

veryvery  
alpha!

# IKS Early Adopters Workshop

Salzburg, June 22-23, 2010

## Introducing FISE, the IKS RESTful semantic engine

Bertrand Delacrétaz

Senior Developer, Day Software ([www.day.com](http://www.day.com))

Member, Apache Software Foundation ([apache.org](http://apache.org))



NOT  
a semantic  
guru!

IKS – Interactive Knowledge Stack for Semantic Content Management Systems



IKS is co-funded by the European Union and develops  
new technology for intelligent content management



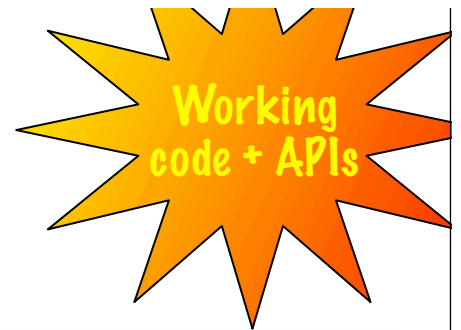
# What?

does FISE do and

# How?

FISE:  
Furtwangen  
IKS  
Semantic  
Engine  
(«phase»)

# FISE server and plug-ins



**RESTful HTTP interface**  
POST content  
GET enhanced content  
GET query results

↑ ↓

SPARL queries as well

FISE server (OSGi-based)



Queries (Apache Clerezza)



Storage (in-memory/Clerezza)



FISE EnhancementEngine API

Autotagging



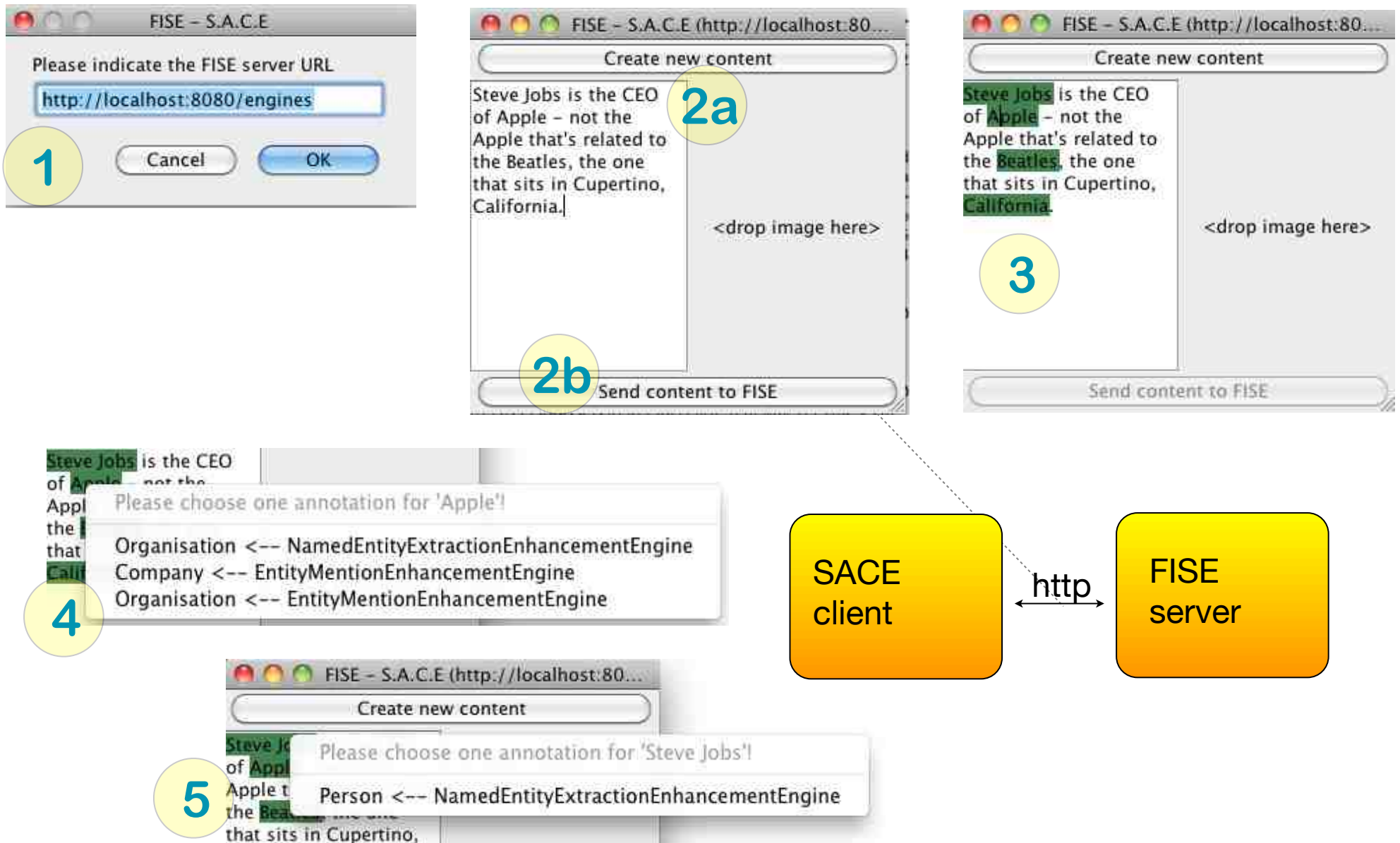
FISE EnhancementEngine API

Named Entity Recognition

Plug-in APIs for more engines...



