Retrieval of MPEG-7 based Semantic Descriptions

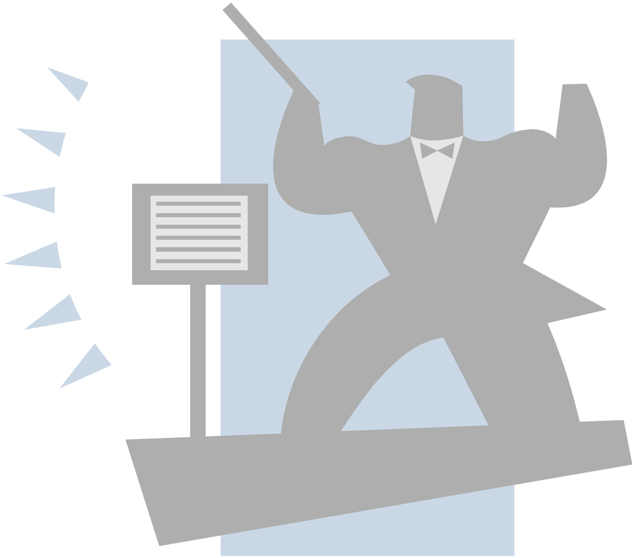
Mathias Lux, Michael Granitzer
Know-Center

{mlux, mgrani}@know-center.at

http://www.**know-center**.at

Content

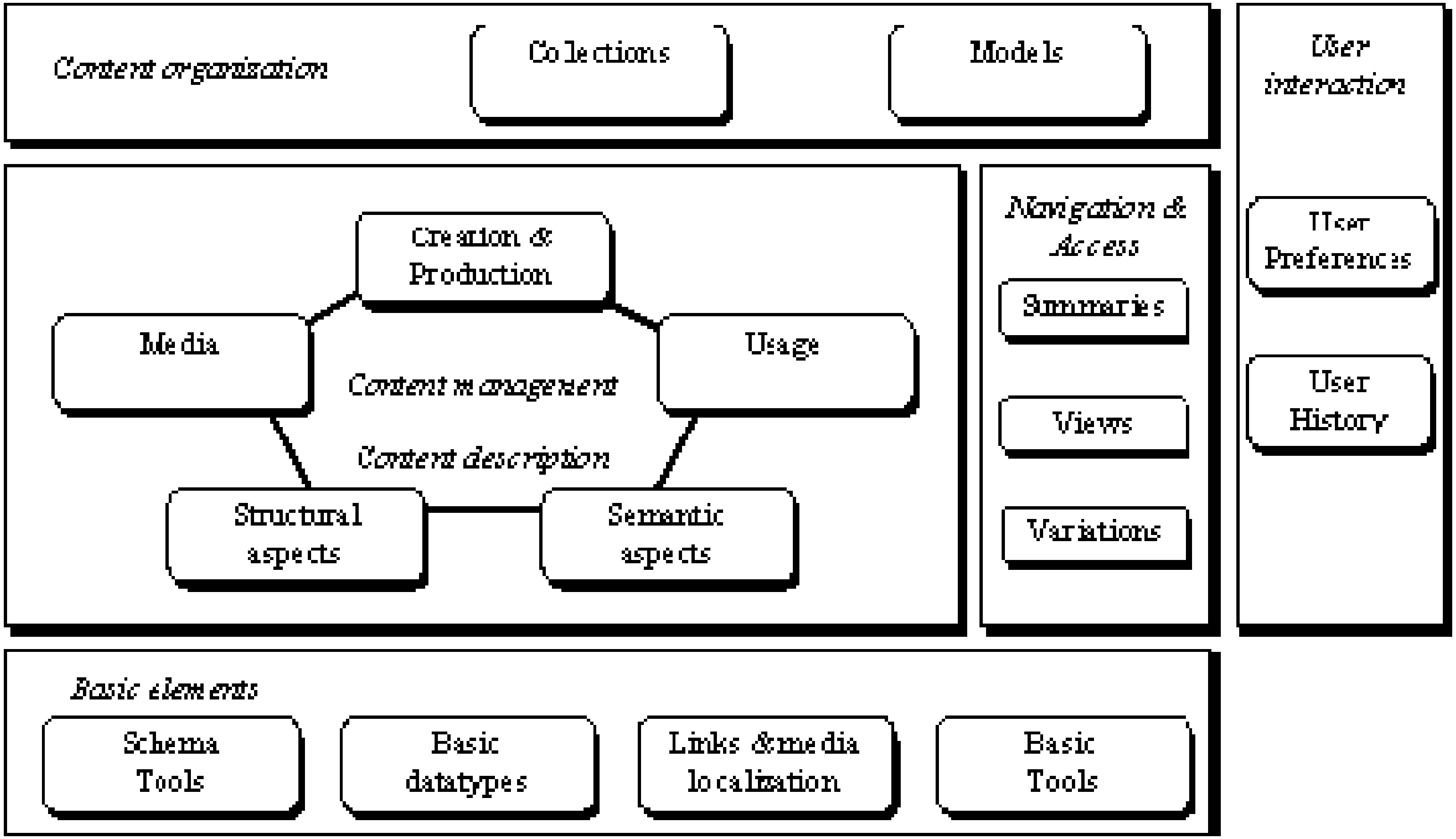
- MPEG-7 and Semantic Descriptions
- Indexing & Search of Semantic Descriptions
- Implementation Details
- Evaluation
- Demonstration
- Future Work



