Haystack

A Platform for Creating, Visualizing, and Organizing Information Using RDF

Dennis Quan (dquan@media.mit.edu)

David Huynh (dfhuynh@ai.mit.edu)

http://haystack.lcs.mit.edu/

Outline

- Motivation
- Demo

- System Architecture
- Language for Manipulating RDF
- User Interface Paradigm

Motivation

- Use Semantic Web technologies in personal information management domain?

- See all important things on one screen
- Send e-mail to ourselves
- Overflowing inbox

Folders?

- Out of sight, out of mind
- Which folder?
- Search

Root Problem

- People and computers think about information in different ways

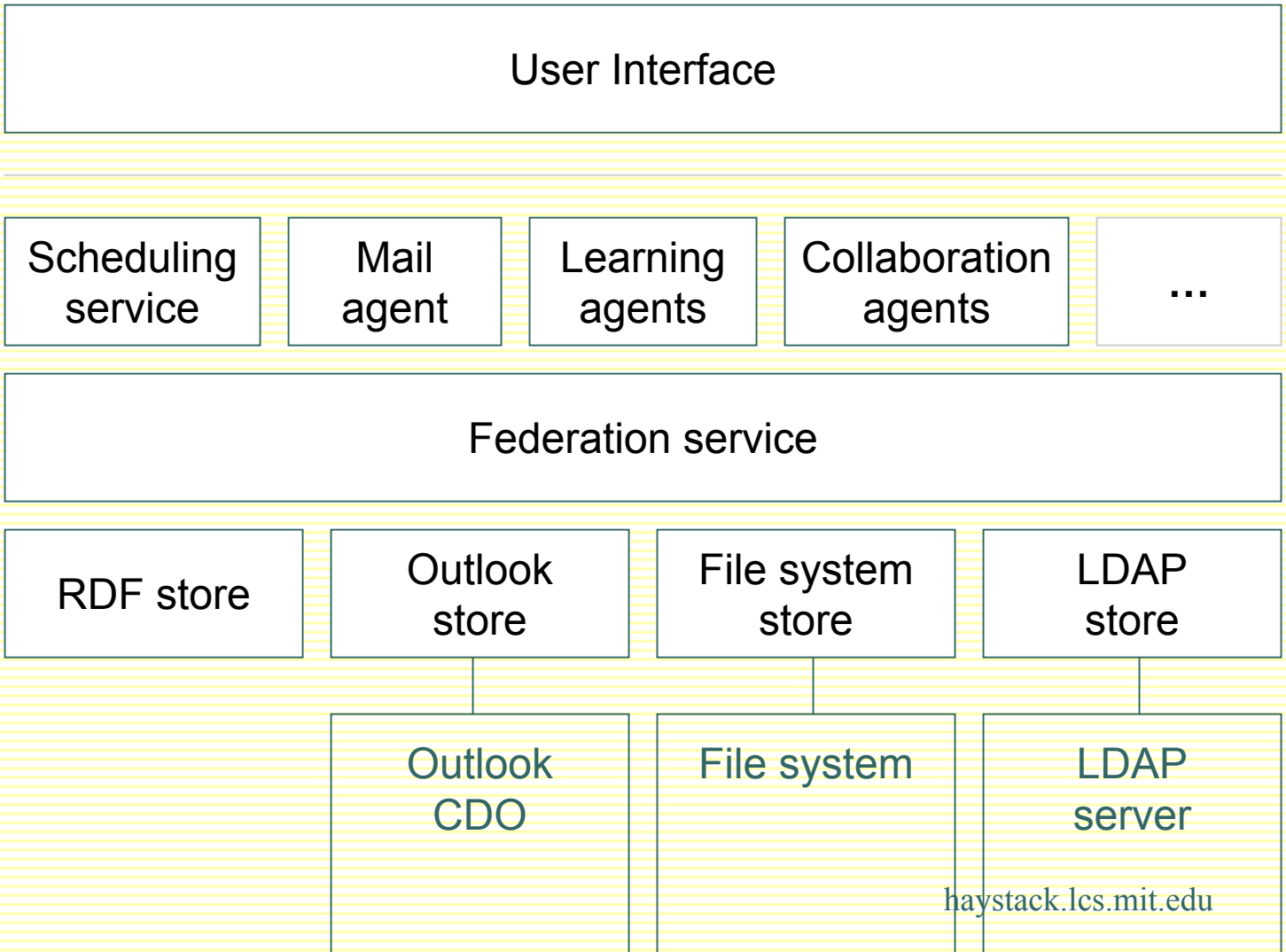
Solution

- Add semantics to bridge the gap
- Model all data in RDF

Demo

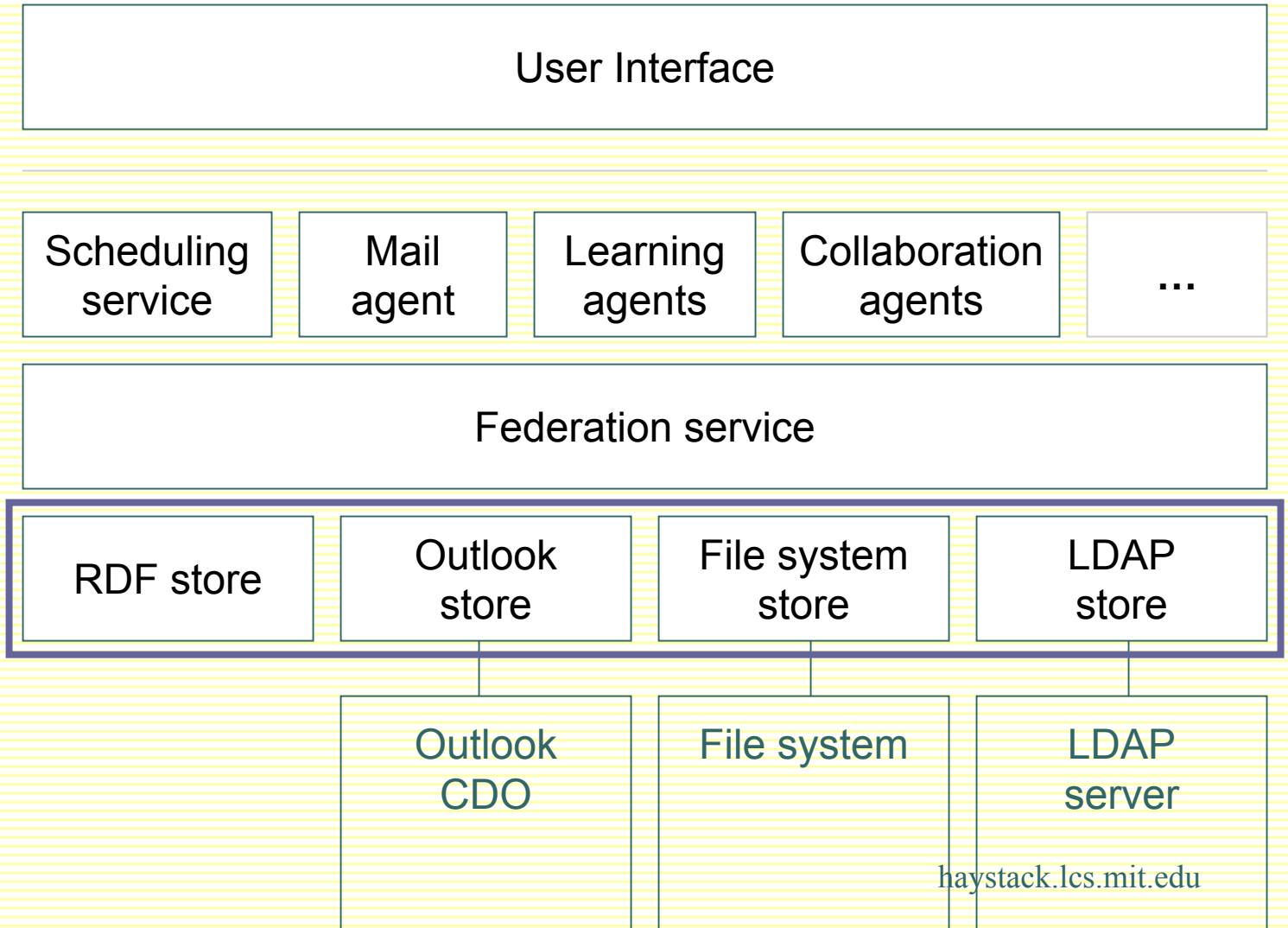
haystack.lcs.mit.edu

System Architecture



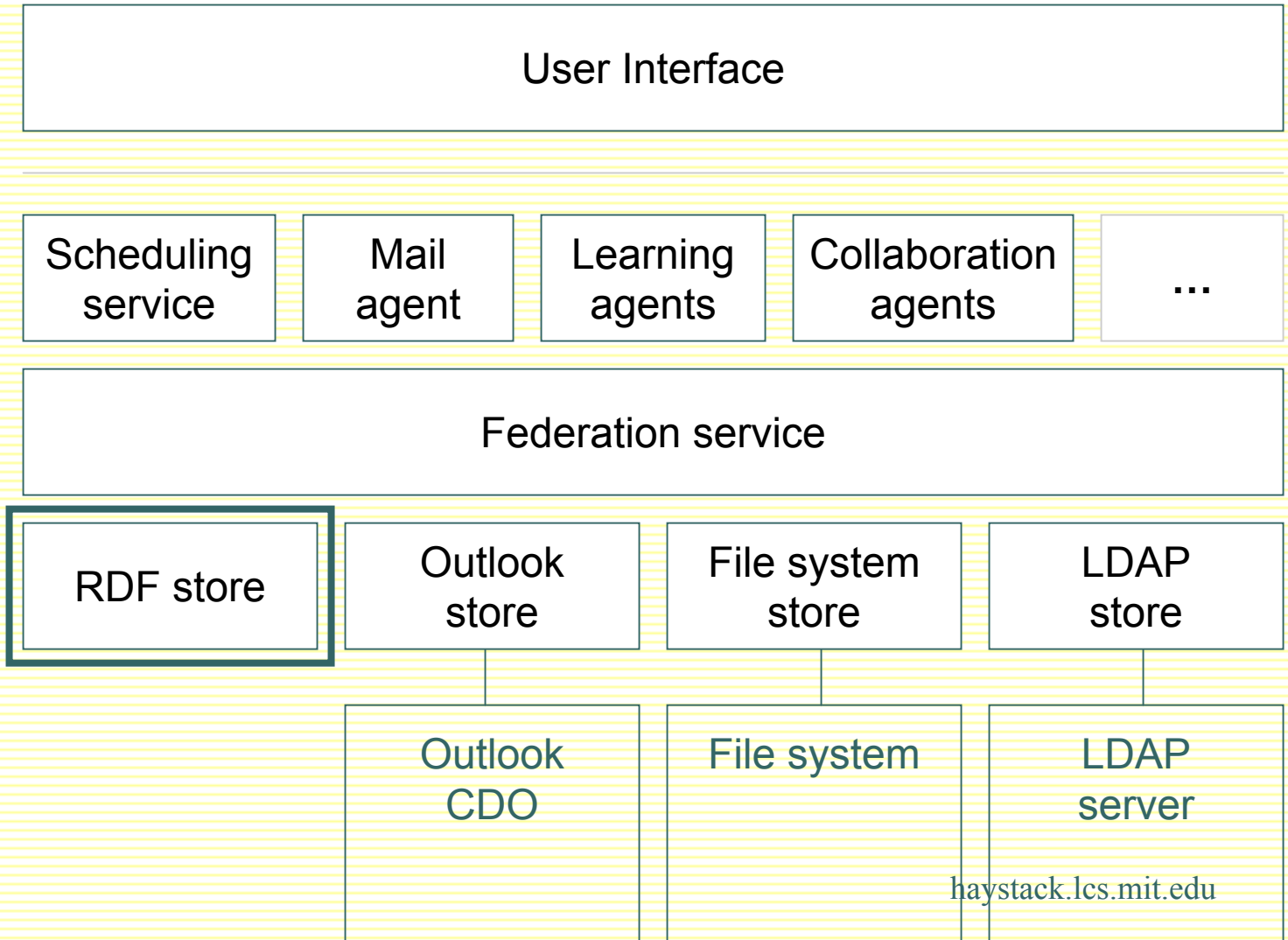
haystack.lcs.mit.edu

System Architecture

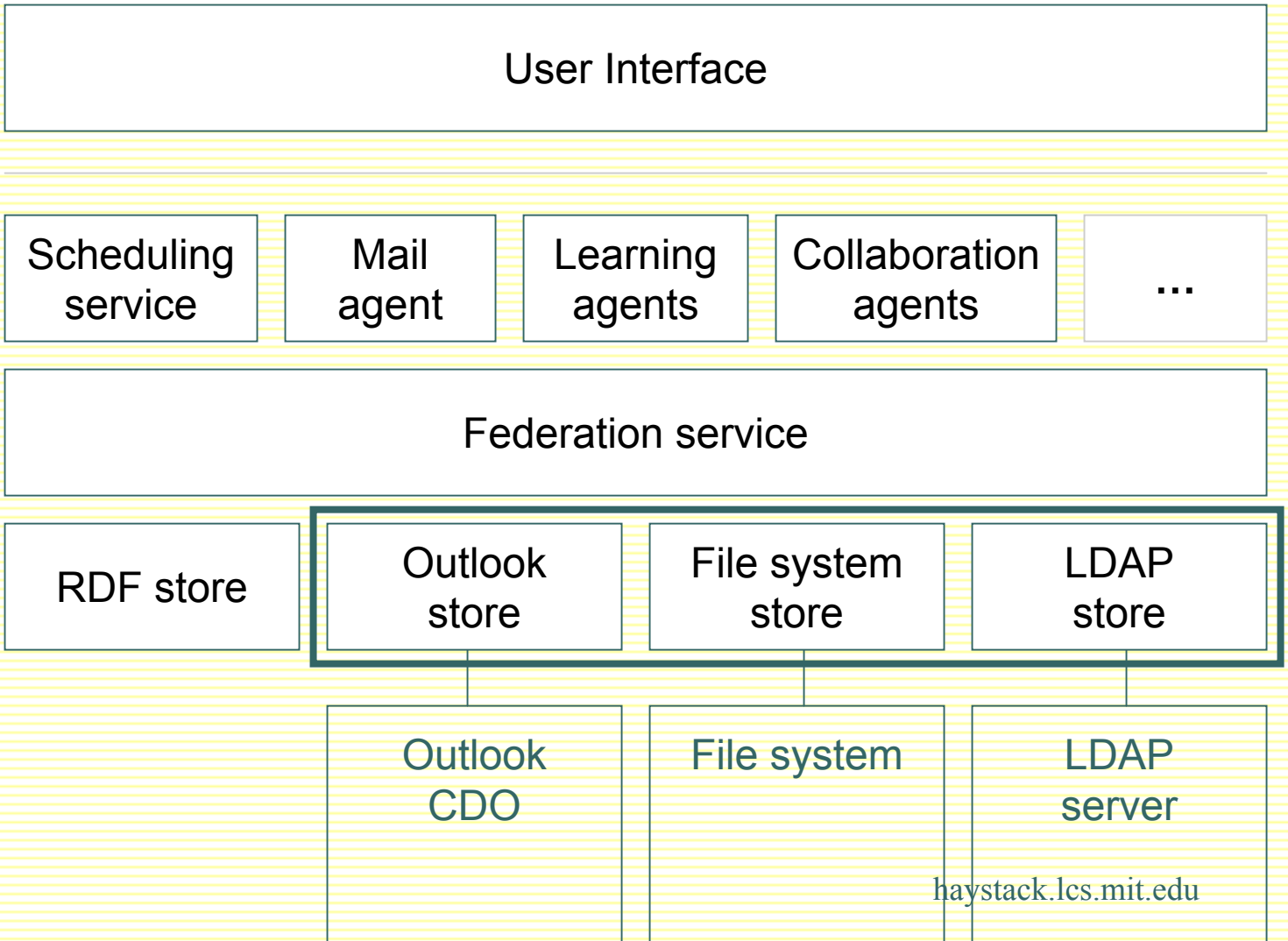