# FISE demo: SACE client

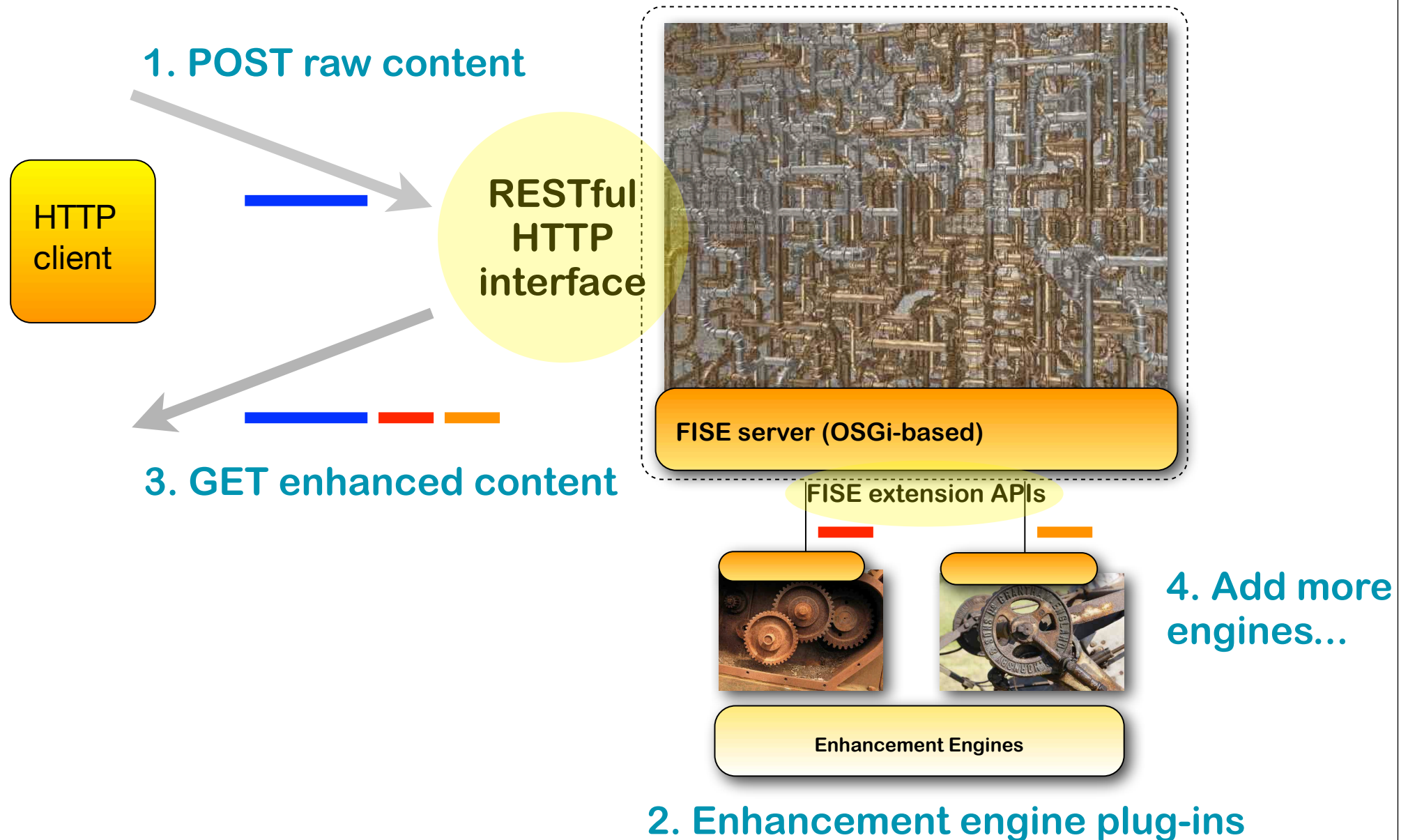


# SACE client GETs RDF from FISE

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:j.0="http://purl.org/dc/terms/"
  xmlns:j.1="http://rdfs.org/sioc/ns#"
  xmlns:j.2="http://fise.iks-project.eu/ontology/" >
<rdf:Description rdf:about="urn:enhancement-ce190f1b-698c-5038-3057-88da26c3f722">
  <rdf:type rdf:resource="http://fise.iks-project.eu/ontology/Enhancement"/>
  ...
  <j.2:extracted-from
    rdf:resource="urn:content-item-sha1-8559dbc3ce78b24"/>
  <j.0:creator rdf:datatype="http://www.w3.org/2001/XMLSchema#string">eu.iksproject....EntityMentionEnhancementEngine</j.0:creator>
  <j.2:entity-reference
    rdf:resource="http://dbpedia.org/resource/Berkeley%2C%20California"/>
  <j.2:entity-label rdf:datatype="http://www.w3.org/2001/XMLSchema#string">Berkeley, California</j.2:entity-label>
  <j.2:confidence rdf:datatype="http://www.w3.org/2001/XMLSchema#double">3.4116923809051514</j.2:confidence>
