The Structure of Unstructured Data

Dr. Arshin Rezazadeh
CS 4417B/9117/9647
The University of Western Ontario

Thanks to

- Professors
  - Dan Lizotte and Hanan Lutfiyya

Today's Topics

- Data format standards associated with "unstructured data"
- Software format standards associated with "unstructured data"

“Structure”?

ID	First Name	Last Name	E-mail	Office	Full Time?	Hours/wk
024028	Dan	Lizotte	dlizotte	MC363	T	40
024814	Hanan	Lutfiyya	hanan	MC355e	T	40
025710	Cheryl	McGrath	cheryl	MC355a	F	20
015027	Janice	Wiersma	janice	MC355c	T	40

“Structure”?

- From Wikipedia “Unstructured Data”, Problems with the term:
  - Structure, while not formally defined, can still be implied.
  - Data with some form of structure may still be characterized as unstructured if its structure is not helpful for the processing task at hand.
  - Unstructured information might have some structure (semi-structured) or even be highly structured but in ways that are unanticipated or unannounced.

"Structure"?

- From OED:
  - 3. a. **a.** *fig.* The arrangement and organization of mutually connected and dependent elements in a system or construct.

- Compositional structure
  - What useful ways can data be divided into elements?

- Relational structure
  - What useful ways can elements be related to each other?

*Not intended to be mutually exclusive or exhaustive.
There may be lots of other useful ways to define structure!

Examples of structure

- What might be some of the compositional and relational structure of web pages?

- What might be some of the compositional and relational structure in google image search results?

Data Standards

- How do we describe the structure of data?

- Considerations for choosing a data standard
  - Flexibility
  - Interoperability
  - Efficiency

Standards: Data Formats XML

Use iClicker

Who is familiar with XML?

A) Yes
B) No

Standards: Data formats

Thunder and lightning. Enter three Witches

FIRST WITCH

When shall we three meet again
In thunder, lightning, or in rain?

SECOND WITCH

When the hurlyburly's done,
When the battle's lost and won.

XML

```
<?xml version="1.0" encoding="utf-8"?>
<EXCERPT>
<STAGEDIR>Thunder and lightning. Enter three
Witches</STAGEDIR>
<SPEECH>
<SPEAKER ROLE="WITCH">First Witch</SPEAKER>
<LINE>When shall we three meet again</LINE>
<LINE>In thunder, lightning, or in rain?</LINE>
</SPEECH>
<SPEECH>
<SPEAKER ROLE="WITCH">Second Witch</SPEAKER>
<LINE>When the hurlyburly’s done,</LINE>
<LINE>When the battle’s lost and won.</LINE>
</SPEECH>
</EXCERPT>
```

XML

```
<?xml version="1.0" encoding="utf-8"?>
<EXCERPT>
<STAGEDIR>Thunder and lightning. Enter three
Witches</STAGEDIR>
<SPEECH>
<SPEAKER ROLE="WITCH">First Witch</SPEAKER>
<LINE>When shall we three meet again</LINE>
<LINE>In thunder, lightning, or in rain?</LINE>
</SPEECH>
<SPEECH>
<SPEAKER ROLE="WITCH">Second Witch</SPEAKER>
<LINE>When the hurlyburly’s done,</LINE>
<LINE>When the battle’s lost and won.</LINE>
</SPEECH>
</EXCERPT>
```

What kind of compositional and relational structure does the XML tell us?

XML Rules

Legal Unicode

No syntax chars (e.g. <, &) in content. Use *entities*, e.g. <, &

Tags cannot overlap, e.g. <A><B></A></B>

Tags are case-sensitive

Tags cannot use special chars or start with digit

There is a single "root" element that contains all the other elements.

```
<?xml version="1.0" encoding="utf-8"?>
<EXCERPT>
<STAGEDIR>Thunder and lightning. Enter three
Witches</STAGEDIR>
<SPEECH>
<SPEAKER ROLE="WITCH">First Witch</SPEAKER>
<LINE>When shall we three meet again</LINE>
<LINE>In thunder, lightning, or in rain?</LINE>
</SPEECH>
<SPEECH>
<SPEAKER ROLE="WITCH">Second Witch</SPEAKER>
<LINE>When the hurlyburly's done,</LINE>
<LINE>When the battle's lost and won.</LINE>
</SPEECH>
</EXCERPT>
```

