Metadata & Documentation

for

Data Curation

INFX 551

Spring 2017

- Report outs: Project updates; Repository Profile; Data Pitch (individual)

- Lecture and Discussion

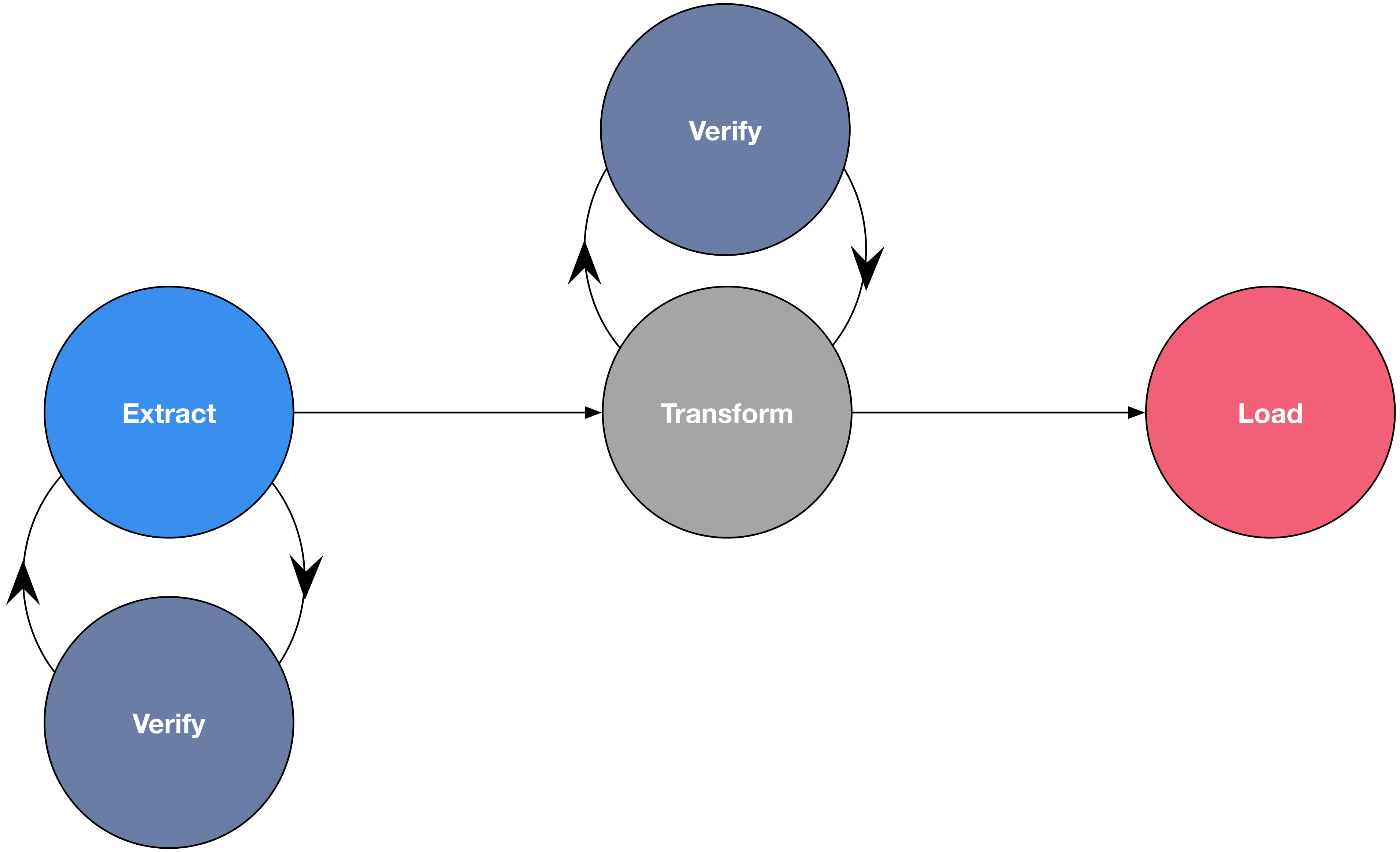
- Group Work: aUser Stories + Unstructured Metadata

- Sebastian Karcher Lecture and Discussion

- Some class notes:
    - Next weeks reading material will also be light.
    - Will be last of Module 2
    - Continue to split our focus between hands - on and conceptual
    - Directions for next steps of your group project will be posted Monday.