  <j.2:entity-type rdf:resource="http://www.w3.org/2002/07/owl#Thing"/>
  <j.2:entity-type rdf:resource="http://dbpedia.org/ontology/Place"/>
  <j.2:entity-type rdf:resource="http://dbpedia.org/ontology/PopulatedPlace"/>
  <j.2:entity-type rdf:resource="http://dbpedia.org/ontology/Area"/>
</rdf:Description>
```

The EntityMentionEnhancementEngine thinks that content item is related to the «Berkeley, California» Place from dbpedia.

# The FISE scenario



# SPARQL endpoint

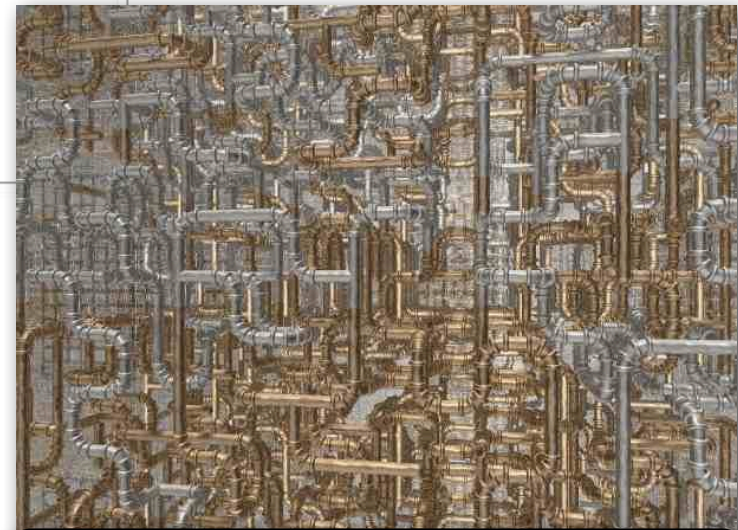
powered by Apache Clerezza

```
PREFIX fise: <http://fise.iks-project.eu/ontology/>
PREFIX dc: <http://purl.org/dc/terms/>
SELECT distinct ?enhancement ?content ?engine ?extraction_time
WHERE {
  ?enhancement a fise:Enhancement .
  ?enhancement fise:extracted-from ?content .
  ?enhancement dc:creator ?engine .
  ?enhancement dc:created ?extraction_time .
}
ORDER BY DESC(?extraction_time) LIMIT 5
```

Run SPARQL query

http  
GET