What is MPEG-7?

- „Multimedia Content Description Interface“
- ISO/IEC Standard for MuMe Meta Data
- Representation
  - XML as well as Compressed Binary
- Organized in Descriptors (D) and Descriptor Schemes (DS)

MPEG-7



Content organization
Collections
Models
User interaction
Navigation & Access
Summaries
Views
Variations
User Preferences
User History
Creation & Production
Media
Usage
Content management
Content description
Structural aspects
Semantic aspects
Basic elements
Schema Tools
Basic datatypes
Links & media localization
Basic Tools

MPEG-7 Semantic Descriptions (1/2)

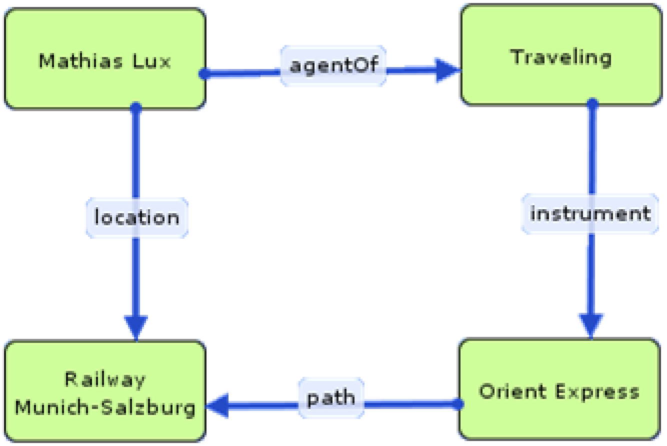
Semantic DS allows Semantic Descriptions:

- Base Descriptor „SemanticBase“
- Inherited are Agents, Places, Times, Events, Concepts, Objects, ...
- D Instances are interpreted as Nodes

Instanz	**Descriptor**
Mathias Lux	Semantic Agent
Orient Express	Semantic Object
Traveling	Semantic Event
Railway München-Salzburg	Semantic Location

MPEG-7 Semantic Descriptions (2/2)

- Semantic Relations interconnect Semantic Objects.
- 45 different Relations, 44 inverse.
- Relations are directed Edges interconnecting Nodes.



Properties of Semantic Descriptions

- Semantic Descriptions are „Labeled Graphs“.
- Node Labels are unique.
- Number of possible Edge Labels is bounded.
- One Semantic Object Instance can be found in multiple Graphs (Domain specific).