The goal of this lecture is disabuse you of the notion that —> “metadata is data about data” is an acceptable response the importance of metadata in doing data curation.

This week we focus on the edges, not the nodes.



Knowledge transfer depends on the formalization of tacit understandings…

Working and living in a material world (and we are material girls!) this requires us to make the ‘context’ of data production and use explicit…

Ok… what does that mean, really?

Knowledge Representation

(the five dollar term for ‘documentation and metadata’)

We said Ontology is a way of talking about what exists in the world…

In Information Science, we also use ontology to formally represent ‘things’ that exist in a domain, as well as how those things relate to other domains…

Ontological components

Instances - things (objects)

Classes - kinds of things

Sub-class & Super-Class - introduces hierarchy in kind structure

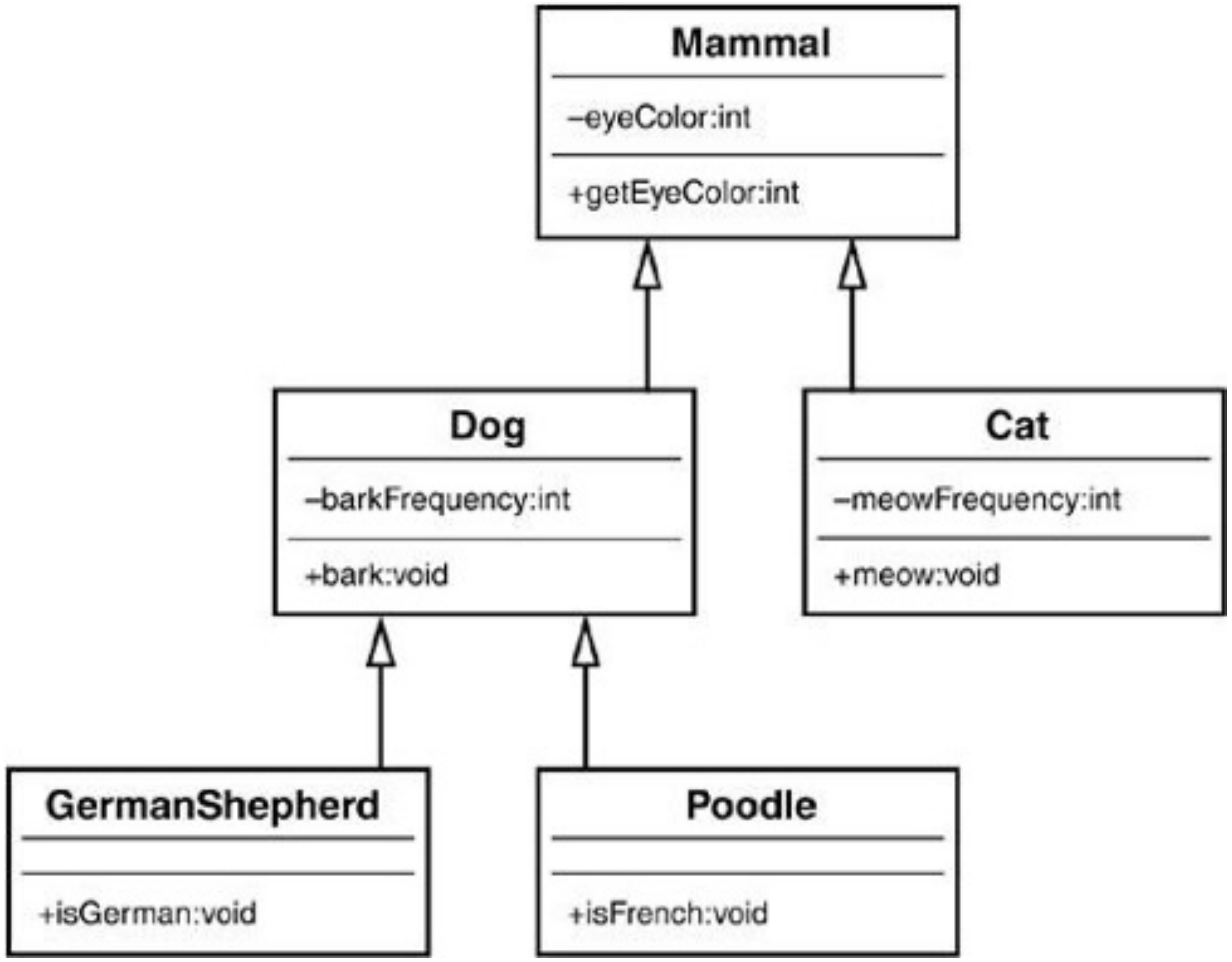
Attributes - properties, features, or characteristics of instances (and by inheritance, classes)