XML parsing

- XML parsing is the process of converting XML data from its serialized string format to its hierarchical format. (https://www.ibm.com/docs/en/db2/10.5?topic=data-xml-parsing)
- A tree

Parsed XML: A tree

- <EXCERPT>
  - <STAGEDIR>
  - <SPEECH>
    - <SPEAKER ROLE = "WITCH">
    - <LINE>
    - <LINE>
  - <SPEECH>
    - <SPEAKER ROLE = "WITCH">
    - <LINE>
    - <LINE>

XML’s evolution

- Primary data format used by IBM
- Used by MS Office as a file format
- HTML5 is valid XML

Questions about XML

- Flexibility
  - What kinds of structure can XML easily describe?
  - What kinds of structure are not easy to describe with XML?
- Interoperability
  - What kinds of interoperability does XML support?
- Efficiency
  - What kinds of operations on data are efficient if XML is used as the data format? What kinds of operations are not efficient?

Standards: Data Formats JSON

Use iClicker

Who is familiar with JSON?

A) Yes, I have prior experience
B) Yes, a little
C) No

JSON (JavaScript Object Notation)

- JavaScript arrays:
  - `var myarray = ["one", "two", "three"];`
  - Then `myarray[1] == "two"`

- JavaScript objects
  - `var myobject = {firstName:"John", lastName:"Doe"};`

  - Then `myobject.firstName == "John"`

JSON (Javascript Object Notation)

```
{
 "EXCERPT": {
  "STAGEDIR": "Thunder and lightning. Enter three Witches",
  "SPEECH": [
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "First Witch"
     },
     "LINE": [
            "When shall we three meet again",
            "In thunder, lightning, or in rain?"
            ]
   },
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "Second Witch"
     },
     "LINE": [
            "When the hurlyburly's done,",
            "When the battle's lost and won."
            ]
   }
  ]
 }
}
```

JSON (Javascript Object Notation)

```
{
 "EXCERPT": {
  "STAGEDIR": "Thunder and lightning. Enter three Witches",
  "SPEECH": [
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "First Witch"
     },
     "LINE": [
            "When shall we three meet again",
            "In thunder, lightning, or in rain?"
            ]
   },
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "Second Witch"
     },
     "LINE": [
            "When the hurlyburly's done,",
            "When the battle's lost and won."
            ]
   }
  ]
 }
}
```

What is the value of `myobject.EXCERPT.SPEECH[1].LINE[0]`?

JSON (Javascript Object Notation)

```
{
 "EXCERPT": {
  "STAGEDIR": "Thunder and lightning. Enter three Witches",
  "SPEECH": [
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "First Witch"
     },
     "LINE": [
            "When shall we three meet again",
            "In thunder, lightning, or in rain?"
            ]
   },
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "Second Witch"
     },
     "LINE": [
            "When the hurlyburly's done,",
            "When the battle's lost and won."
            ]
   }
  ]
 }
}
```

What is the value of `myobject.EXCERPT.SPEECH[1].LINE[0]`?

JSON (Javascript Object Notation)

```
{
 "EXCERPT": {
  "STAGEDIR": "Thunder and lightning. Enter three Witches",
  "SPEECH": [
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "First Witch"
     },
     "LINE": [
            "When shall we three meet again",
            "In thunder, lightning, or in rain?"
            ]
   },
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "Second Witch"
     },
     "LINE": [
            "When the hurlyburly's done,",
            "When the battle's lost and won."
            ]
   }
  ]
 }
}
```

← How would we reference this value?

JSON (Javascript Object Notation)

```
{
 "EXCERPT": {
  "STAGEDIR": "Thunder and lightning. Enter three Witches",
  "SPEECH": [
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "First Witch"
     },
     "LINE": [
            "When shall we three meet again",
            "In thunder, lightning, or in rain?"
            ]
   },
   { "SPEAKER": {
       "ROLE": "WITCH",
       "text": "Second Witch"
     },
     "LINE": [
            "When the hurlyburly's done,",
            "When the battle's lost and won."
            ]
   }
  ]
 }
}
```

← How would we reference this value?

myobject.EXCERPT.SPEECH[0].SPEAKER.ROLE

JSON's evolution

- Developed for browser ←→ web server communication of JavaScript objects
- Has become widespread
  - Twitter
  - Reddit
  - …

Questions about JSON

- Flexibility
  - What kinds of structure can JSON easily describe?
  - What kinds of structure are not easy to describe with JSON?
- Interoperability
  - What kinds of interoperability does JSON support?
- Efficiency
  - What kinds of operations on data are efficient if JSON is used as the data format? What kinds of operations are not efficient?

