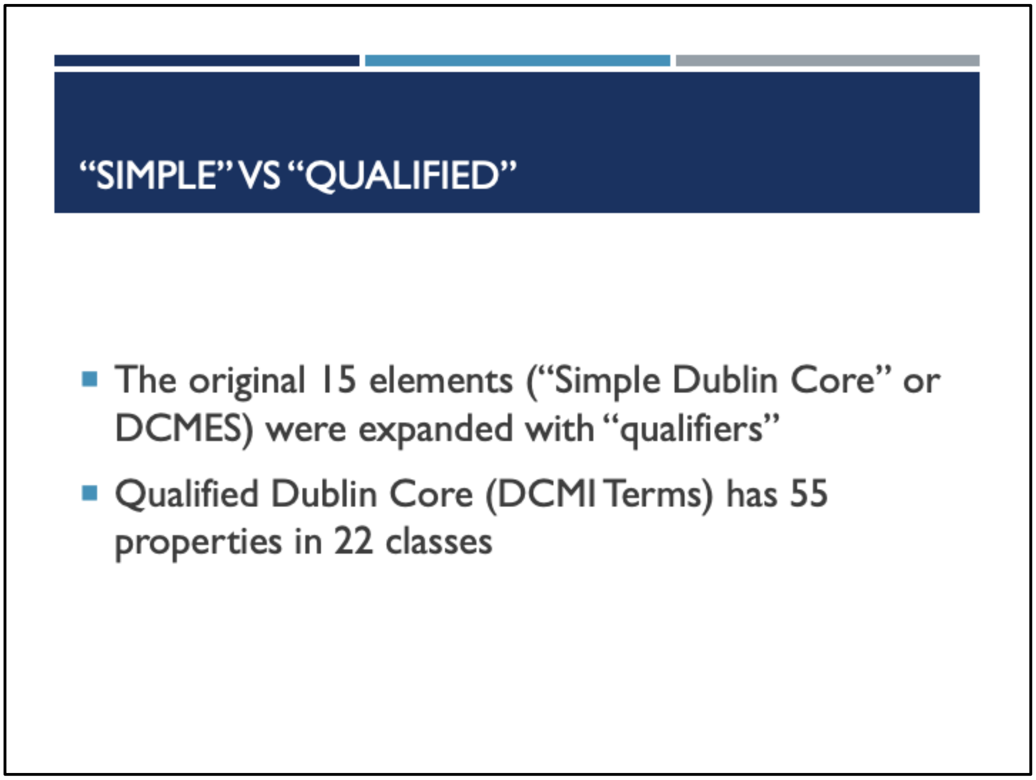
- Relation

- Rights

- Source

- Subject
- Title
- Type

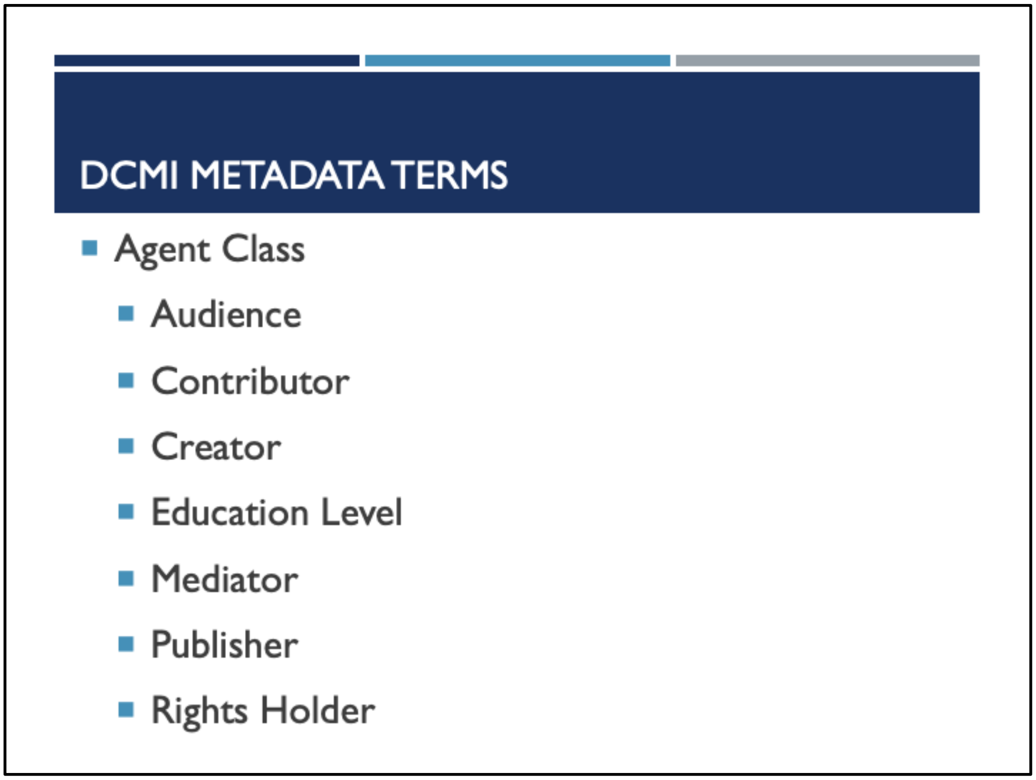
These original 15 Dublin Core Elements are the ones we are going to use for our assignment next week. There are some elements in there that cause perennial user confusion such as “coverage” which is intended to describe either the place or time period that an item is about, or both (as opposed to say the time or place it was created). But coverage is a strange word for that concept – mostly because it’s a complicated concept that would be hard to describe with a simple label.