haystack.lcs.mit.edu

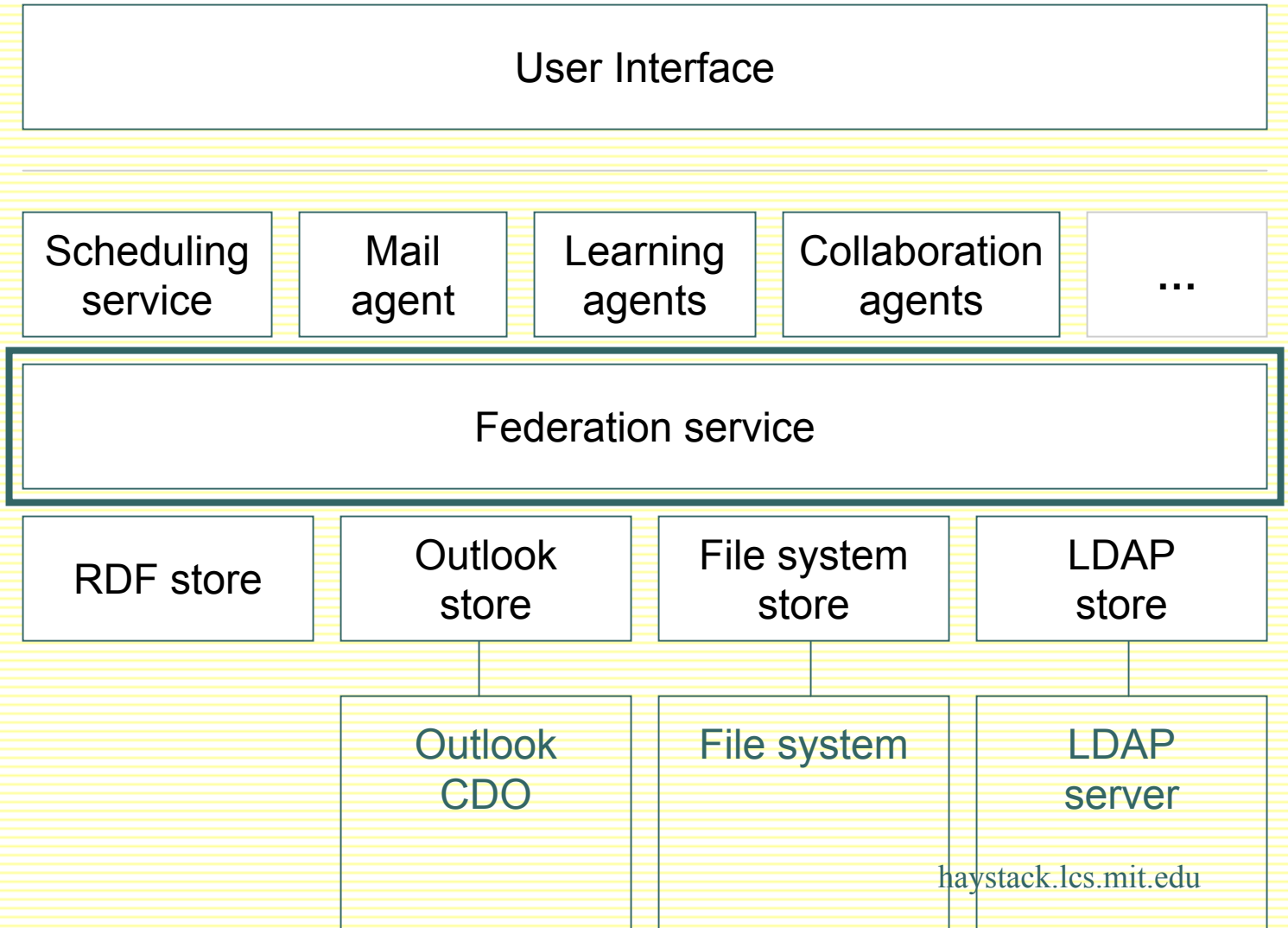
System Architecture



System Architecture

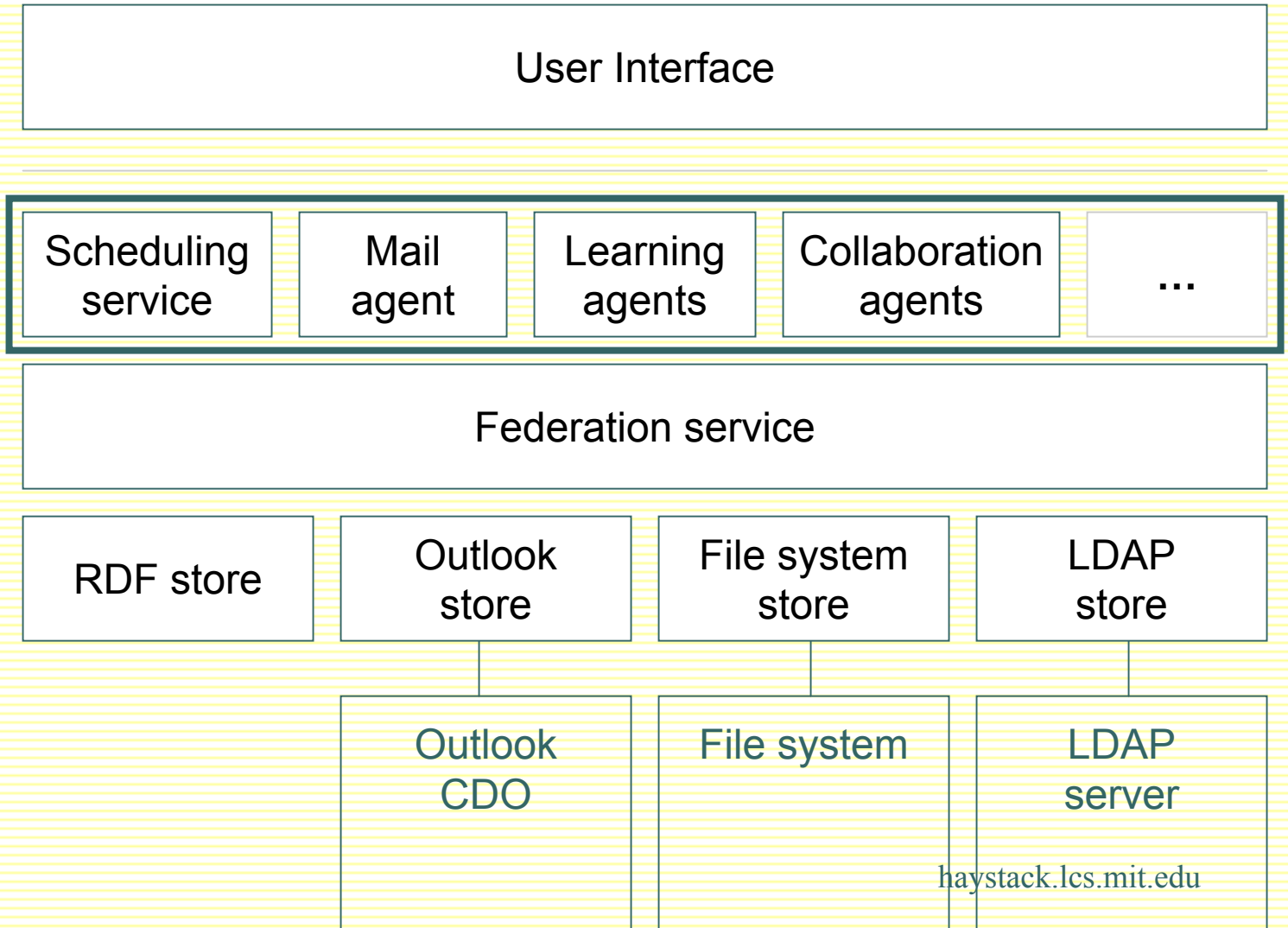


System Architecture

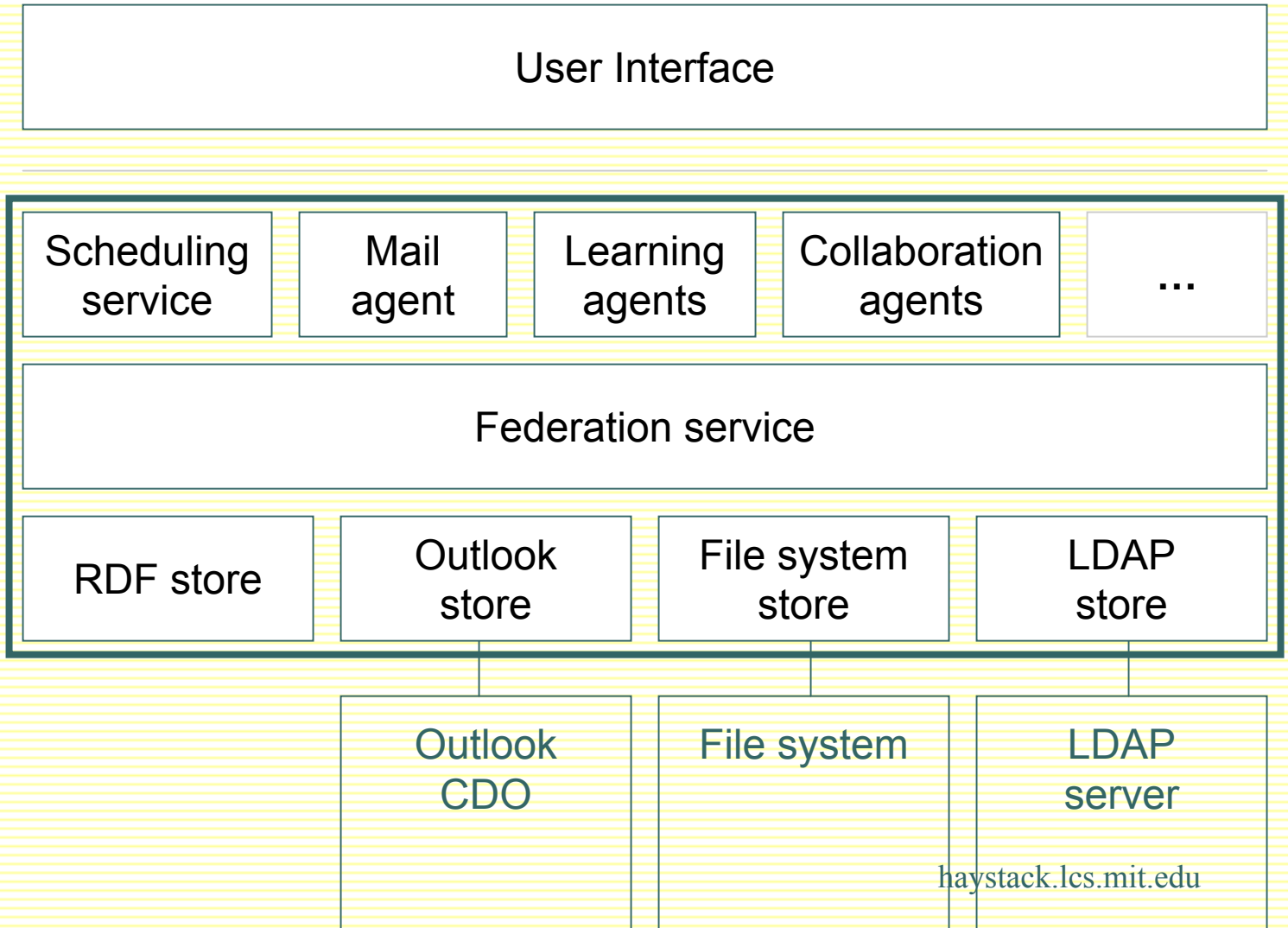


haystack.lcs.mit.edu

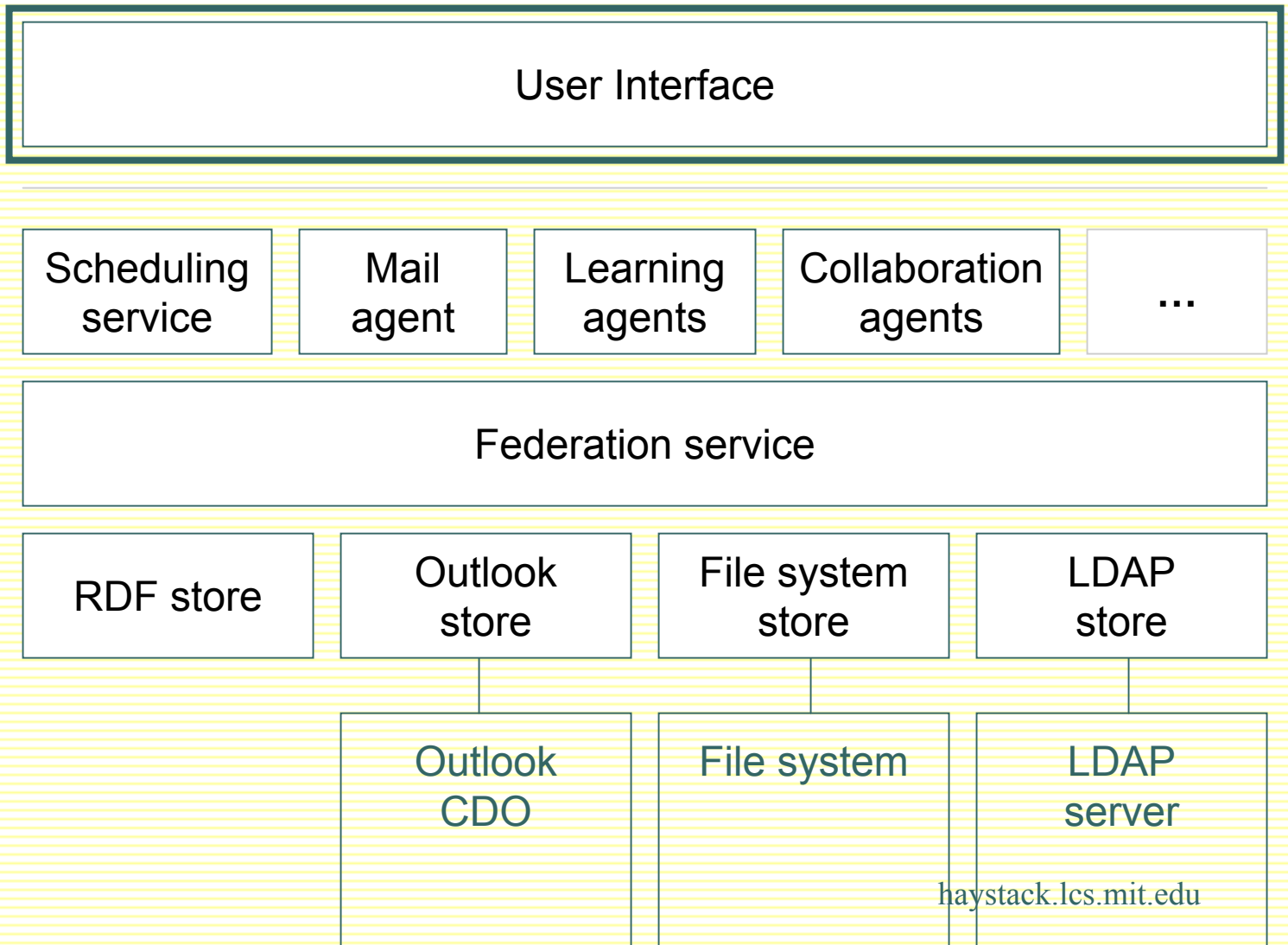
System Architecture



System Architecture



System Architecture

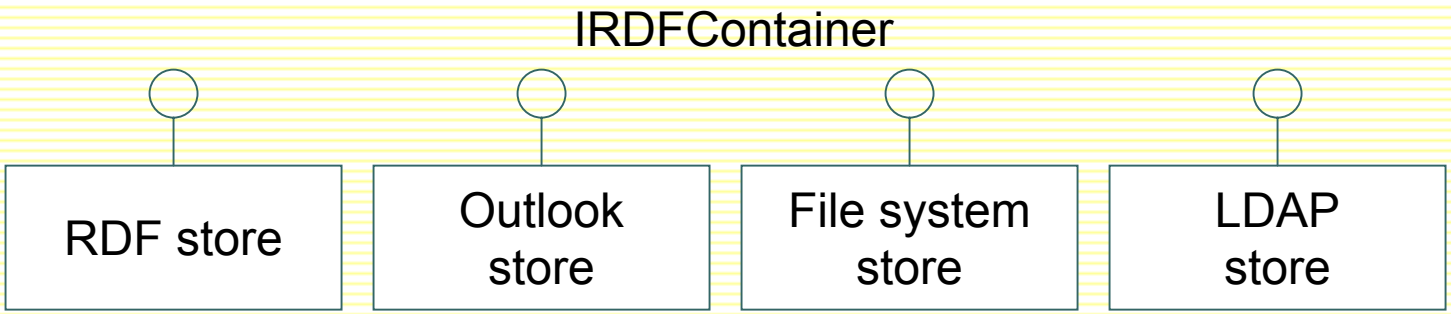


RDF Container Abstraction

- Information stores are RDF containers