XML versus JSON

- What do you think are the main differences between using XML versus using JSON?

- What are the consequences for choosing one over the other?

Data with no standards: system logs

- Jan 11 09:21:14 JRs-MacBook-Pro-2 syslogd[56]: ASL Sender Statistics
- Jan 11 09:21:24 JRs-MacBook-Pro-2 com.apple.xpc.launchd[1] (com.apple.preference.displays.MirrorDisplays): Service only ran for 0 seconds. Pushing respawn out by 10 seconds.
- Jan 11 09:21:38 --- last message repeated 1 time ---
- Jan 11 09:21:38 JRs-MacBook-Pro-2 filecoordinationd[351]: BUG in libdispatch client: kevent[mach_recv] monitored resource vanished before the source cancel handler was invoked
- Jan 11 09:21:44 JRs-MacBook-Pro-2 com.apple.xpc.launchd[1] (com.apple.preference.displays.MirrorDisplays): Service only ran for 0 seconds. Pushing respawn out by 10 seconds.
- Jan 11 09:22:14 --- last message repeated 2 times ---
- Jan 11 09:22:14 JRs-MacBook-Pro-2 com.apple.xpc.launchd[1] (com.apple.preference.displays.MirrorDisplays): Service only ran for 0 seconds. Pushing respawn out by 10 seconds.
- Jan 11 09:22:40 --- last message repeated 2 times ---
- Jan 11 09:22:40 JRs-MacBook-Pro-2 mdworker[811]: mdworker(811,0x700005a2f000) malloc: malloc_memory_event_handler: approaching memory limit. Starting stack-logging.
- Jan 11 09:22:40 JRs-MacBook-Pro-2 mdworker[811]: mdworker(811,0x700005a2f000) malloc: recording malloc (and VM allocation) stacks using lite mode
- Jan 11 09:22:42 JRs-MacBook-Pro-2 mdworker[811]: mdworker(811,0x700005a2f000) malloc: malloc_memory_event_handler: stopping stack-logging
- Jan 11 09:22:42 JRs-MacBook-Pro-2 mdworker[811]: mdworker(811,0x700005a2f000) malloc: turning off recording malloc (but not VM allocation) stacks using lite mode

Standards: Software Components

Standards: Software components

- UIMA – Unstructured Information Management Architecture
  - “Yoo-ee-mah”

- **Software architecture standard**
  - Specifies component interfaces in an analytics pipeline
  - If you write UIMA-compliant software, it can work with other UIMA-compliant software. Like IBM Watson.
  - It suggests two data representations: in-memory for **processing**, XML-based for **communicating**

Unstructured Information Management Applications

- "An unstructured information management (UIM) application may be generally characterized as a software system that analyzes large volumes of unstructured information (text, audio, video, images, etc.) to discover, organize and deliver relevant knowledge to the client or application end-user. An example is an application that processes millions of medical abstracts to discover critical drug interactions. Another example is an application that processes tens of millions of documents to discover key evidence indicating probable competitive threats."
- http://docs.oasis-open.org/uima/v1.0/uima-v1.0.html

UIMA's Definition of Unstructured Data

- "Unstructured information may be defined as the **direct product of human communication**. Examples include natural language documents, email, speech, images and video. It is information that was **not specifically encoded for machines** to process but rather authored by humans for humans to understand. We say it is "unstructured" because it lacks explicit semantics ("structure") required for applications to interpret the information as intended by the human author or required by the end-user application. "

- http://docs.oasis-open.org/uima/v1.0/uima-v1.0.html

UIMA Definitions

- artifact – a segment of unstructured content (e.g., a document, a video etc.)
- analysis – act of assigning semantics to a region of an artifact
- analytic (or Analysis Engine [AE]) – software component or service that performs the analysis
- artifact metadata – results of the analysis of an artifact by an analytic

UIMA Analytics

- Analytics are typically **reused** and **combined** to perform application-specific analyses
- For example:
  - first analytic identifies distinct words in a document
  - second analytic identifies parts of speech (verb, noun, …)
  - third analytic uses the output of the previous two to identify instances of persons, organizations and the relationships between them

Analytics bridge the Unstructured & Structured worlds

Unstructured Information

Text, Chat, Email, Audio, Video

UIMA

- Identify Semantic Entities, Induce Structure
- Chats, Phone Calls, Transfers
- People, Places, Org, Events
- Times, Topics, Opinions, Relationships
- Threats, Plots, etc.