“SIMPLE” VS “QUALIFIED”

- The original 15 elements (“Simple Dublin Core” or DCMES) were expanded with “qualifiers”
- Qualified Dublin Core (DCMI Terms) has 55 properties in 22 classes

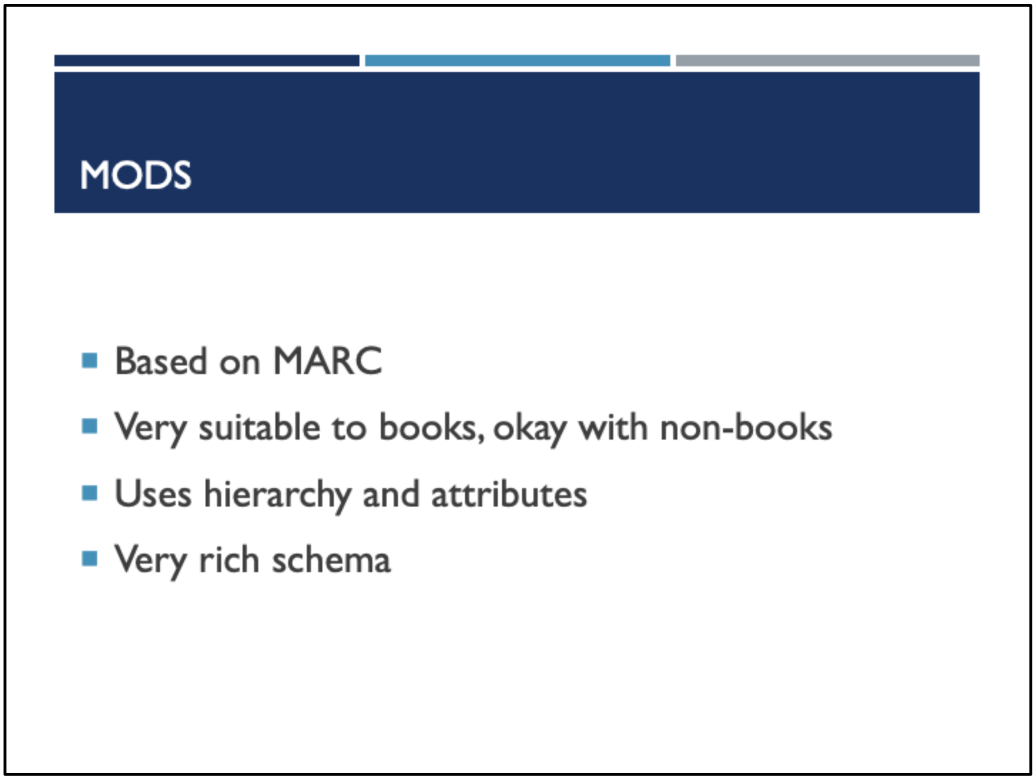
Because the original set of Dublin Core terms was in fact so simple, they were eventually expanded with qualifiers, These are basically sub-elements that provide more detail for some of those concepts like “coverage” that are very complex. Qualified Dublin Core has 55 properties which are organized in 22 classes



DCMI METADATA TERMS

- Agent Class
  - Audience
  - Contributor
  - Creator
  - Education Level
  - Mediator
  - Publisher
  - Rights Holder

I’m not going to put them all up here, but this is a look at one class, the Agent class. So you can see it’s both a reorganization of the simple Dublin Core (organizing the original creator, contributor, and publisher terms together), but it’s also an expansion.