- IRDFContainer: adding, removing, querying
- Federation Service is also an RDF container

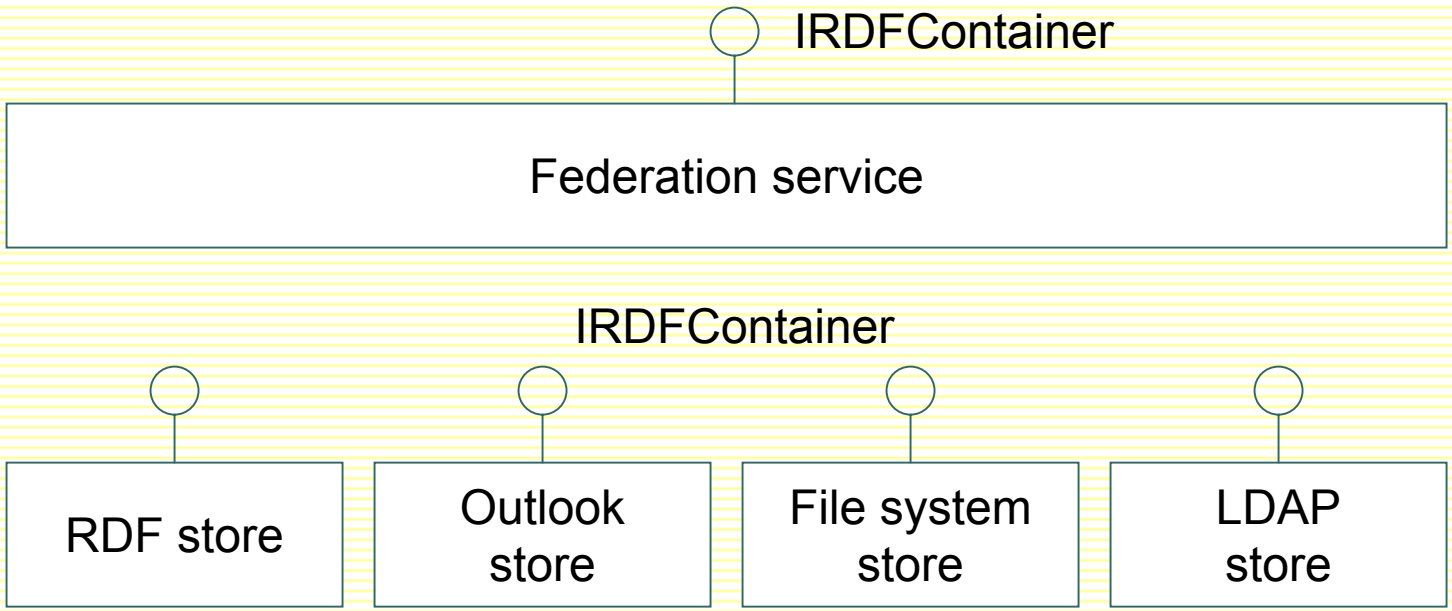
RDF Container Abstraction

- Information stores are RDF containers
- IRDFContainer: adding, removing, querying
- Federation Service is also an RDF container



RDF Container Abstraction

- Information stores are RDF containers
- IRDFContainer: adding, removing, querying
- Federation Service is also an RDF container