```
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
...
<results>
<result>
<binding name="content">
  <uri>/11</uri>
</binding>
<binding name="engine">
  <literal datatype="http://www.w3.org/2001/XMLSchema#string">
    eu.iksproject.fise.EntityMentionEnhancementEngine</literal>
</binding>
<binding name="extraction_time">
  <literal datatype="http://www.w3.org/2001/XMLSchema#dateTime">
    2010-06-21T23:22:15.802+02:00</literal>
</binding>
<binding name="enhancement">
  <uri>urn:enhancement-d7c755a7-09d0-8a1f-5a15-ff177c1d3593</uri>
</binding>
</result>
<result>
```



FISE server (OSGi-based)

the

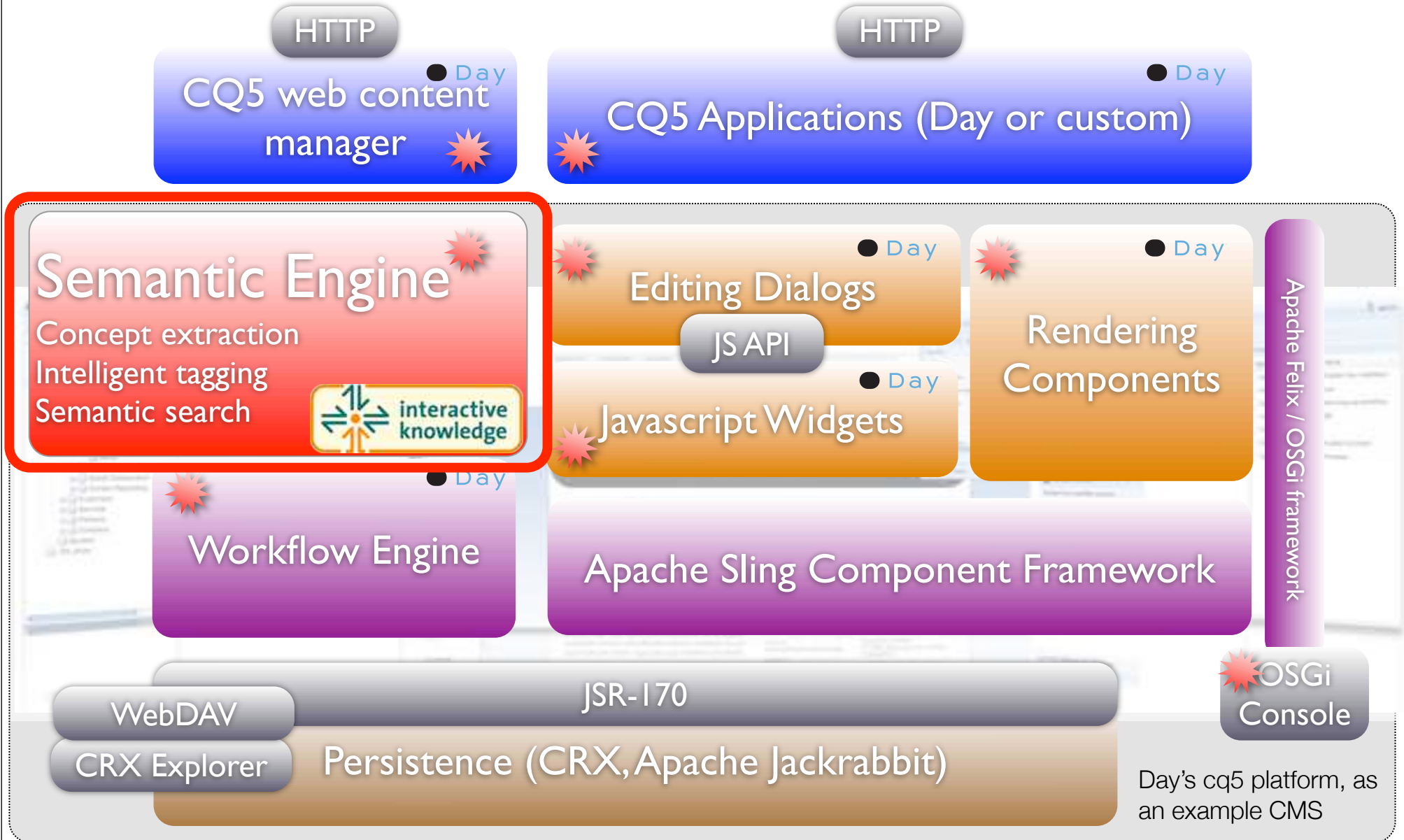
# Architecture

of FISE