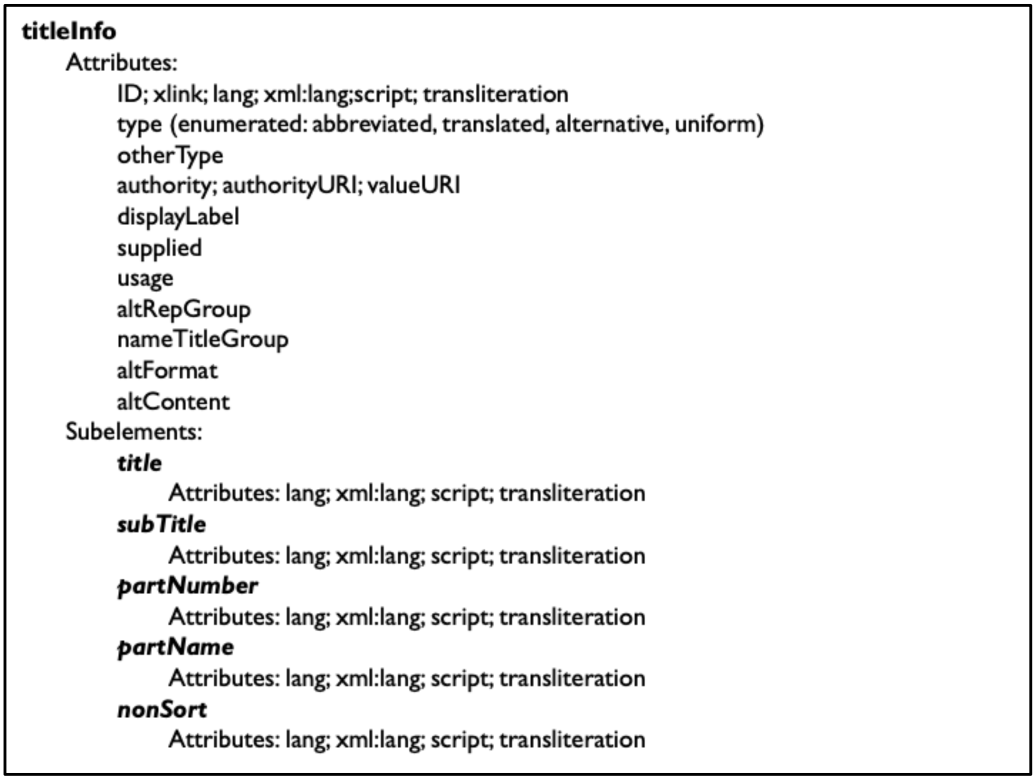
MODS

- Based on MARC
- Very suitable to books, okay with non-books
- Uses hierarchy and attributes
- Very rich schema

The next schema we'll talk is **MODS**: metadata object description schema. This is Library of Congress's attempt to make a metadata schema based on a very simplified version of MARC.

So MODS is very good for books, and it's pretty good with non-books. It's very popular for organizations that want to use more granularity than Dublin Core but still use a generalized schema suitable for many materials.

MODS is very rich, and has lots of hierarchy and a lot of attributes for many elements.