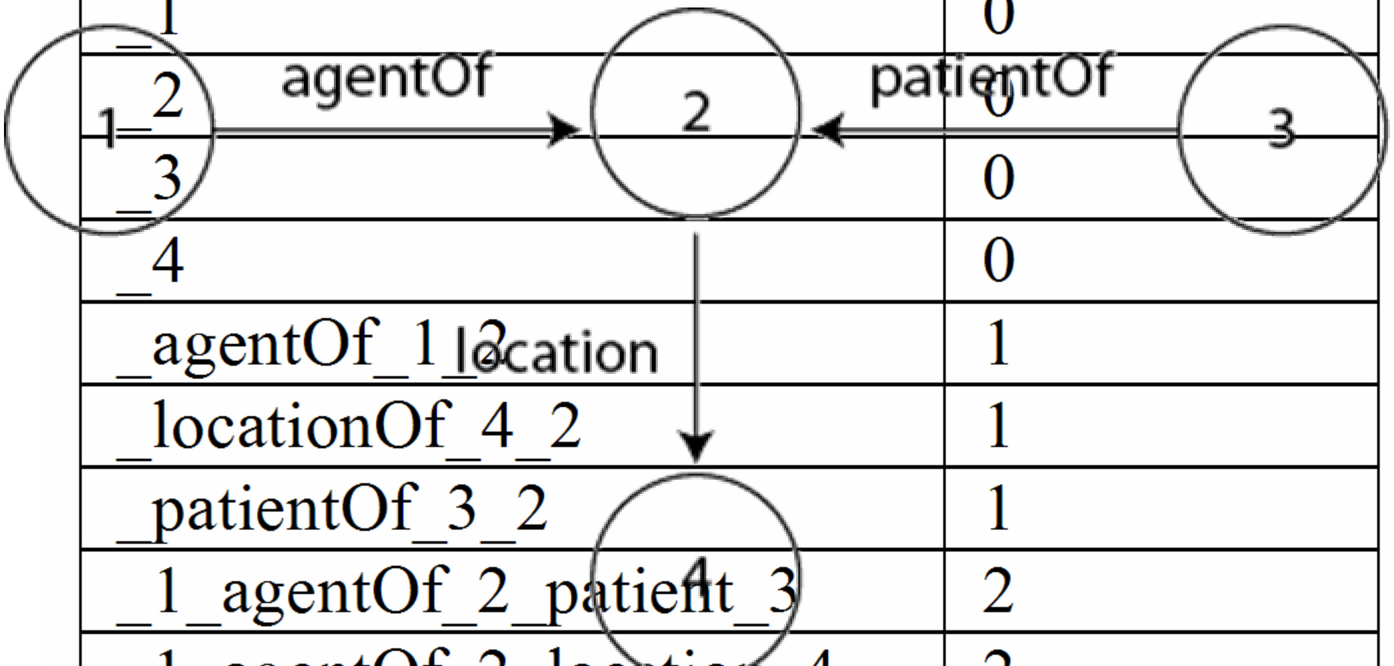
Indexing

Indexing is needed for

- Semantic Objects (SO), which are the Nodes
  - Fulltext Index
  - Node IDs
- Semantic Descriptions (SD), which are the Graphs
  - Based on the Paths in the Graphs
  - Paths as Strings

Paths in the Index

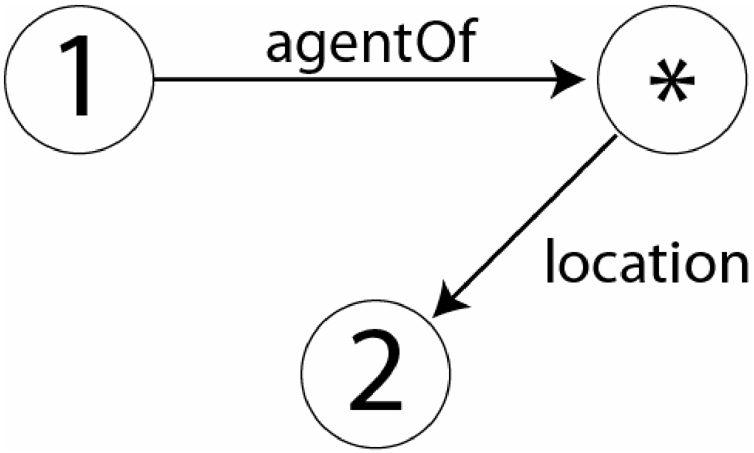
Term	Path length
_1	0
_2	0
_3	0
_4	0
_agentOf_1_2	1
_locationOf_4_2	1
_patientOf_3_2	1
_1_agentOf_2_patient_3	2
_1_agentOf_2_location_4	2
_3_patientOf_2_location_4	2



Constructing a Query (1/2)