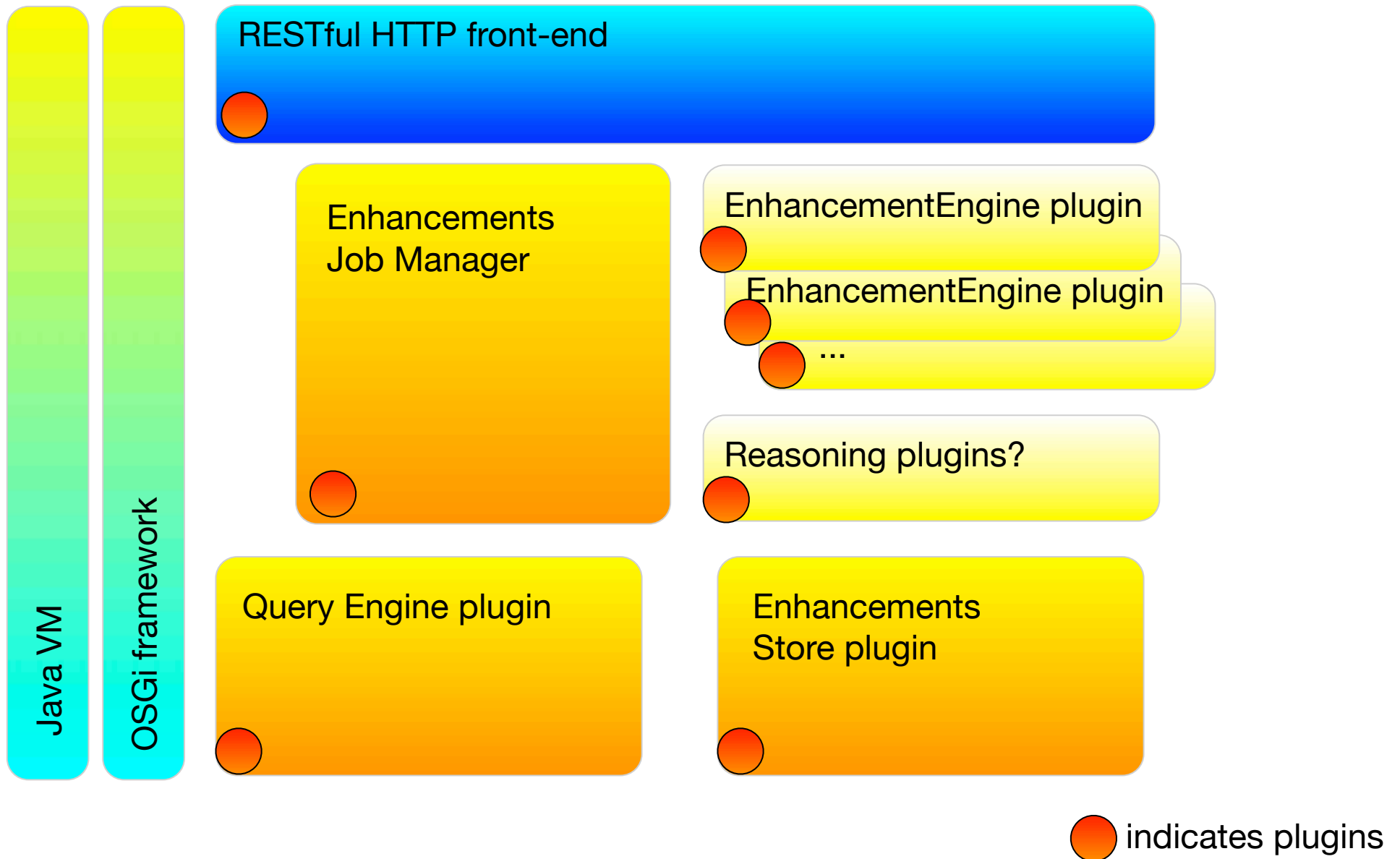


# FISE vision: RESTful semantic engine

add-on for existing CMS



# FISE: minimal core with plugins



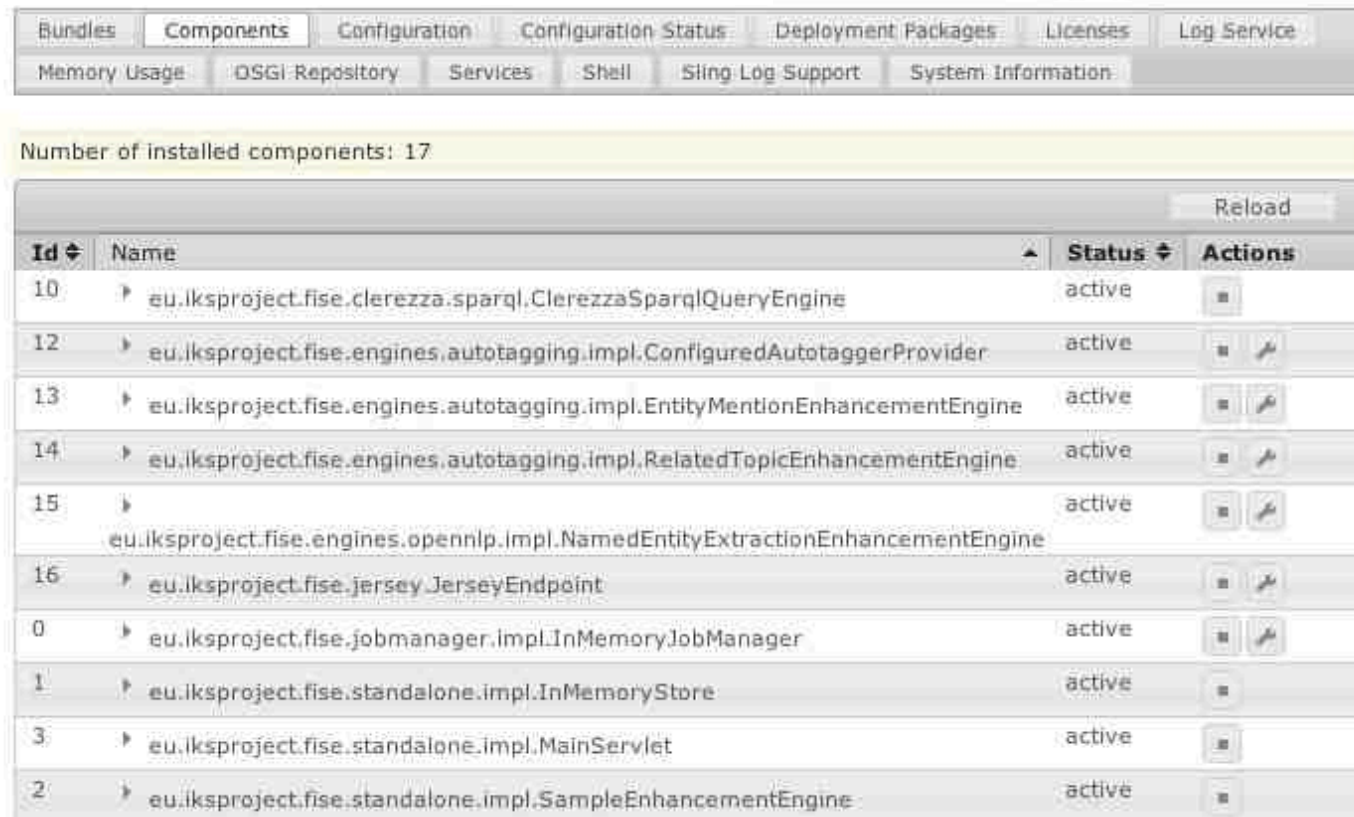
# OSGi for modularization



Maven plugins and **OSGi console** provided by Apache Felix.

Launcher provided by Apache Sling for standalone server.

Semantic components provided by Apache Clerezza



Number of installed components: 17

Id	Name	Status	Actions
10	eu.iksproject.fise.clerezza.sparql.ClerezzaSparqlQueryEngine	active	[Stop]
12	eu.iksproject.fise.engines.autotagging.impl.ConfiguredAutotaggerProvider	active	[Stop] [Refresh]
13	eu.iksproject.fise.engines.autotagging.impl.EntityMentionEnhancementEngine	active	[Stop] [Refresh]
14	eu.iksproject.fise.engines.autotagging.impl.RelatedTopicEnhancementEngine	active	[Stop] [Refresh]
15	eu.iksproject.fise.engines.opennlp.impl.NamedEntityExtractionEnhancementEngine	active	[Stop] [Refresh]
16	eu.iksproject.fise.jersey.JerseyEndpoint	active	[Stop] [Refresh]
0	eu.iksproject.fise.jobmanager.impl.InMemoryJobManager	active	[Stop] [Refresh]
1	eu.iksproject.fise.standalone.impl.InMemoryStore	active	[Stop]
3	eu.iksproject.fise.standalone.impl.MainServlet	active	[Stop]
2	eu.iksproject.fise.standalone.impl.SampleEnhancementEngine	active	[Stop]

the FISE

# Services API

excerpts

# FISE ContentItem API