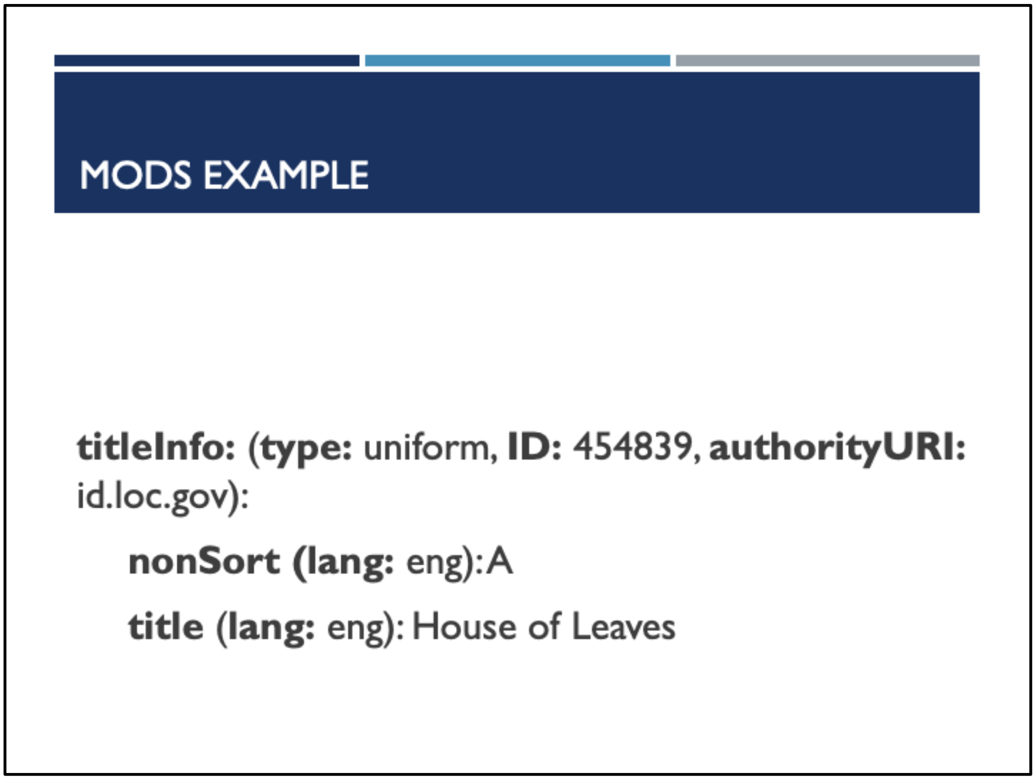
titleInfo

- Attributes:
  - ID; xlink; lang; xml:lang;script; transliteration
  - type (enumerated: abbreviated, translated, alternative, uniform)
  - otherType
  - authority; authorityURI; valueURI
  - displayLabel
  - supplied
  - usage
  - altRepGroup
  - nameTitleGroup
  - altFormat
  - altContent
- Subelements:
  - ***title***
    - Attributes: lang; xml:lang; script; transliteration
  - ***subTitle***
    - Attributes: lang; xml:lang; script; transliteration
  - ***partNumber***
    - Attributes: lang; xml:lang; script; transliteration
  - ***partName***
    - Attributes: lang; xml:lang; script; transliteration
  - ***nonSort***
    - Attributes: lang; xml:lang; script; transliteration

Here’s a look at the definition for a single element, titeInfo, in the MODS data dictionary.

titleInfo itself has 10 groupings of attributes (so you could use ID or xlink OR xml:lang from the first group to indicate some administrative identification number, and then you could also use type to describe a classification like primary or alternate title, and then maybe authority OR authorityURI, to indicate that this has a LC controlled vocabulary ID, and so on)

Title then has 5 possible subelements, which themselves can use 5 different attributes. I don’t think that you can break the subelements down further within titleInfo, but other elements do have more layers of nesting.



MODS EXAMPLE

titleInfo: (**type:** uniform, **ID:** 454839, **authorityURI:** id.loc.gov):

nonSort (lang: eng):A

title (**lang:** eng): House of Leaves

So here's what an actual record might look like.

We know that this is the uniform title, it has an id, that came from the controlled vocabulary authority at id.loc.gov. The title has a nonsorting word "A" which is english and then the rest of the title proper is "House of Leaves" which is also in english.