Structured Information

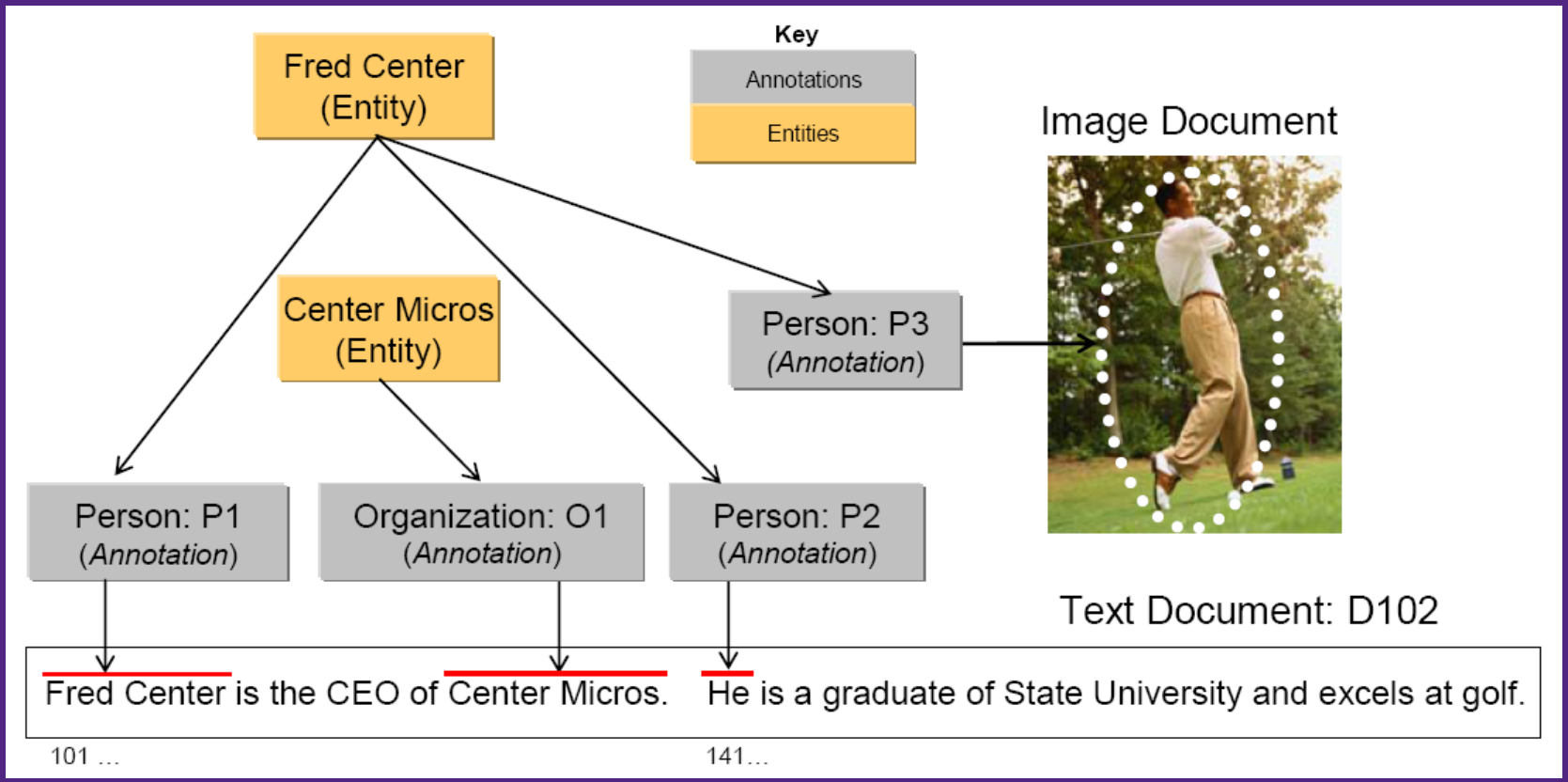
Indices

DBs

KBs

High-Value
Most Current Content
BUT ...
Buried in Huge Volumes
Lots of Noise, Implicit Semantics
Inefficient Search