Query: *Mathias Lux* is doing *something* at the *I-Know*

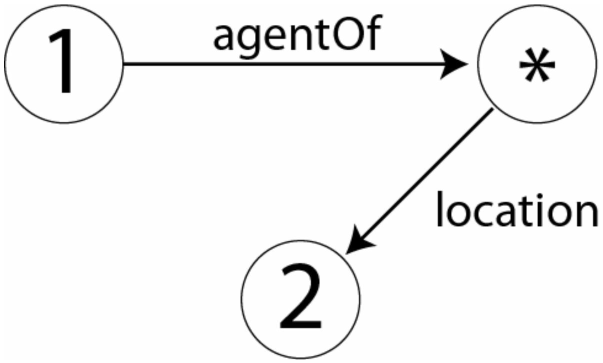
- *Mathias Lux* is identified as node with ID 1
- I-Know is identified as node with ID 2
- „Something“ is a wildcard



Constructing a Query (2/2)

The Query is based on the paths:

- Paths with Length 0:
  „_1“ and „_2“
- Paths with Length 1:
  „_agentOf_1_*“ and „_locationOf_2_*“
- Paths with Length 2:
  „_2_agentOf_*_location_2”



http://www.know-center.at

Query Interface

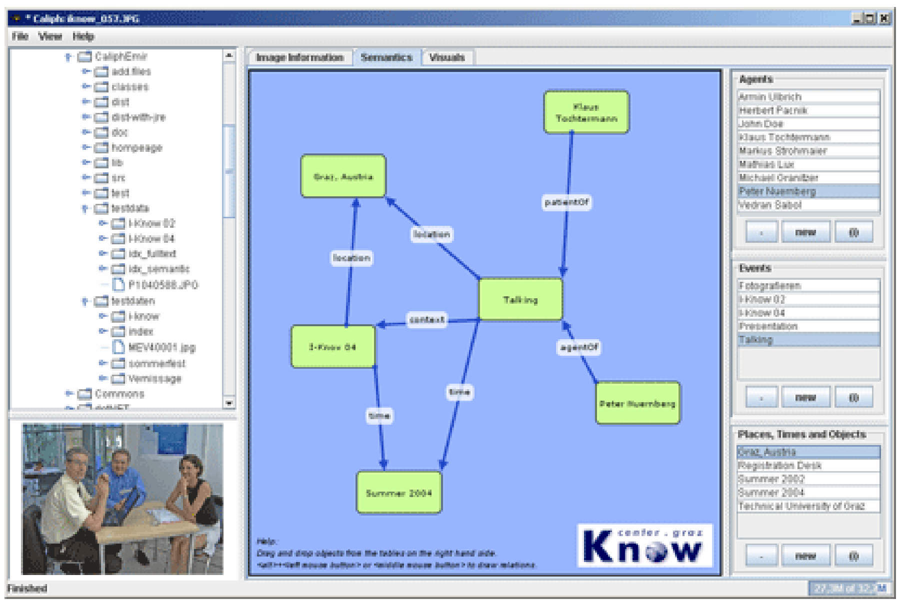
Creating Queries is like drawing Graphs.

- Possibe Candidates for Nodes are identified.
- With Query Expansion Query Graphs are constructed.
- From each Query Graph a Query String is constructed.
- Support for Wildcards

Implementation (1/2)

Open Source Applications for Annotation and Retrieval of Digital Photos: Caliph & Emir

Caliph: **C**ommon **a**nd **Li**ght Weight **Ph**oto Annotation

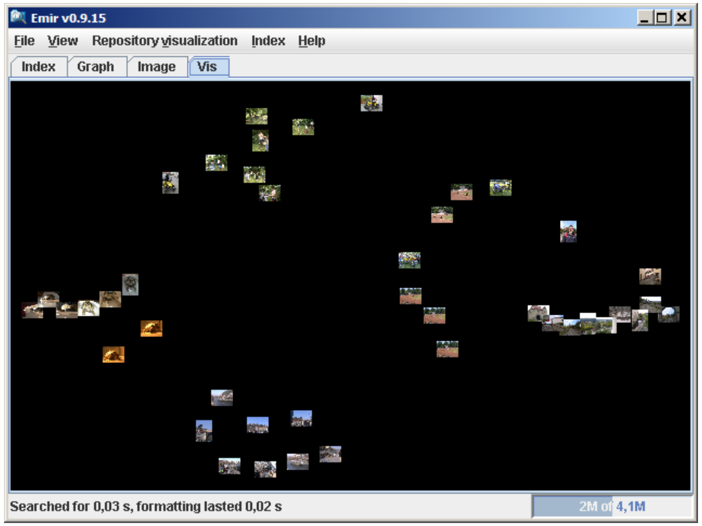


http://www.**know-center**.at

Implementation (2/2)