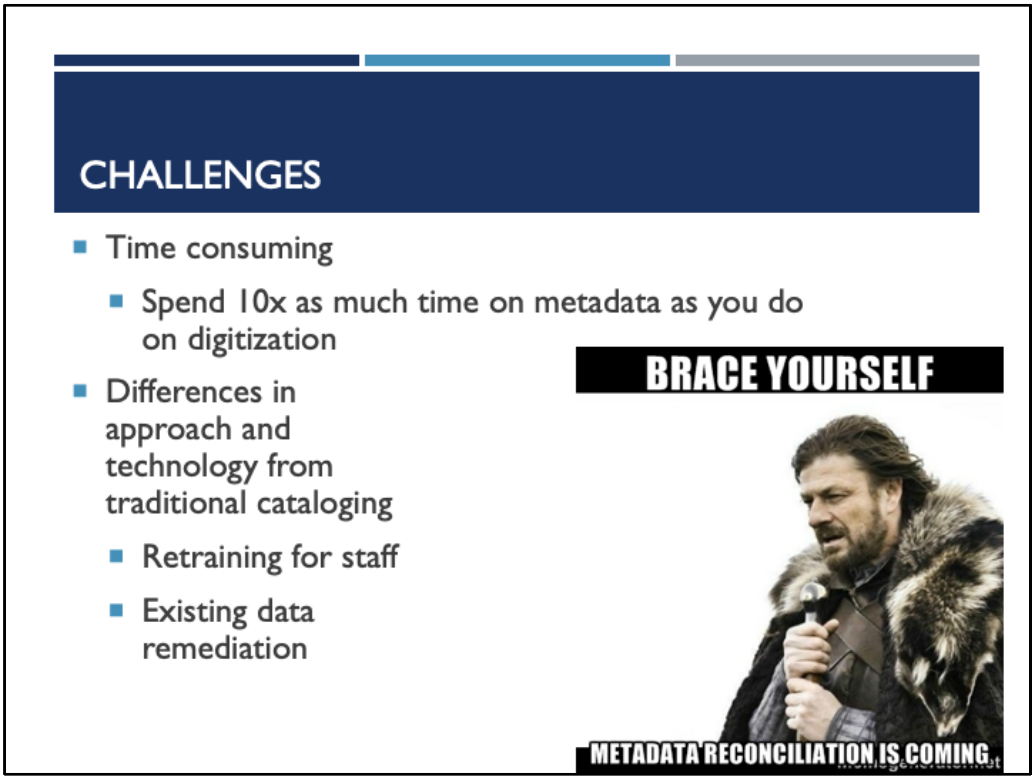
YAY GRANULARITY…

There are plenty of other standards we could talk about in the information science domain: schemas just for preservation and technical metadata, metadata wrappers, encoding standards, etc. But I don't want to overwhelm you with them today. I think the MODS and DC examples give you an idea of the breadth of what can be expressed anyway.

METADATA

HOW AWESOME IS IT?

So, we've learned a lot about metadata. And maybe it just sounds unbelievable awesome. It is, but….



CHALLENGES

- Time consuming
  - Spend 10x as much time on metadata as you do on digitization
- Differences in approach and technology from traditional cataloging
  - Retraining for staff
  - Existing data remediation

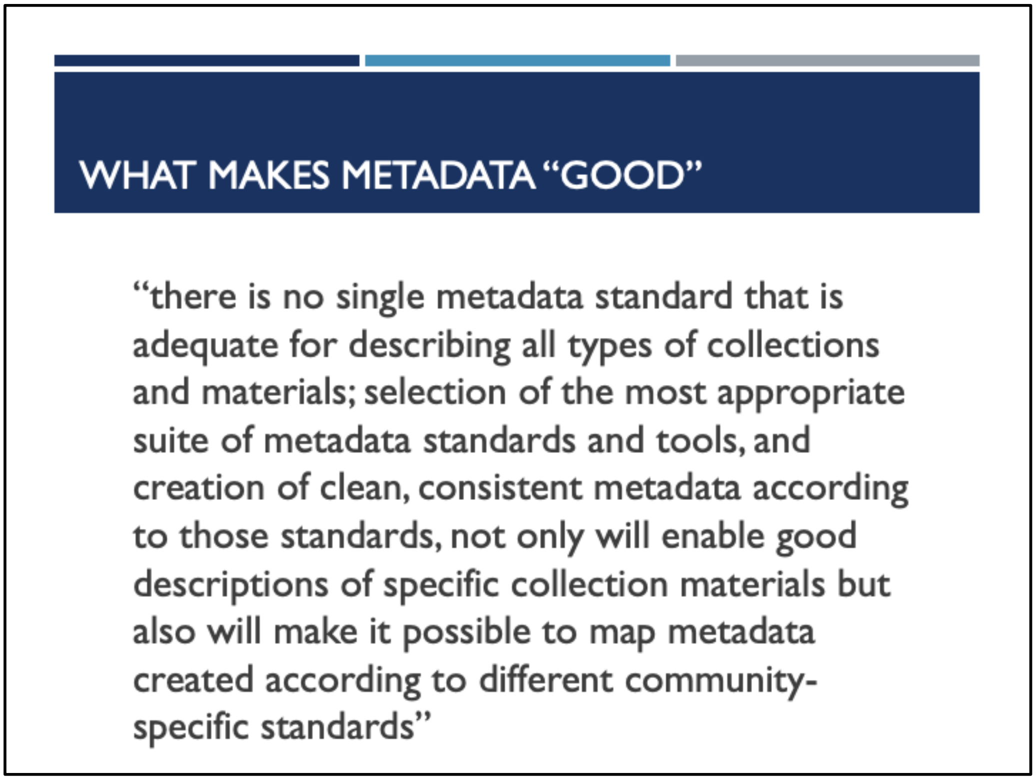
There are some challenges.