Automation

- Services and agents
- Callable entities with SOAP-like interfaces
- WSDL-like ontology describes service interfaces
- All interface information stored in RDF store
- Written in Java, Python

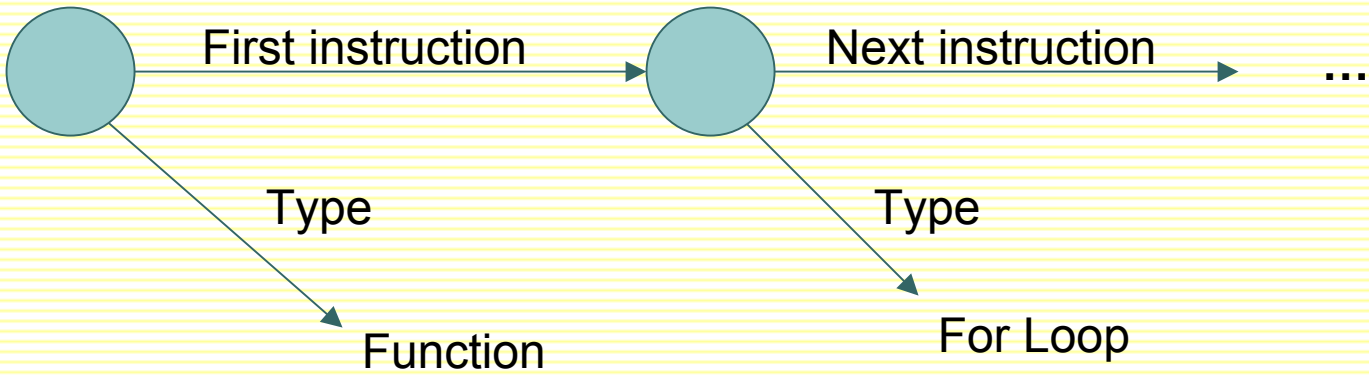
Adenine

- rdfContainer.add (new Statement (
  new Resource (“<urn:mySchema:John>”),
  new Resource (“<urn:mySchema:likes>”),
  new Resource (“<urn:mySchema:Mary>”)));

- add { :John :likes :Mary }
- = friendsOfMary (query {
  ?x :friendOf :Mary
  ?x dc:title ?y
  }
  (List ?y)
)