Explicit Structure
Explicit Semantics
Efficient Search
Focused Content



UIMA idefines building blocks called **Analysis Engines** (AEs).
One way to think about AEs is as software agents that automatically discover and record *meta-data* about original content, e.g.:

(1) The Topic of document D102 is "CEOs and Golf".
(2) The span from position 101 to 112 in document D102 denotes a Person
(3) The Person denoted by span 101 to 112 and the Person denoted by span 141 to 143 in document D102 refer to the same Entity.

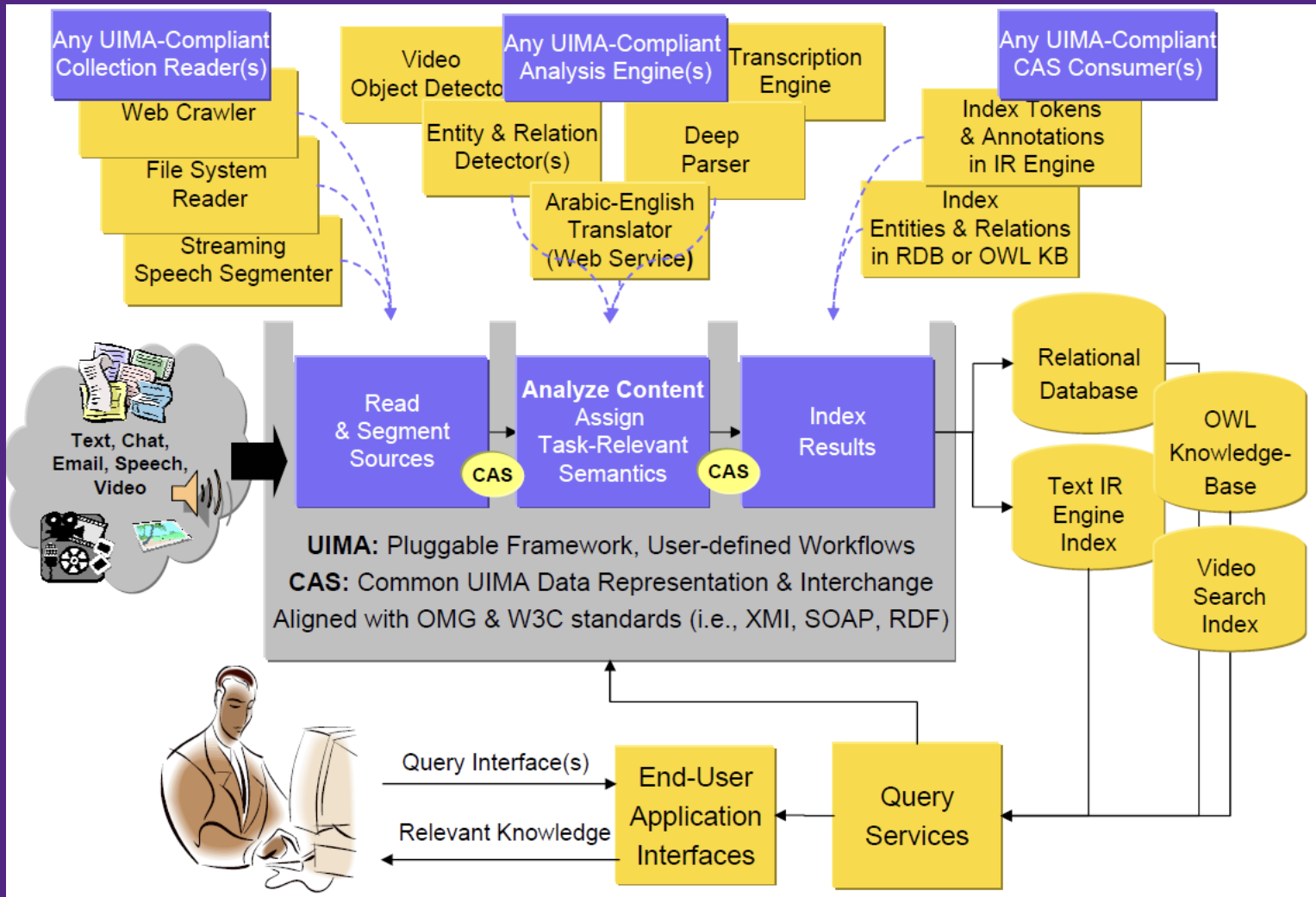
UIMA-based systems

- IBM Research's Watson uses UIMA for standardizing how it processes data.
- The Clinical Text Analysis and Knowledge Extraction System (Apache cTAKES) is a UIMA-based system for information extraction from medical records.
- DKPro Core is a collection of reusable UIMA components for general-purpose natural language processing.