Relations - ways to link different instances or classes to one another

Class

Class

Class



Instance

Attributes - properties, features, or characteristics of instances (and by inheritance, classes)

Attribute: Value

Eye_Color: Blue

The Halting Problem of Knowledge Representation

Expressivity VS. Tractability

The more expressive we make a knowledge representation system, the less tractable it is in terms of generating, managing, and computing against the KR.

What does any of this have to do with metadata?

(In short, everything)

Three basic forms of metadata

Descriptive Metadata: Tells us about objects, their creation, and the context in which they were created (Title, Author, Date)

Technical Metadata: Tells us about the context of the data collection (Instrument, Computer, Algorithm)

Administrative Metadata: Tell us about the management of that data (Rights statements, Provenance, etc.)

Attributes - properties, features, or characteristics of instances (and by inheritance, classes)

Attribute: Value

Eye_Color: Blue

Descriptive metadata about an instance

But we can also have attributes of a class….

This introduces a distinction between item-level, and collection-level metadata.

(With a small collection you may benefit from creating both)

Attributes - properties, features, or characteristics of instances (and by inheritance, classes)

We can totally do better than this.

Attribute: Value

Eye_Color: Blue

Descriptive metadata about an instance

Encoding attribute-value pairs in machine readable formats.

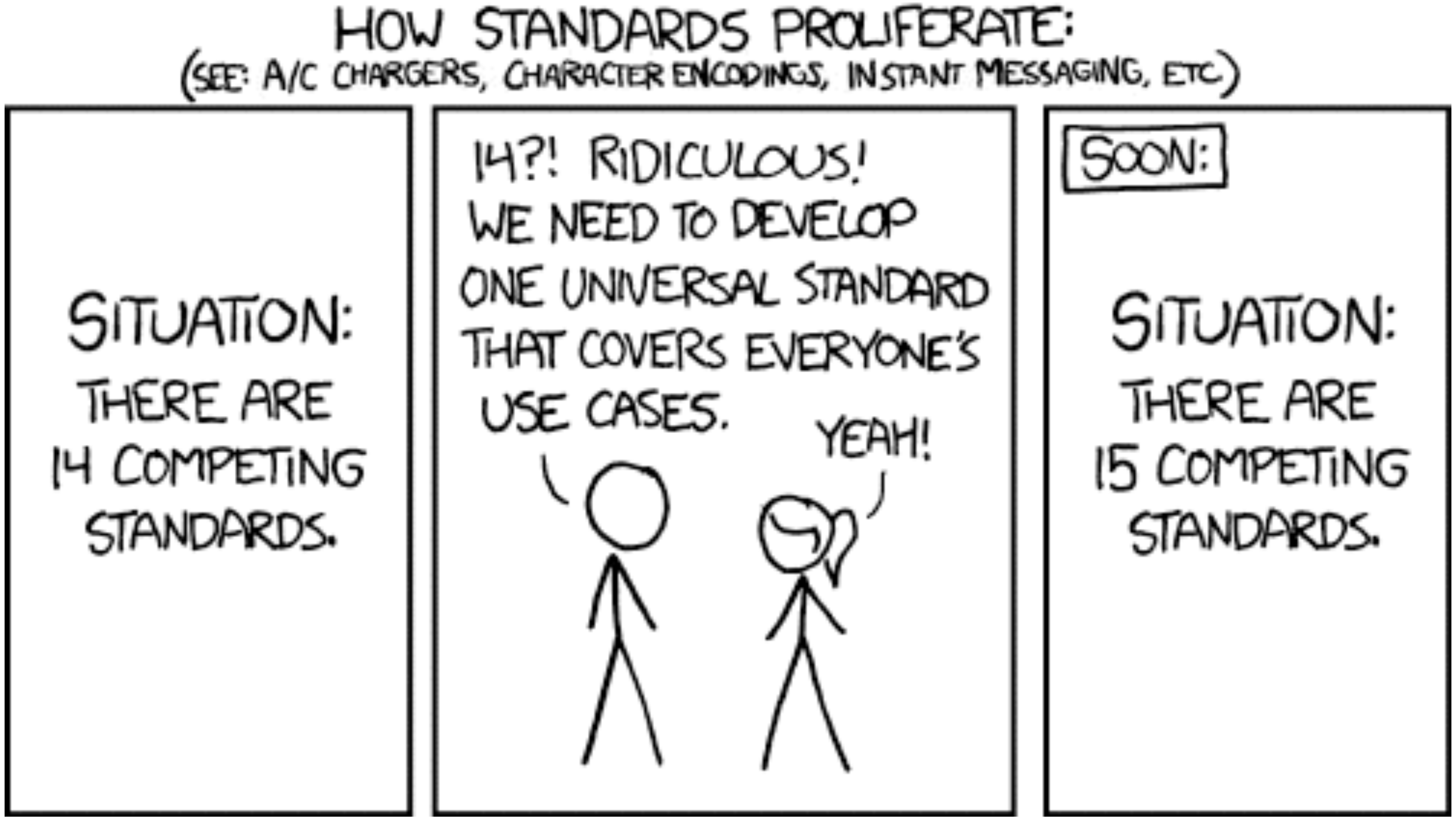
XML

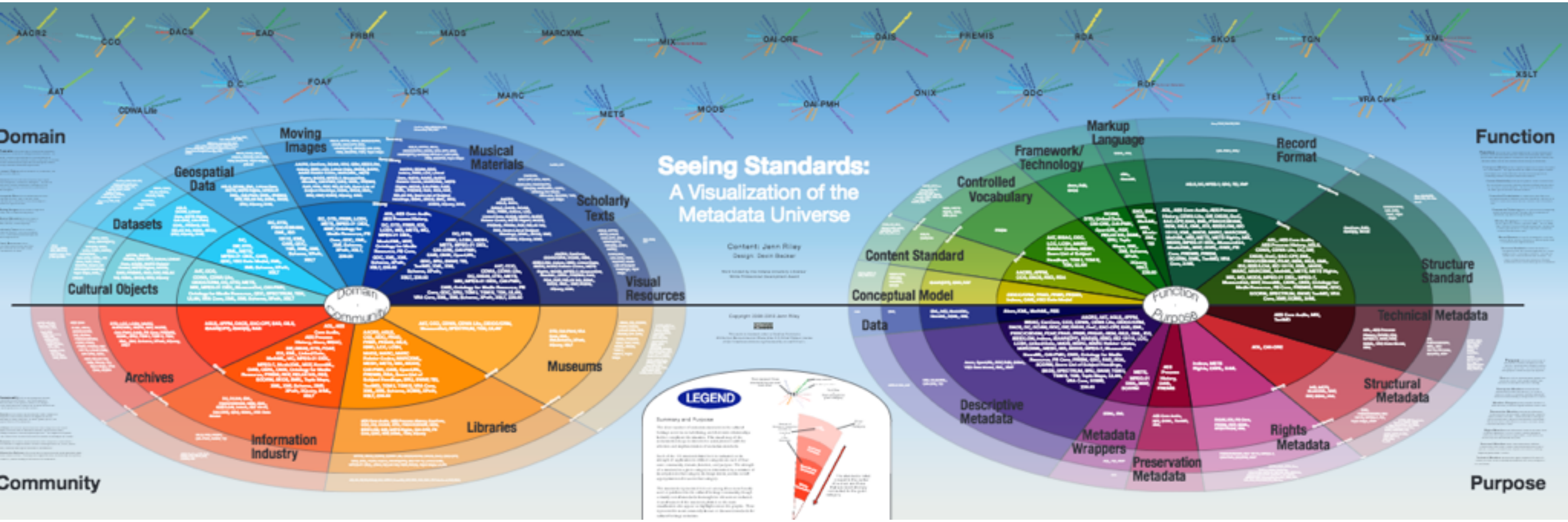
```
<eye_color>blue</eye_color>
```

JSON

```
{
  “eye_color": "blue",
}
```

Using standard metadata schemas


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.