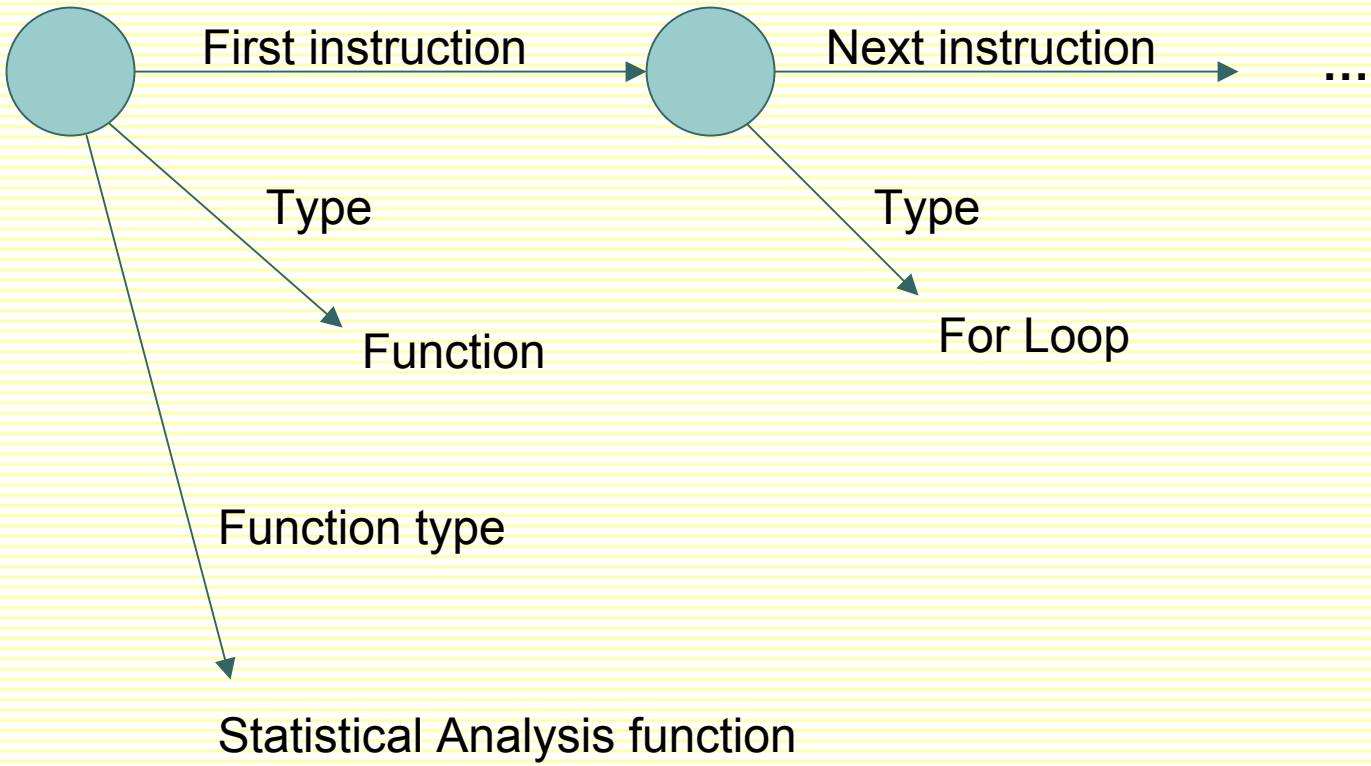
Code is Data

- Adenine compiles into RDF



Code is Data

- Adenine compiles into RDF



UI is Data

- Data for displaying data

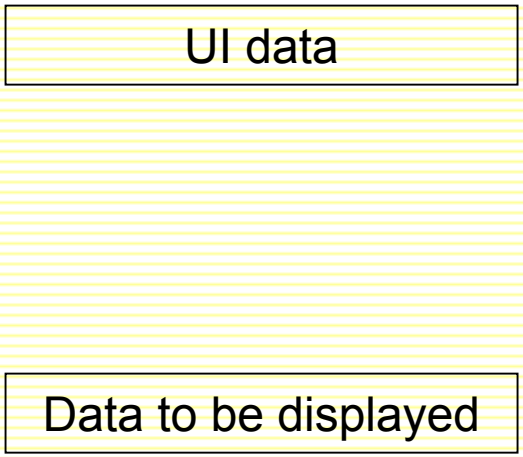
UI is Data

- Data for displaying data

Data to be displayed

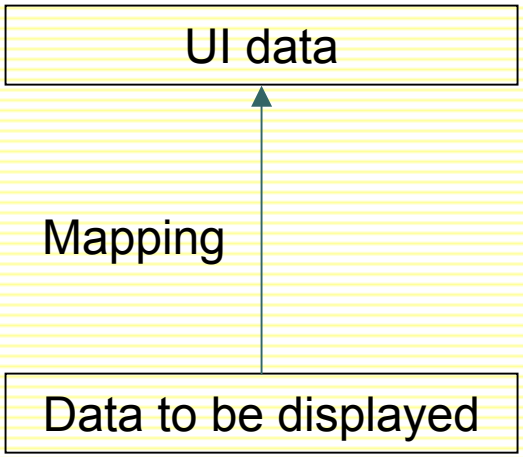
UI is Data

- Data for displaying data



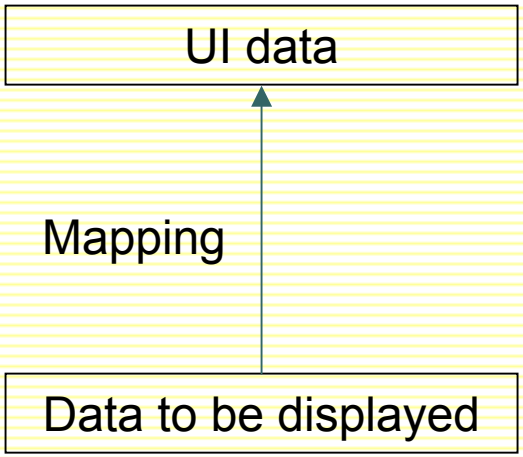
UI is Data

- Data for displaying data



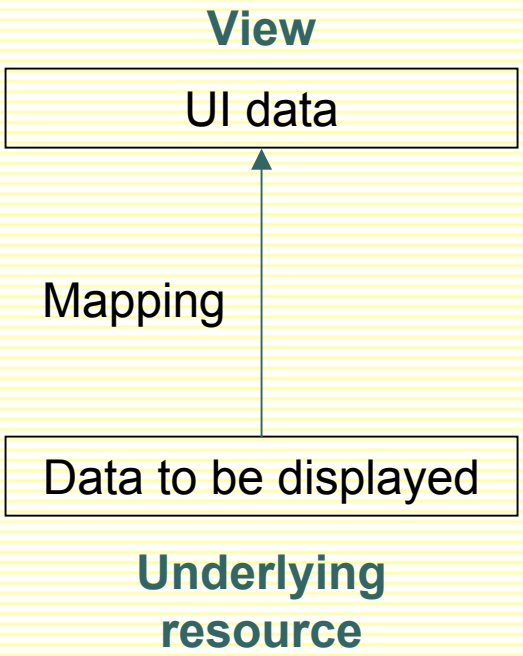
UI is Data

- Data for displaying data



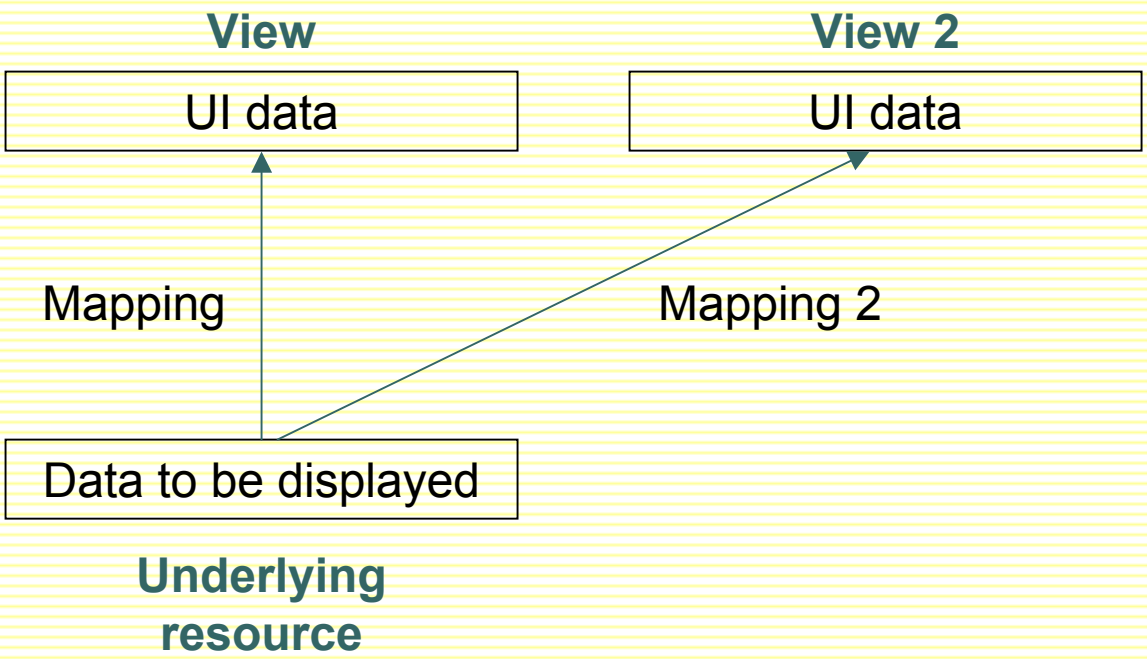
UI is Data

- Data for displaying data



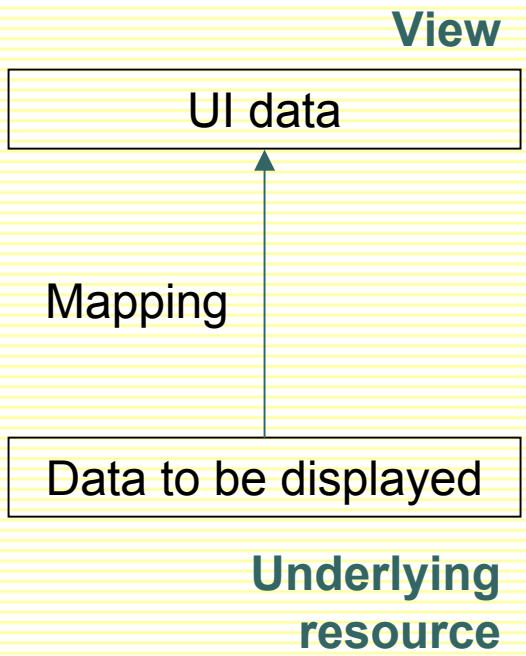