```
/** A unit of content that FISE can enhance.
 *
 * Gives access to the binary content that
 * was registered, and the Graph that represents its metadata
 * (provided by client and/or generated).
 */
public interface ContentItem {
    /** Unique ID, either supplied by client or generated by FISE */
    String getId();

    /** The binary content stream */
    InputStream getStream();

    /** Mime-type of the input stream */
    String getMimeType();

    /** Optional metadata */
    MGraph getMetadata();
}
```



Adapter pattern  
instead of  
getInputStream?  
Helpers to add  
Enhancements to  
the Graph?

# FISE EnhancementEngine API

```
/**
 * Interface to internal or external semantic enhancement engines. There will
 * usually be several of those, that the EnhancementJobManager uses to enhance
 * content items.
 */
public interface EnhancementEngine {

    /**
     * Indicate if this engine can enhance supplied ContentItem, and if it
     * suggests enhancing it synchronously or asynchronously. The
     * {@link EnhancementJobManager} can force sync/async mode if desired, it is
     * just a suggestion from the engine.
     *
     * @throws EngineException if the introspecting process of the content item
     * fails
     */
    int canEnhance(ContentItem ci) throws EngineException;

    /**
     * Compute enhancements for supplied ContentItem. The results of the process
     * are expected to be stored in the metadata of the content item.
     *
     * The client (usually an {@link EnhancementJobManager}) should take care of
     * persistent storage of the enhanced {@link ContentItem}.
     *
     * @throws EngineException if the underlying process failed to work as
     * expected
     */
    void computeEnhancements(ContentItem ci) throws EngineException;
}
```



Collaboration  
between engines:  
Android-like  
intents, event-  
based tuple  
space?

current FISE

# Enhancement Engines

# FISE EnhancementEngine plugins

early June 2010

## **NamedEntityExtractionEnhancementEngine:**

Uses OpenNLP sentence detector and name finder to detect persons, places and organisations.

## **EntityMentionEnhancementEngine:**

Creates EntityAnnotations using an index of DBpedia entities.

## **LocationEnhancementEngine:**

Creates fise:EntityAnnotations based on the <http://geonames.org> dataset.

## **Tagging using the Zemanta API:**

Simple wrapper for Zemanta

## **Language identification:**

Identifies a text as: German(de), English(en), ....., Slovenian(sl), Danish(da), Hungarian(hu)

## **Metaxa text extractor:**

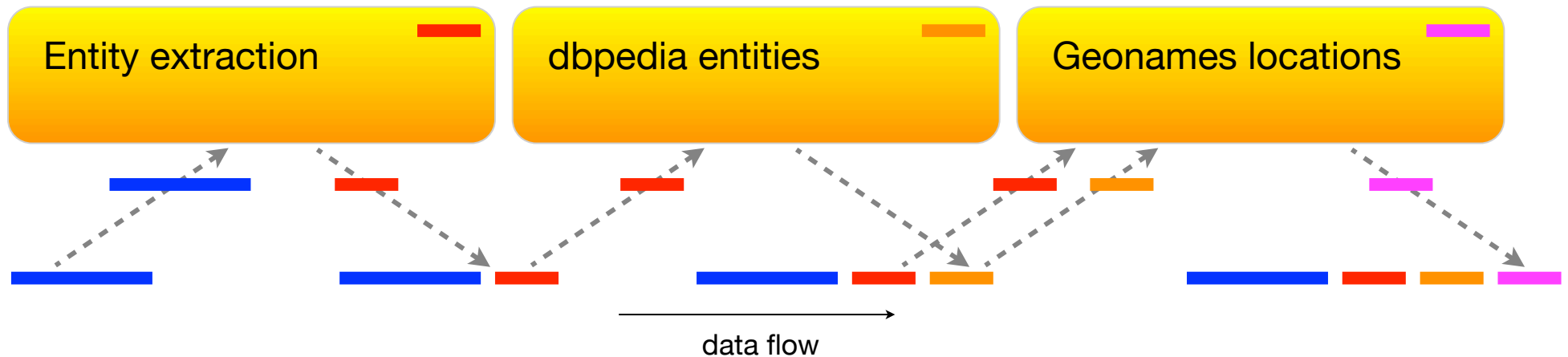
Extracts metadata and text from document formats like PDF, Word, etc.



Fat bundles,  
need to optimize  
dependencies.



# Enhancement Engines collaboration



Engines use metadata from other engines as input.  
Iteratively enhance incoming content.

Resulting output:

content      entities      dbpedia      geonames  
                  



Currently based on engine ordering.  
Need a better mechanism, intents, tuple space etc.

# Getting Started

# Getting started with FISE

Start at <http://wiki.iks-project.eu/index.php/FISE> or Google «IKS FISE».

Binary downloads are available there.

Build from source code, not for the faint of heart currently (Maven snapshots!)

```
java -Xmx512M  
-jar iks-fise-server-rNNN.jar
```

(add -h at the end to see more options)

Then open  
<http://localhost:8080>

## Welcome to IKS FISE!

Here are your entry points to FISE, the RESTful semantic engine:

### [The official FISE Wiki](#)

The FISE project wiki for latest news, documentation and mailing list.

### [/engines](#)

List the registered EnhancementEngines and simple HTML form interface.

### [/fise](#)

*Note: this is going to be renamed to /store in a future version of the API.*

1. POST content to `./fise/content-id` with `Content-Type=text/plain`.
2. GET enhancements from the same URL.

`content-id` any valid path that can be used to fetch your item back later.

On a unix-ish box you can use run the following command from the top-level source directory to populate content items:

```
for i in data/text-examples/*.txt;  
do  
  curl -H'Content-Type:text/plain' -T $i http://localhost:8080/fise/${basename $i};  
done
```

### [/sparql](#)

Sparql endpoint for the FISE store.

[Sparql](#) is the standard query language the most commonly used to provide interactive access to semantic data.

Use 'q' request parameter to specify query against enhancements created when content is added to FISE. Use the above interfaces before using this.

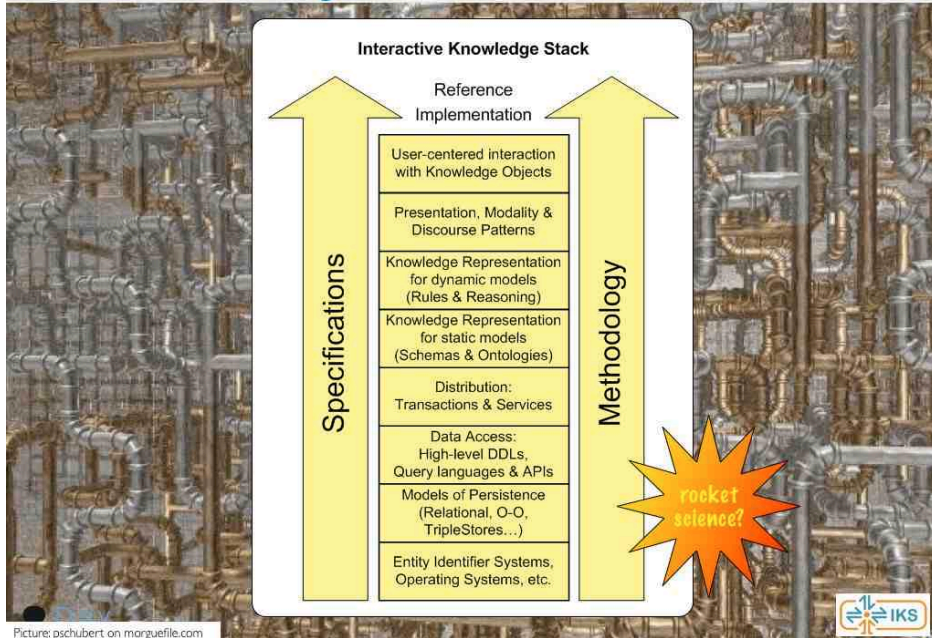
```
PREFIX fise: <http://fise.iks-project.eu/ontology/>  
PREFIX dc: <http://purl.org/dc/terms/>  
SELECT ?enhancement ?content ?engine ?extraction_time  
WHERE {  
  ?enhancement a fise:Enhancement .  
  ?enhancement fise:extracted-from ?content .  
  ?enhancement dc:creator ?engine .  
  ?enhancement dc:created ?extraction_time .  
}  
ORDER BY DESC(?extraction_time)
```

Run SPARQL query

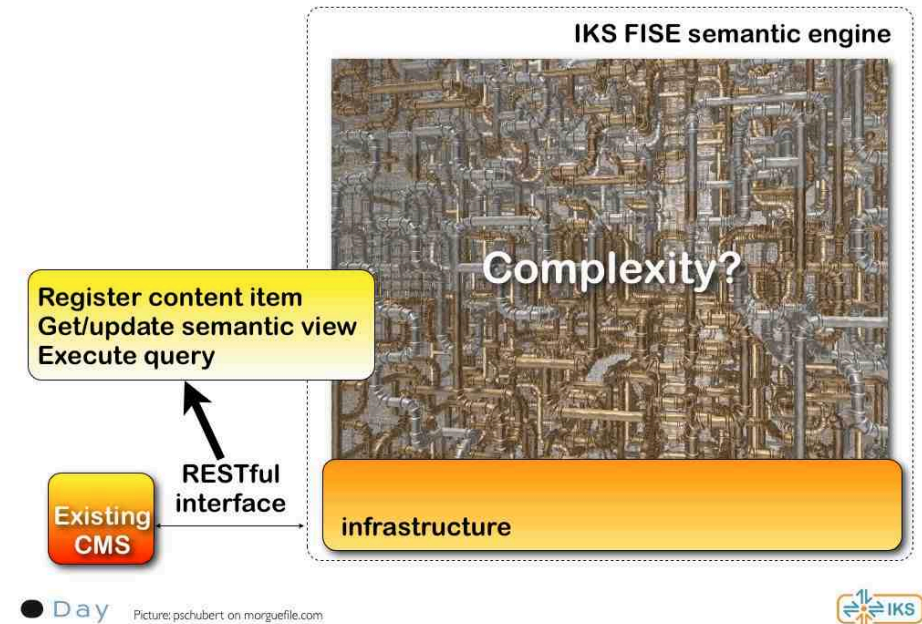
# What's next?

# Conclusions...and What's Next?

## A revolutionary semantic CMS stack



## A RESTful semantic engine



Our main goal is to **build a community** of FISE users, which **provides feedback** on semantic features that add value to existing CMS. And contributes to FISE!

Ideas from the full IKS stack will **percolate into FISE** based on actual use cases from the field.














Let's keep FISE **simple, understandable and robust**, while allowing our researcher's «wild ideas» plugins to be used for more experimental work.



Let us know if FISE would add value to your CMS!



# The IKS Consortium

<p>Project Lead and Coordination Salzburg Research</p> 	<p>Wernher Behrendt Salzburg Research Forschungsgesellschaft m.b.H. Jakob Haringer Straße 5/3   5020 Salzburg, Austria T +43.662.2288-409   F +43.662.2288-222 wernher.behrendt@salzburgresearch.at <a href="http://www.salzburgresearch.at">www.salzburgresearch.at</a></p>
<p>Deutsches Forschungsinstitut für Künstliche Intelligenz (DFKI)</p> 	<p>Universität St. Gallen Institute of Technology Management  University of St.Gallen</p>
<p>Consiglio Nationale delle Ricerche (CNR)</p> 	<p>Software Quality Lab Universität Paderborn</p>  
<p>Software Research and Development Consultancy Ltd (SRDC)</p> 	<p>Hochschule Furtwangen </p>
<p>Nuxeo Sa.</p> 	<p>Alkacon Software GmbH</p> 
<p>TXT Polymedia</p> 	<p>Pisano Holding GmbH</p> 
<p>Nemein Oy</p> 	<p>Day Software AG</p> 