Emir: **E**xperimental **M**etadata Based **I**mage **R**etrieval

- Similar Images (CBIR)
- Keywords in Full Text Index
- Semantic Descriptions
- Visualization based on:
  - CBIR (Color, Edges)
  - Similarity of Semantic Graphs



Evaluation

- Definition of a Test Set
  - No Standardized Test Set available
- Evaluation of Retrieval Performance compared to
  - Full Text Search
  - Different Ranking (Scoring) Algorithms
- Assumption: Maximum Common Subgraph Metric yields Optimal Results

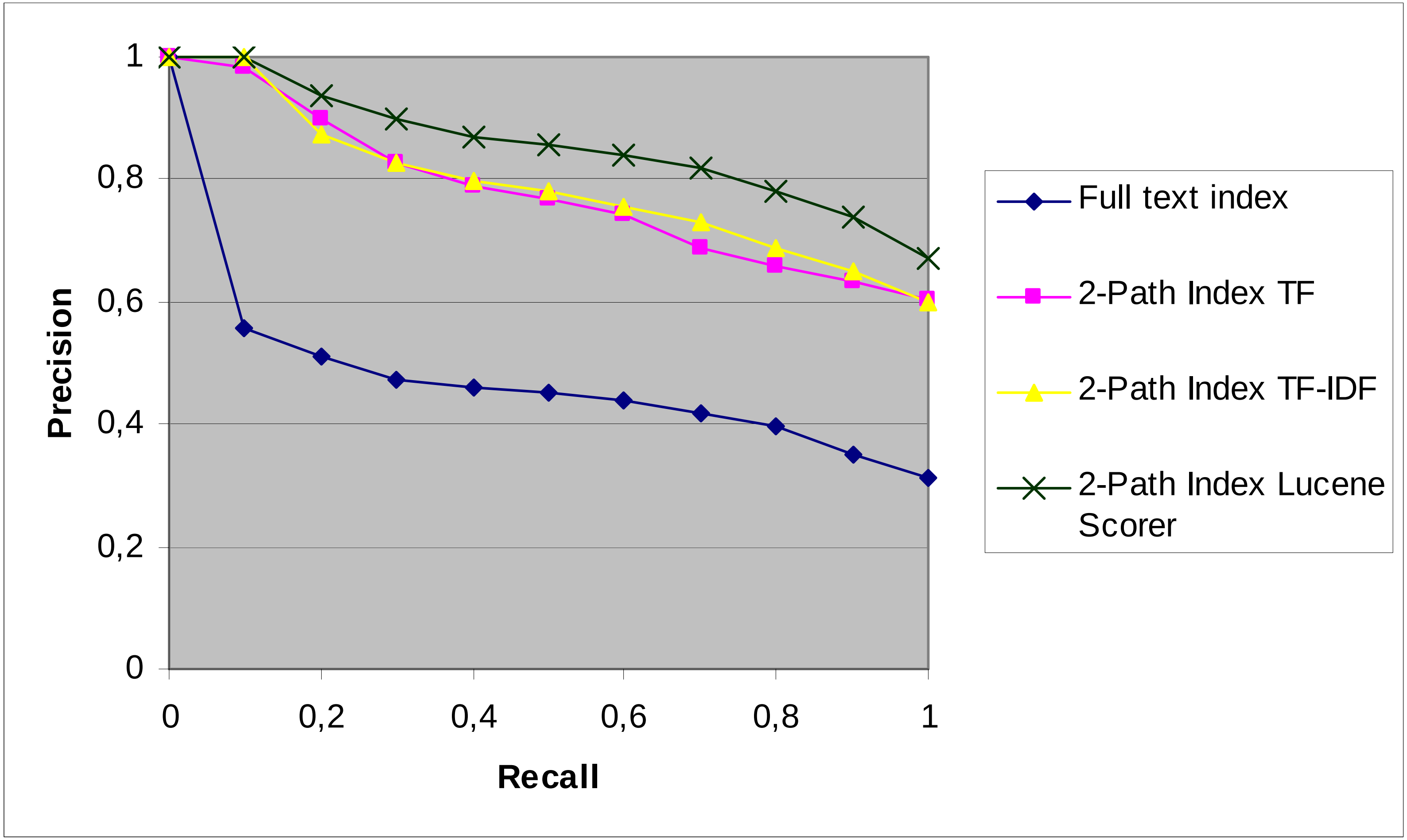
$$similarity(G_1, G_2) = \frac{|mcs(G_1, G_2)|}{\max(|G_1|, |G_2|)}$$