Metadata creation is going to be the most expensive part of your digitization workflow. It can easily take 10x as much time to create metadata as it does to scan something.

Creating metadata in the digital library environment also usually presents a paradigm shift for traditional catalogers, or at the very least retraining on tools. MARC was created for physical world --- remember weinberger's second order from our reading? It gives us a physical surrogate for where something is physically. It does maybe contain some elements of miscellaneous-ness – multiple subject headings, subject cards vs. author cards in a card catalog – but it is still limited (no more than 10 subject headings!). When we move towards environments that use more than just MARC and that can do more sophisticated search and retrieval, we have to think more broadly about how resources will be used and accessed than we did when just dealing with the catalog.

That said, you can re-use MARC data, and convert it to other metadata standards. That is sometimes called **metadata reconciliation** – the process of converting data in any structured format into another – something we'll talk about next week. MARC specifically is what a lot of libraries have in abundance, but really you could do this with any type of data (a spreadsheet could be converted into records as an example).

During the reconciliation process you will likely need to add data that the new schema required and the old didn’t. For example, if we were converting a record from simple Dublin Core to MODS, we may need to split up that coverage field into separate time and place elements. So there is almost always manual work involved in reconciliation. And again, when trying to convert MARC you run into a lot of obstacles based on the differences between tracking a physical item and a digital one.

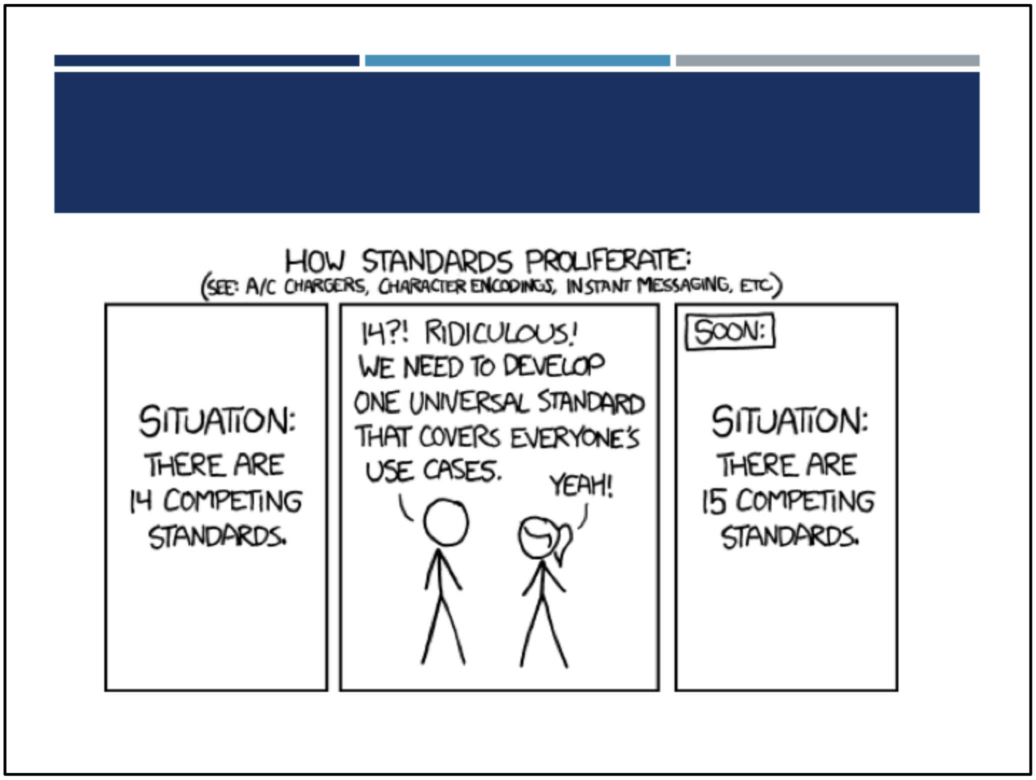


WHAT MAKES METADATA "GOOD"