UIMA Implementation

- Common Analysis Structure (CAS): Container for Data Structures in user-defined data model
  - (which can be defined in UML)
- Annotator (analysis engine): Pluggable component (Java or C++, among others) that reads and writes a CAS
- Aggregate Analysis Engine: Collection of Annotators

- The Common Analysis Structure (CAS) is an object-based data structure. It logically contains the documents to be analyzed. Analysis engines present and share their results in a CAS. With CAS, you can represent objects, properties, and values. (https://www.ibm.com/docs/en/db2/10.1.0?topic=concepts-common-analysis-structure)


Any UIMA-Compliant Collection Reader(s)
Web Crawler
File System Reader
Streaming Speech Segmenter
Video Object Detecto
Any UIMA-Compliant Analysis Engine(s)
Transcription Engine
Entity & Relation Detector(s)
Deep Parser
Arabic-English Translator (Web Service)
Any UIMA-Compliant CAS Consumer(s)
Index Tokens & Annotations in IR Engine
Index Entities & Relations in RDB or OWL KB
Text, Chat, Email, Speech, Video
Read & Segment Sources
CAS
Analyze Content Assign Task-Relevant Semantics
CAS
Index Results
Relational Database
OWL Knowledge-Base
Text IR Engine Index
Video Search Index
UIMA: Pluggable Framework, User-defined Workflows
CAS: Common UIMA Data Representation & Interchange
Aligned with OMG & W3C standards (i.e., XMI, SOAP, RDF)
Query Interface(s)
Relevant Knowledge
End-User Application Interfaces
Query Services
T-FaNT 07
© 2007 IBM Corporation – All Rights Reserved –

Why a standard?

- EXAMPLE: find all telephone numbers in running text
  Regular expression: [0-9]{3} -? [0-9]{4}
- Idea: in-line annotations, e.g., modify text
  - "Call Jenny; her number is *PHONE*867-5309*PHONE*."
- Gets very messy very quickly:
  - "*VERB*Call*VERB**NOUN**PERSON*Jenny*PERSON*NOUN*; *PRONOUN*her*PRONOUN* *NOUN**SUBJECT*number*SUBJECT**NOUN* is *PHONE*867-5309*PHONE*."
- Makes the next component of the pipeline's job a total pain; discourages software re-use.

Example from Paul Fodor, SUNY

UIMA Idea: "Standoff Annotations"

- Keep original text
- Add annotations with associated offsets in the original text

- "Call Jenny; her number is 867-5309."

- Phone number annotation: (27,34)

UIMA Idea: "Type Systems"