UI is Data

- Data for displaying data



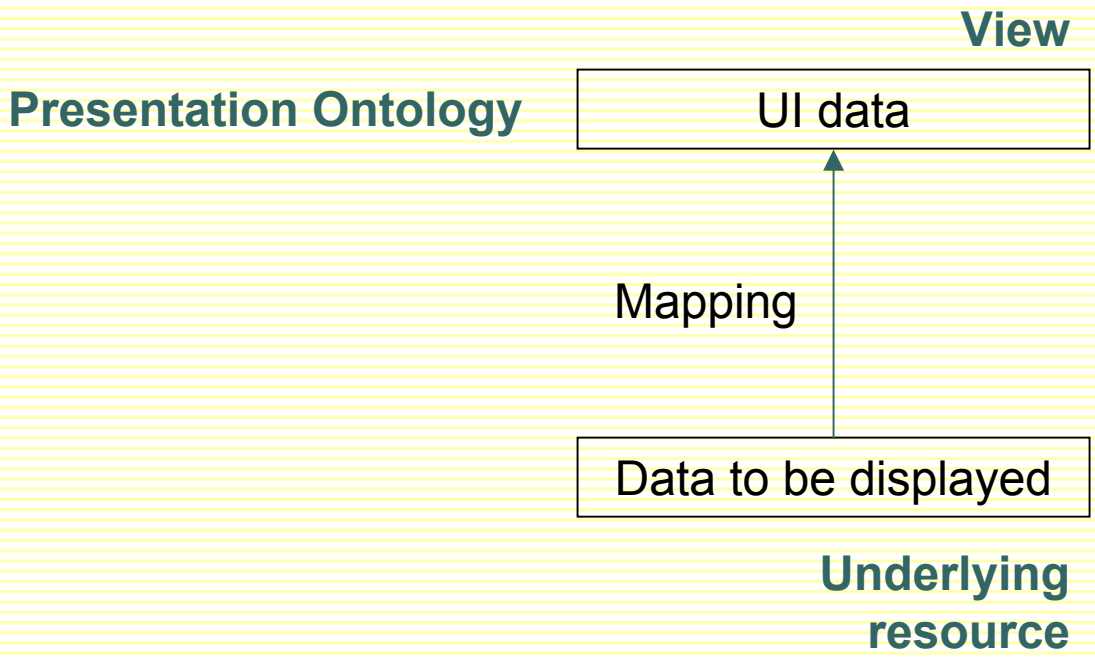
Semantic UI

- Presentation ontology
- Rendering engine



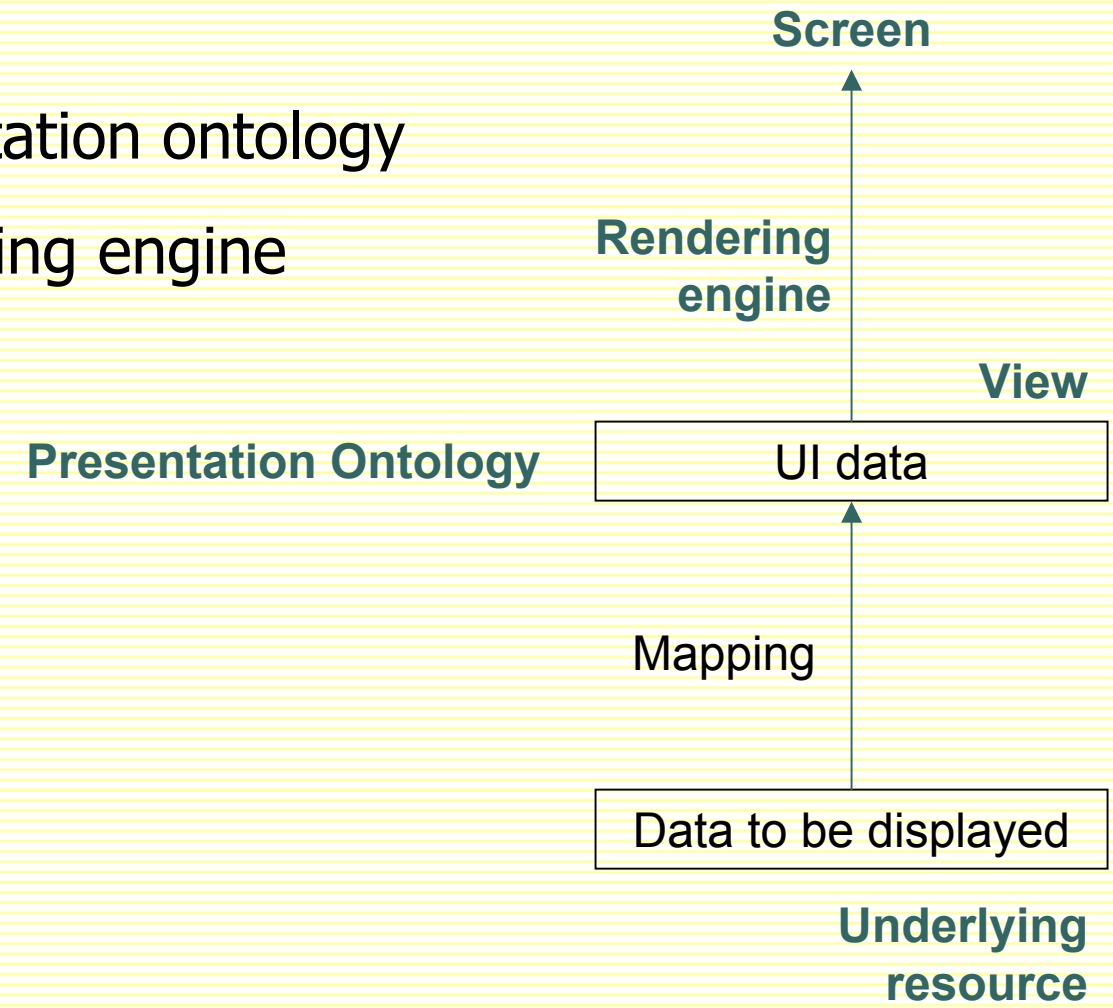
Semantic UI

- Presentation ontology
- Rendering engine

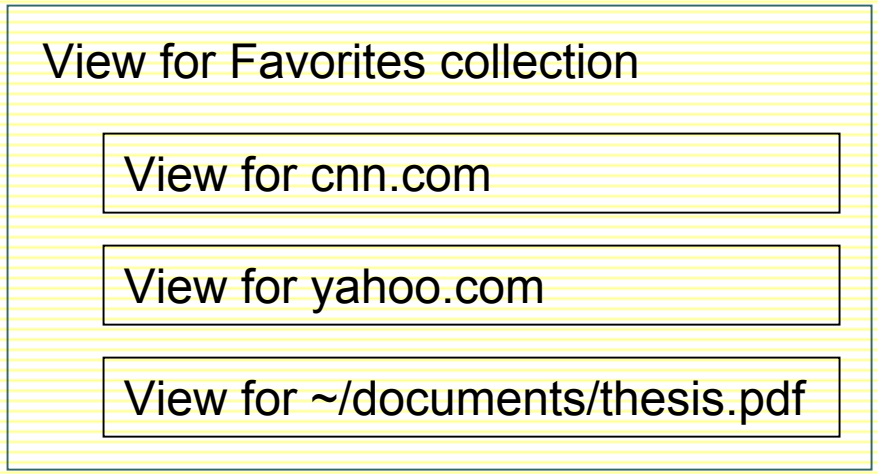


Semantic UI

- Presentation ontology
- Rendering engine

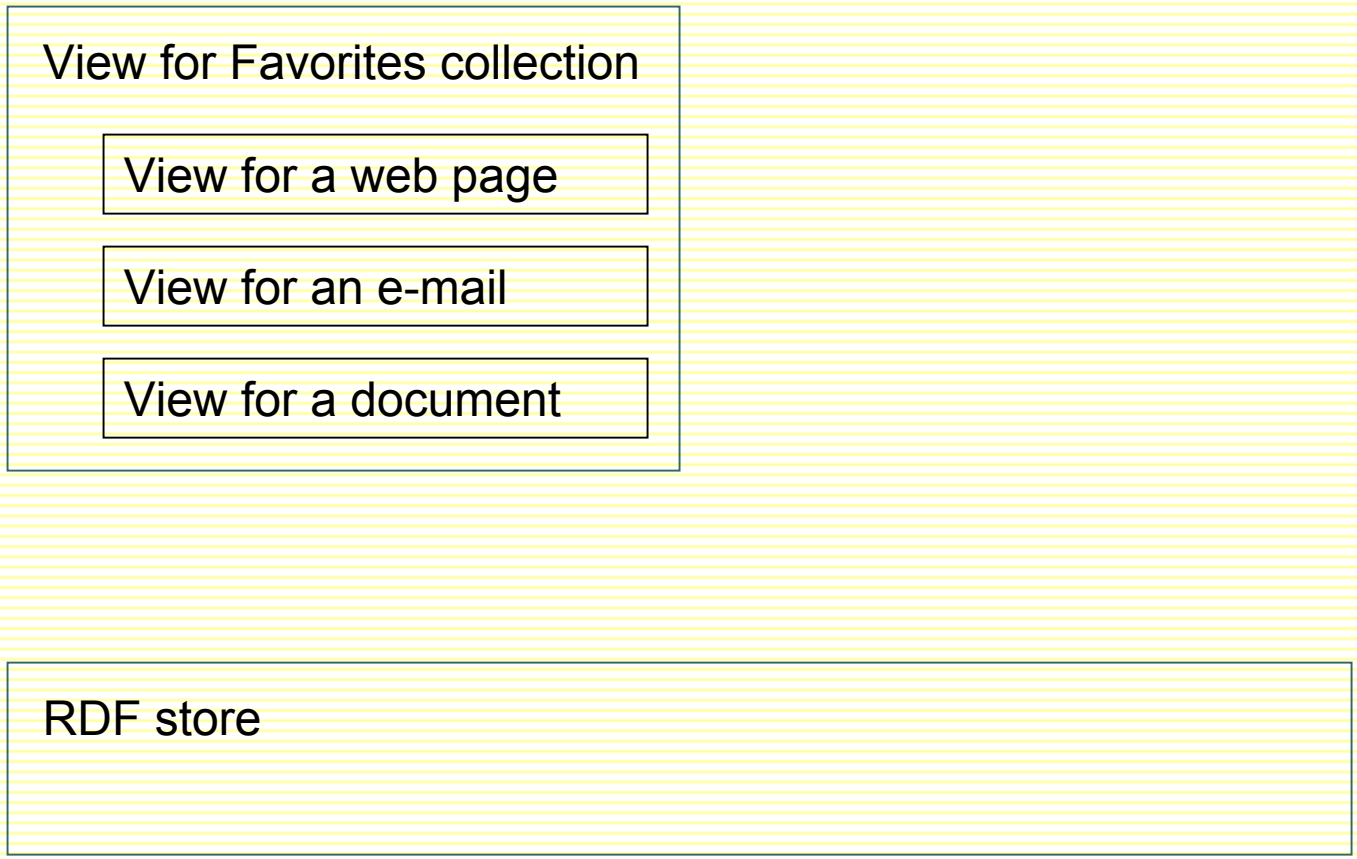


Composing Views



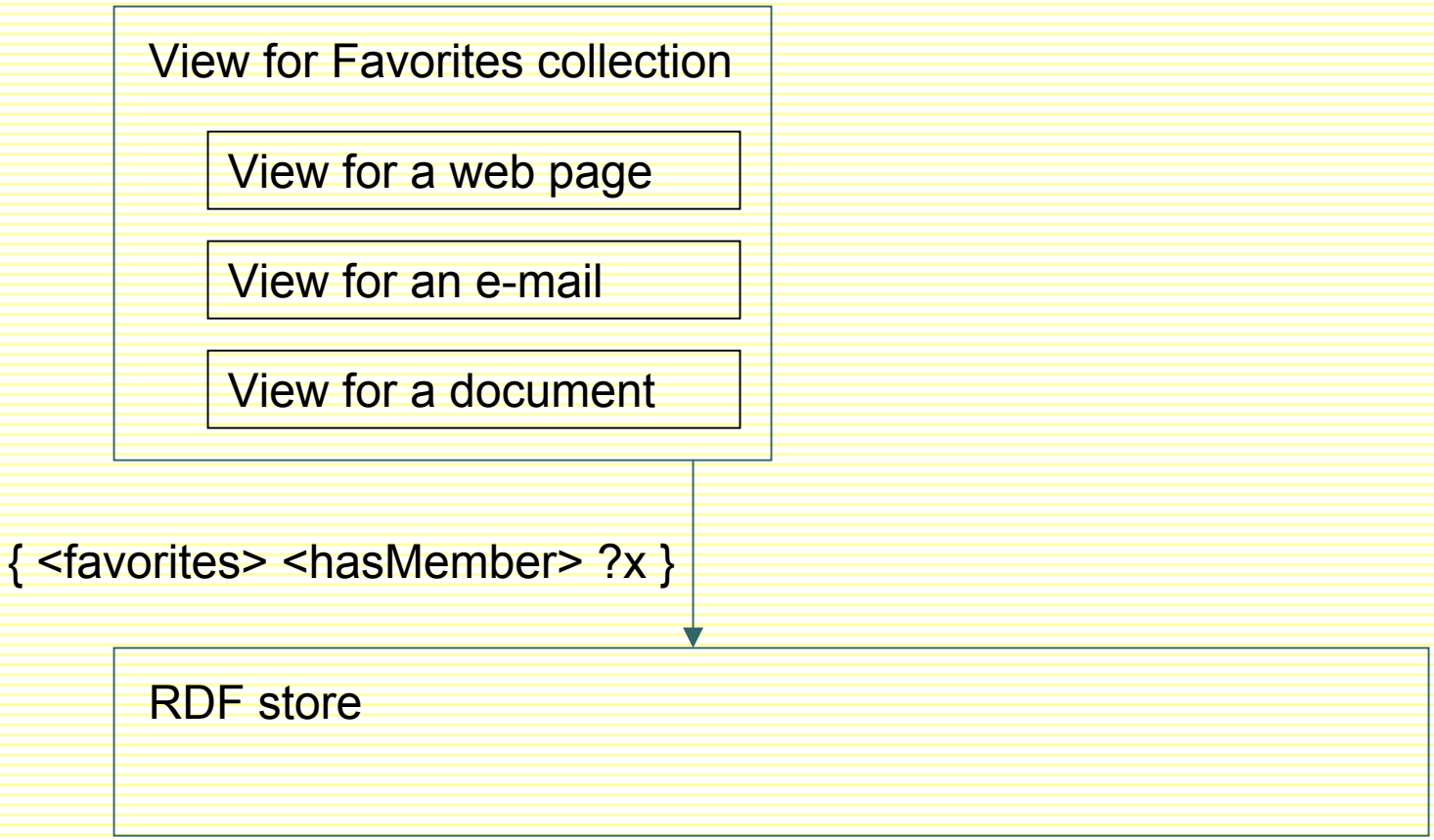
Event Firing

- Event firing from RDF store



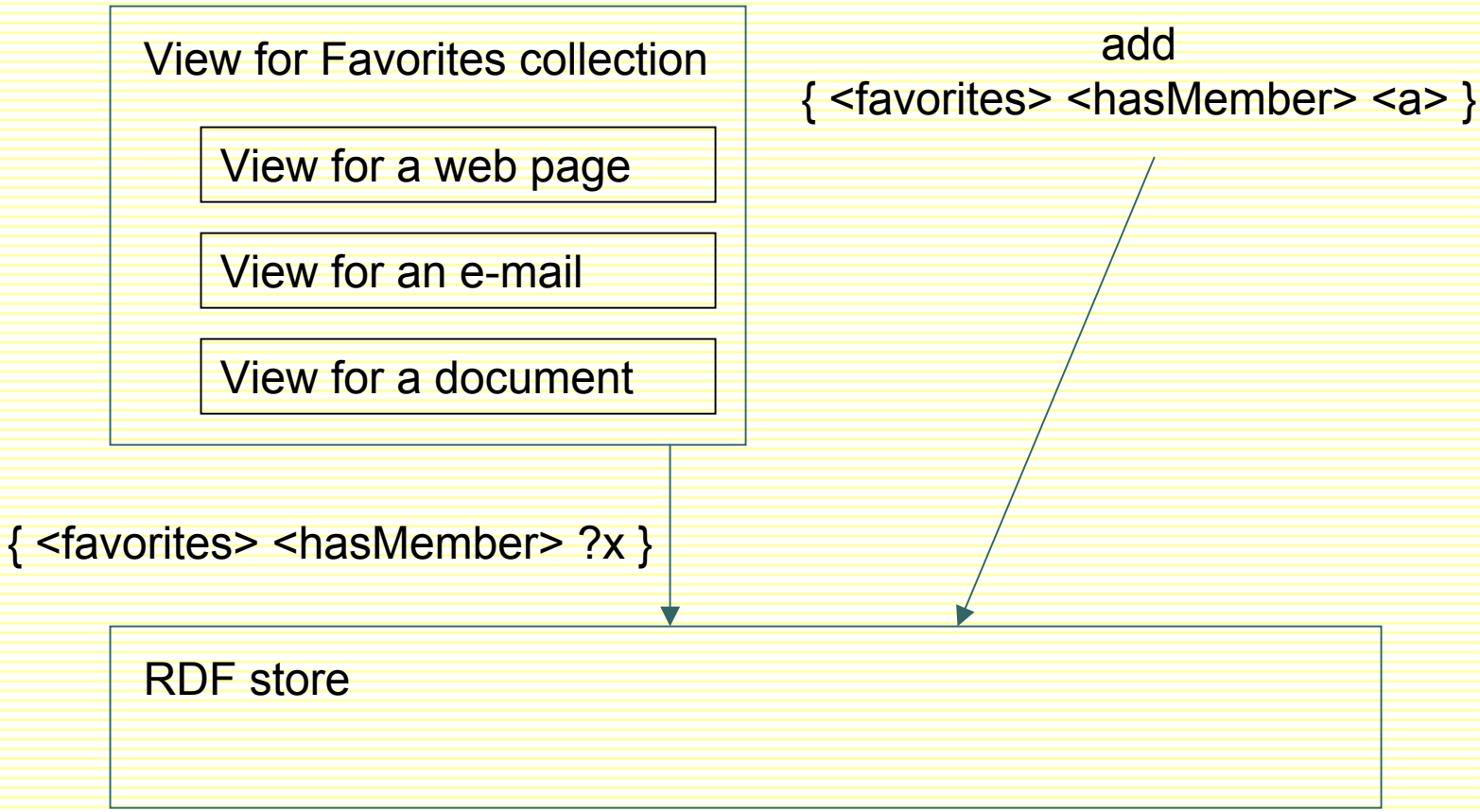
Event Firing

- Event firing from RDF store



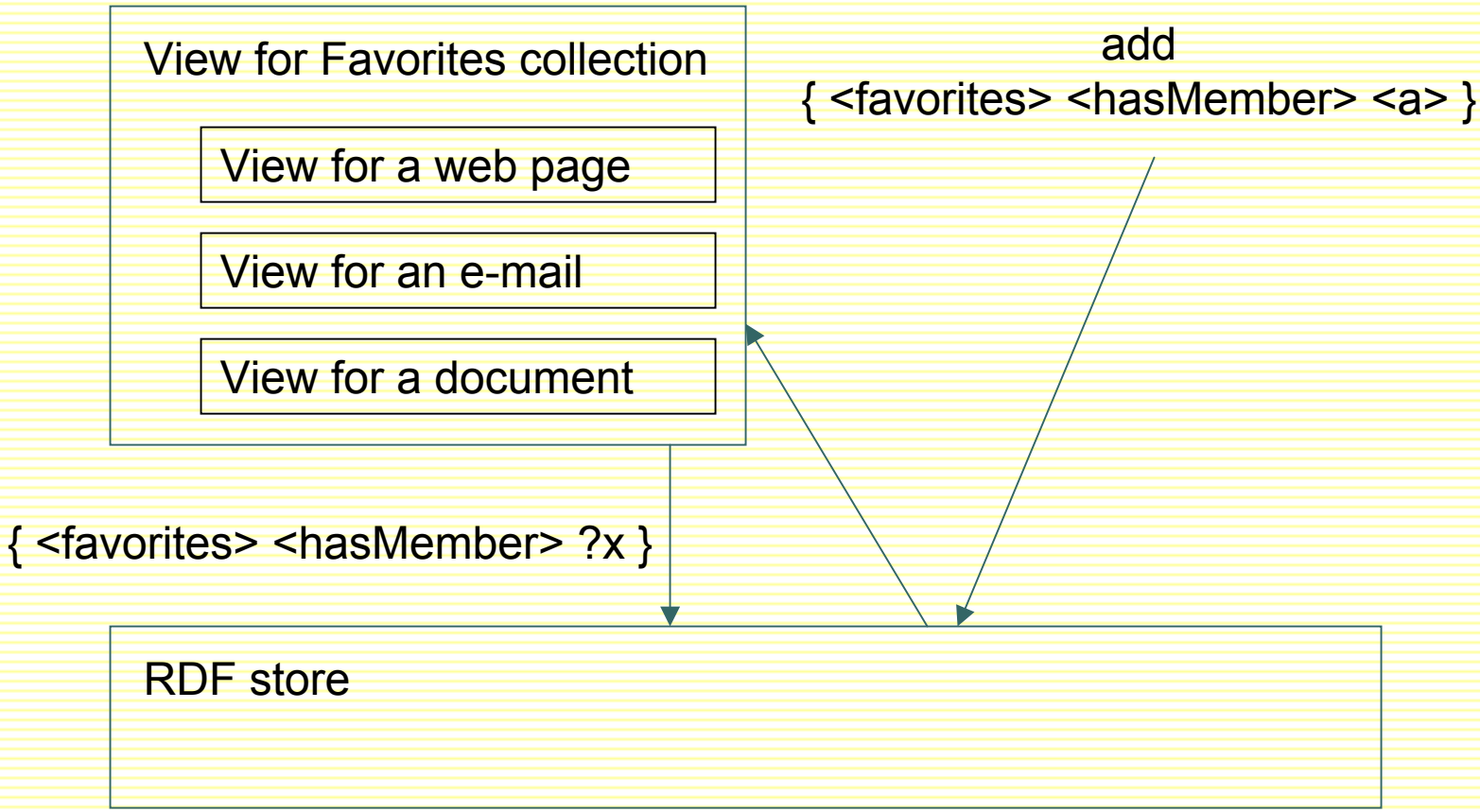
Event Firing

- Event firing from RDF store



Event Firing

- Event firing from RDF store



Benefits

- Information processing decoupled from presentation
- Lower barrier of entry for development
- Uniform support for features like context menus

- Internationalization? Accessibility?
  - "Open on Monday, Tuesday and Thursday"
  - "□□□□□□□□"

Summary

- Using Semantic Web technology to improve end user experience
- Unified storage format; RDF used like a file system
- Adenine: manipulates RDF natively
- Semantic UI

Thank You For Your Attention

Dennis Quan (dquan@media.mit.edu)

David Huynh (dfhuynh@ai.mit.edu)

Paper

http://www.ai.mit.edu/people/dquan/overview.pdf