"there is no single metadata standard that is adequate for describing all types of collections and materials; selection of the most appropriate suite of metadata standards and tools, and creation of clean, consistent metadata according to those standards, not only will enable good descriptions of specific collection materials but also will make it possible to map metadata created according to different community-specific standards"

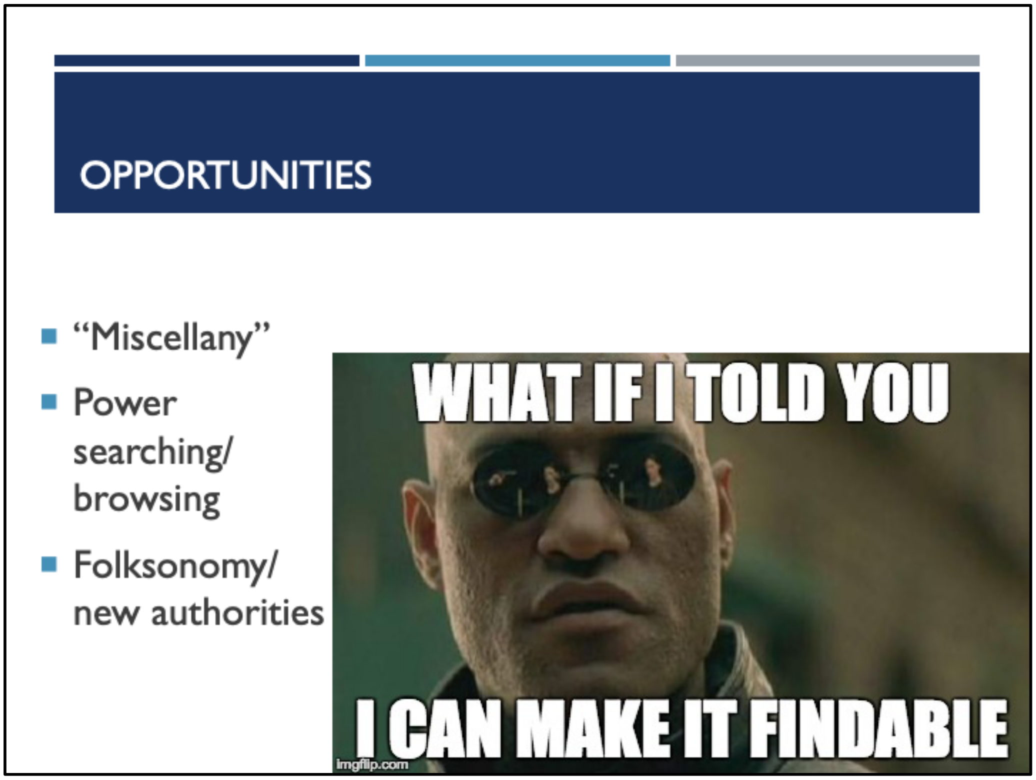
As with traditional MARC data though, the main challenges have to do with creating good data that serves it's purpose and using the type of metadata that fits situation and content.

This is from the Getty article by Baca and I think it is a good summary "there is no single metadata standard that is adequate for describing all types of collections and materials; selection of the most appropriate suite of metadata standards and tools, and creation of clean, consistent metadata according to those standards, not only will enable good descriptions of specific collection materials but also will make it possible to map metadata created according to different community-specific standards"


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

So part of our challenge is evaluating the situation and content and picking the right vehicle, or schema, for our metadata.

This cartoon is very appropriate for the situation.



OPPORTUNITIES

- "Miscellany"
- Power searching/ browsing
- Folksonomy/ new authorities

WHAT IF I TOLD YOU
I CAN MAKE IT FINDABLE
imgflip.com

On the other hand, metadata can really provide us with a lot opportunities if we use it well.