Seeing Standards: A Visualization of the Metadata Universe

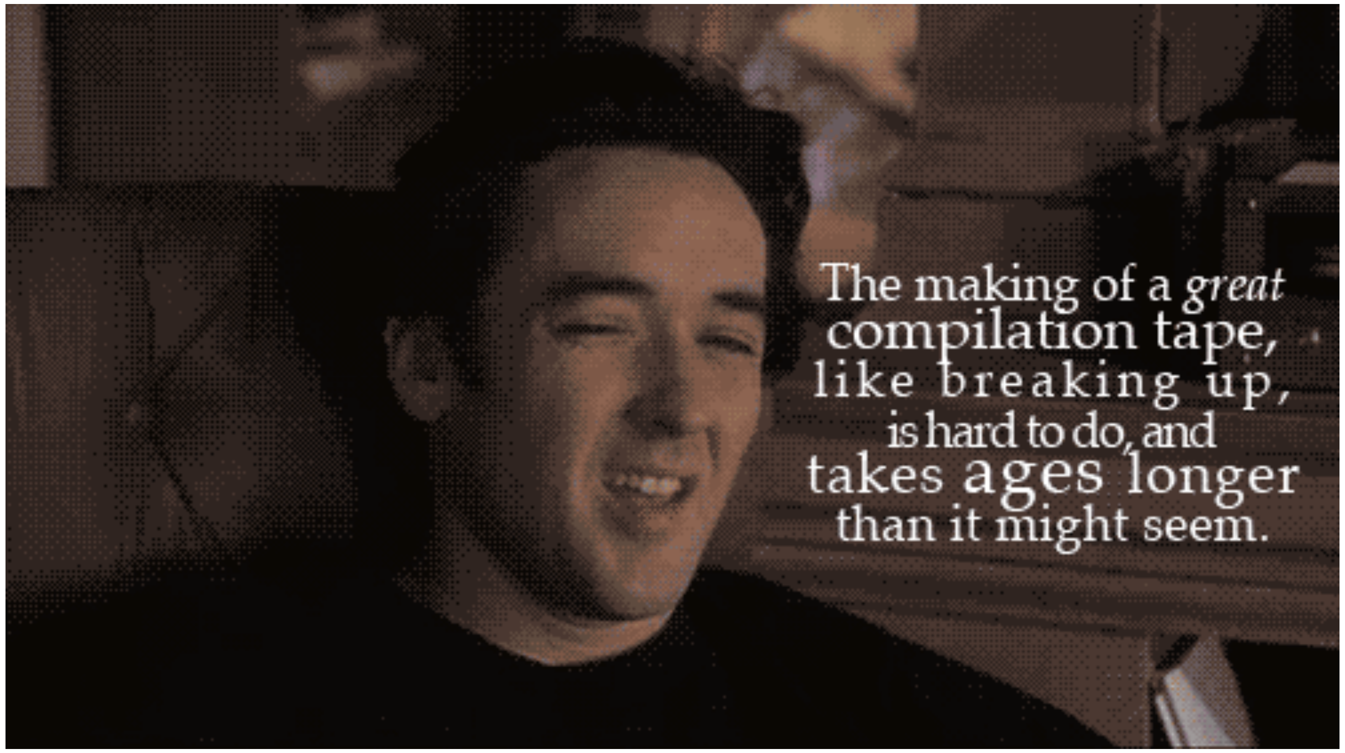
Content: Jenn Riley
Design: Devin Becker

In the header of our XML or JSON we declare the standards we want to use.

This provides a link to the specific standard as a web resource…

The set of standards, and individual attributes you combine may be uniquely defined - this is called a Schema (or application profile)

Think of a Schema like a mixtape. Individual tracks made by other people, but arranged by you to meet a particular purpose.



What makes a good schema?

*Avoids halting problem
*Is expressive and meaningful to your domain
*Modular architecture

What standards did you look at this week?

DCAT
POD
https://project-open-data.cio.gov/v1.1/schema/

Your next mission is to figure out what attributes (of your data) are meaningful to your domain (of users)

How do we know our domain?
Through requirements engineering and user stories (this week)

How do we know meaningful attributes?
Data profiles! (next week)

User Stories …

https://github.com/OpenDataLiteracy/INFX-551-Spring2017/wiki/Class-Activity:-User-Story

For next week post a write-up of this activity.

DCAT
POD
https://project-open-data.cio.gov/v1.1/schema/

Structured vs Unstructured Metadata

Documentation Varietals

(Unstructured Data)

README.txt
Data Dictionary
CodeBook

data.world preview

https://data.world/lilianhj/chicago-lobbyists