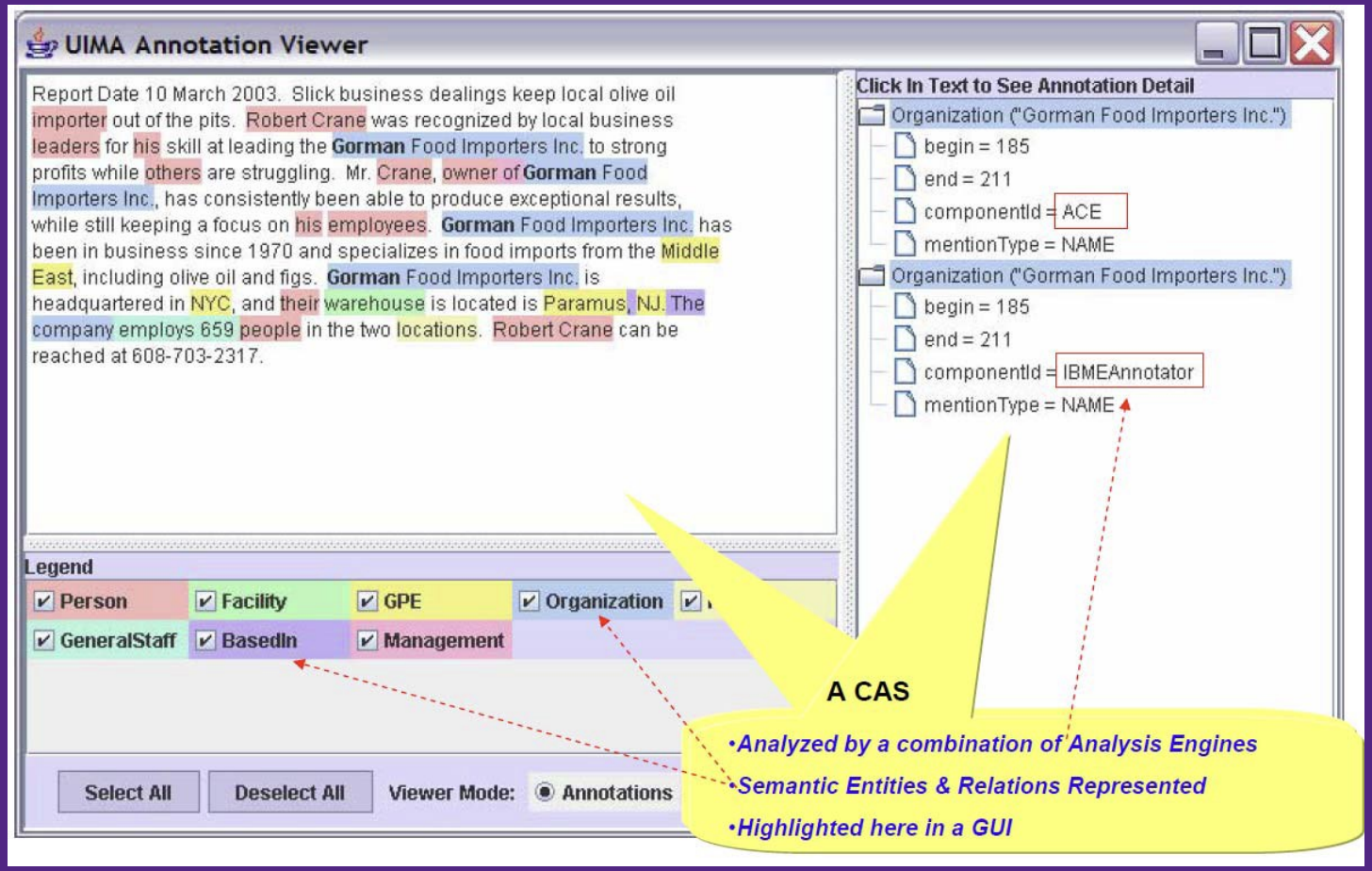
```
…<boilerplate>…
<typeDescription>
<name>org.apache.uima.tutorial.RoomNumber</name>
<supertypeName>uima.tcas.Annotation</supertypeName>
<features>
   <featureDescription>
      <name>building</name>
      <description>Building containing this room</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
   </featureDescription>
</features>
</typeDescription>
…<boilerplate>
```

Example Analysis Engine Code

```
private Pattern myPattern = Pattern.compile("\\b[0-4]\\d-[0-2]\\d\\d\\b");
public void process(JCas aJCas) { //Pass in Common Analysis Structure
        // get document text
        String docText = aJCas.getDocumentText();
        // search for room numbers
        Matcher matcher = myPattern.matcher(docText);
        int pos = 0;
        while (matcher.find(pos)) {
                // found one - create annotation
                RoomNumber annotation = new RoomNumber(aJCas);
                annotation.setBegin(matcher.start());
                annotation.setEnd(matcher.end());
                annotation.setBuilding("Yorktown");
                annotation.addToIndexes();
                pos = matcher.end();
        }
}
```

Optional: https://uima.apache.org/d/uimaj-current/tutorials_and_users_guides.html

Example Annotations



Watson on Jeopardy!

https://www.youtube.com/watch?v=P18EdAKuC1U

UIMA Questions

- Can you describe an application that we haven't discussed that would benefit from a UIMA architecture?

- Can you think of an application for which UIMA would *not* be a good choice of software standard? Why not?

Summary

- Types of structure
  - Compositional and relational
- Formats for unstructured data
  - XML
  - JSON
- Software standards
  - UIMA; "annotation-centred" view of unstructured data analysis