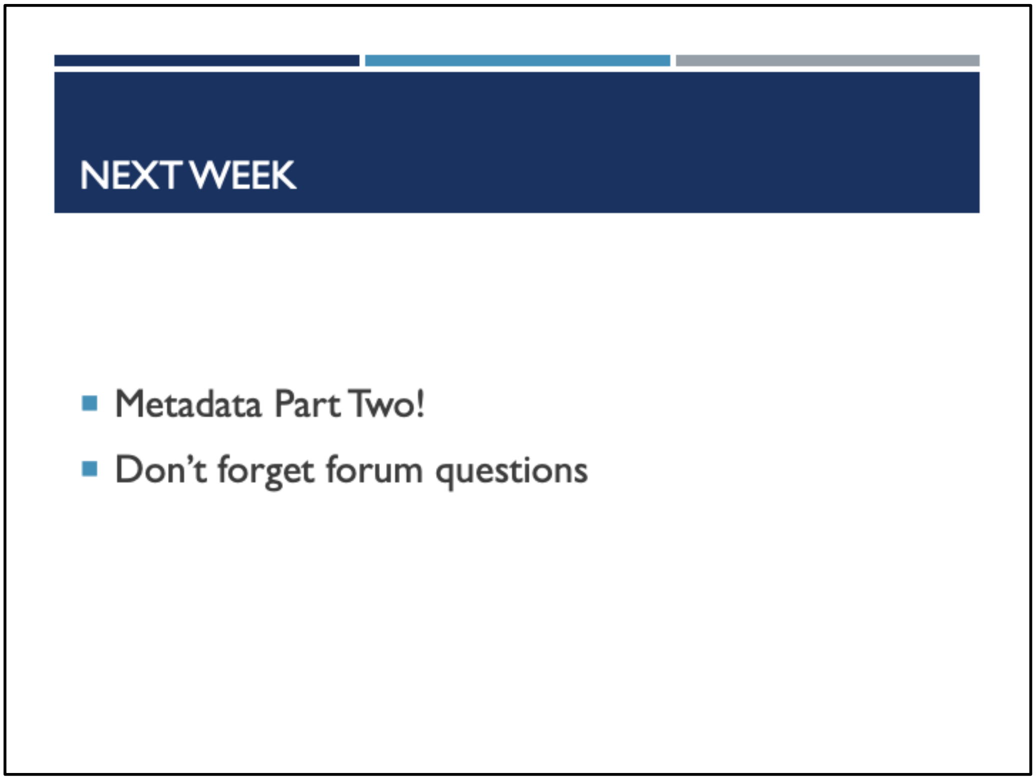
Above and beyond the mere fact that metadata is really the backbone of the information system (without it you can't find anything), I specifically had you read Weinberger's article about miscellany because I think it's a really great way of articulating the opportunities that flexible metadata in a computer-mediated system can afford.

It showed how metadata is different from traditional cataloging. While a lot of the concepts are the same (describing stuff), in the digital world we can index virtually every field in the record, meaning instead of having to find a work by it's title, author, or one of three subject headings, we could find it by date, or by keyword in the summary, or by the 11th subject. In a sense, nothing needs to be classified for search, because we can search every field equally if we choose to. And that's a powerful concept with both pros and cons.

Really granular metadata is what us to do really great and specific power searches and browses. As we showed with the facet example, that kind of ability to narrow your search down to just exactly what you want has been a hugely influential shift in online search and retrieval. That kind of stuff only exists with really good, granular metadata.

In addition, thinking of metadata as a broader term than cataloging, Weinberger introduces the idea of folksonomies (though he doesn't use the word). This is when descriptors are applied to objects by users. Over time, as these things are continually tagged with what you or I think is most important, a consensus emerges. A regular folks-driven taxonomy. Since the internet can allow the public to write back to the record, that folksonomy, as it is sometimes called, can be stored and exploited, which we couldn't do with a rigid paper-based catalog. This was a really big idea in libraries 10-15 years ago, I think we thought it would solve a lot of problems. It didn't. Primarily because it takes a lot of participation to make those kinds of systems really useful. However, the concept of breaking down authorities, using multiple sources of input, giving equal weight to various metata, those are still powerful and useful concepts.

Some would argue that those things degrade the traditional authority of information resources because it means there is no one source of bibliographic records anymore, no one truth. But in fact, that was the case all along, it's just more apparent now. New tools are allowing communities to create their own languages, their own tools, that we might never have used or recognized the importance of.



NEXT WEEK

- Metadata Part Two!
- Don’t forget forum questions

Next week we are going to talk more about these more advanced ways of using metadata and go a little beyond these fundamentals.

We’ll talk about how metadata is shared and what are some of the cool things we can do with it.

Next week will also introduce our first assignment. But until then, don’t forget your forum questions.