Test Set

- 85 different Semantic Descriptions
- Photos of I-Know Conferences in 2002 and 2004
- Each Graph from the database was taken to query the whole database, precision and recall were averaged

	Min	Max	Median
Nodes	3	11	5.5
Relations	2	12	5.6

Auswertung


1
0,8
0,6
0,4
0,2
0
Precision
0
0,2
0,4
0,6
0,8
1
Recall
Full text index
2-Path Index TF
2-Path Index TF-IDF
2-Path Index Lucene Scorer

Lucene Scoring Function

$$score(q,d) = \sum_{t \in q} TF(t,d) \cdot IDF(t) \cdot b(t.field,d) \cdot lNorm(t.field,d) \cdot coord(q,d) \cdot qNorm(q)$$

Lucene Scorer:

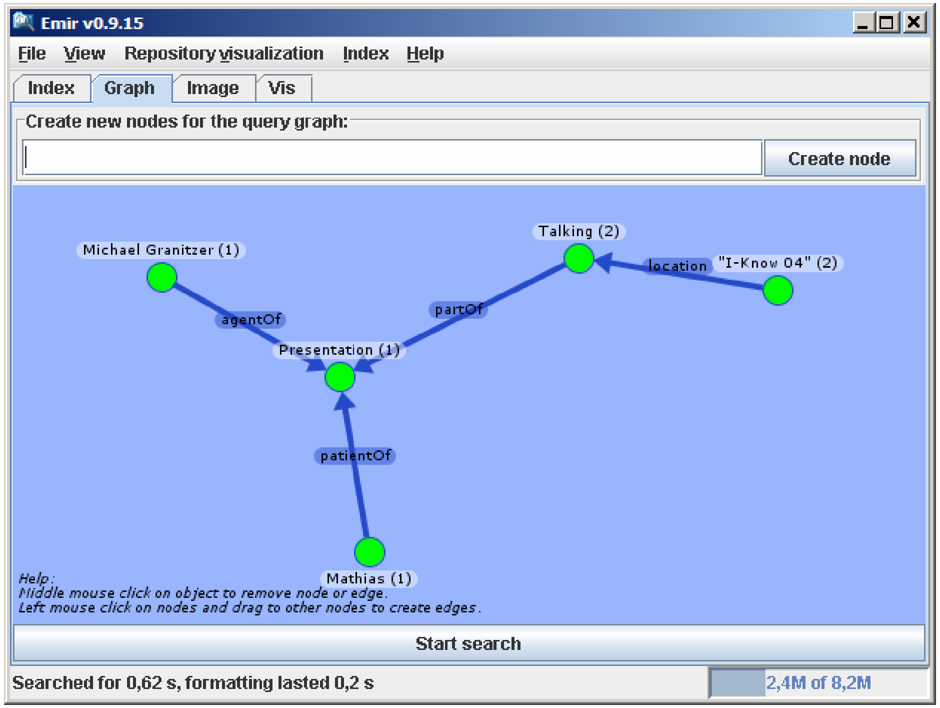
- TF ... Term Frequency
- IDF ... Inverse Document Frequency
- b ... Boost Value
- lNorm ... Normalization based on Field Value Length
- qNorm ... Normalization based on Query
- coord ... Term Frequency in Query and Document

Evaluation Results

- The Path Index based Retrieval outperforms the Full Text Retrieval on this test case.
- Between classical TF*IDF implementation and the term frequency scoring function only slight differences in retrieval performance can be identified.
- We assume that the *coord(q, d)* factor is the reason for the different performance of the classical TF*IDF and the Lucene score function by reflecting the denominator of the maximum common distance metric.

Demonstration

Emir: Experimental Metadata based Image Retrieval



Future Work

- Creating an appropriate Test Set
- Advancing Clustering and MDS Algorithms
- Integration of different Similarity Metrics for Evaluation
  - Path based ST Model
  - Error Correcting Maximum Common Subgraph Metric
  - Different Path Lengths, Selection with TF*IDF
- Implementation for RDF Graphs / OWL

Thank you …

… for your attention!

Visit Caliph & Emir:
http://caliph-emir.sourceforge.net