

# ***Datalog for the Web 2.0: the case of Social Network Data Management***



`follower(Luke, X)`

`close-friends(Danny, X)`

`close-friends(Matthew, X), profile-picture(X, P)`

`conversation(X, [President, Obama, Dalai, Lama])`



*Danilo Montesi, Matteo Magnani*

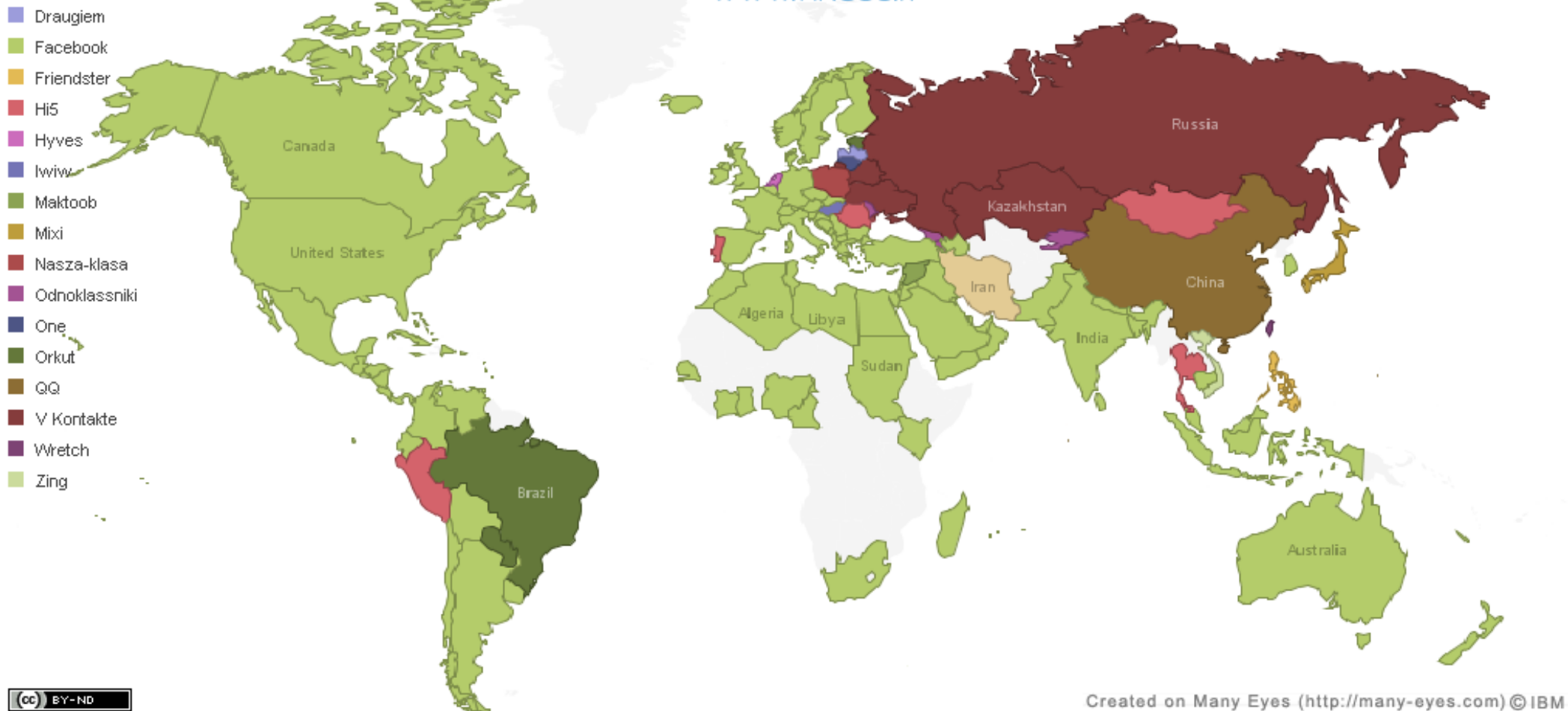
University of Bologna, Dept. of Computer Science  
*montesi @ cs.unibo.it --- matteo.magnani @ cs.unibo.it*



Research project partially supported by Telecom Italia

# WORLD MAP OF SOCIAL NETWORKS

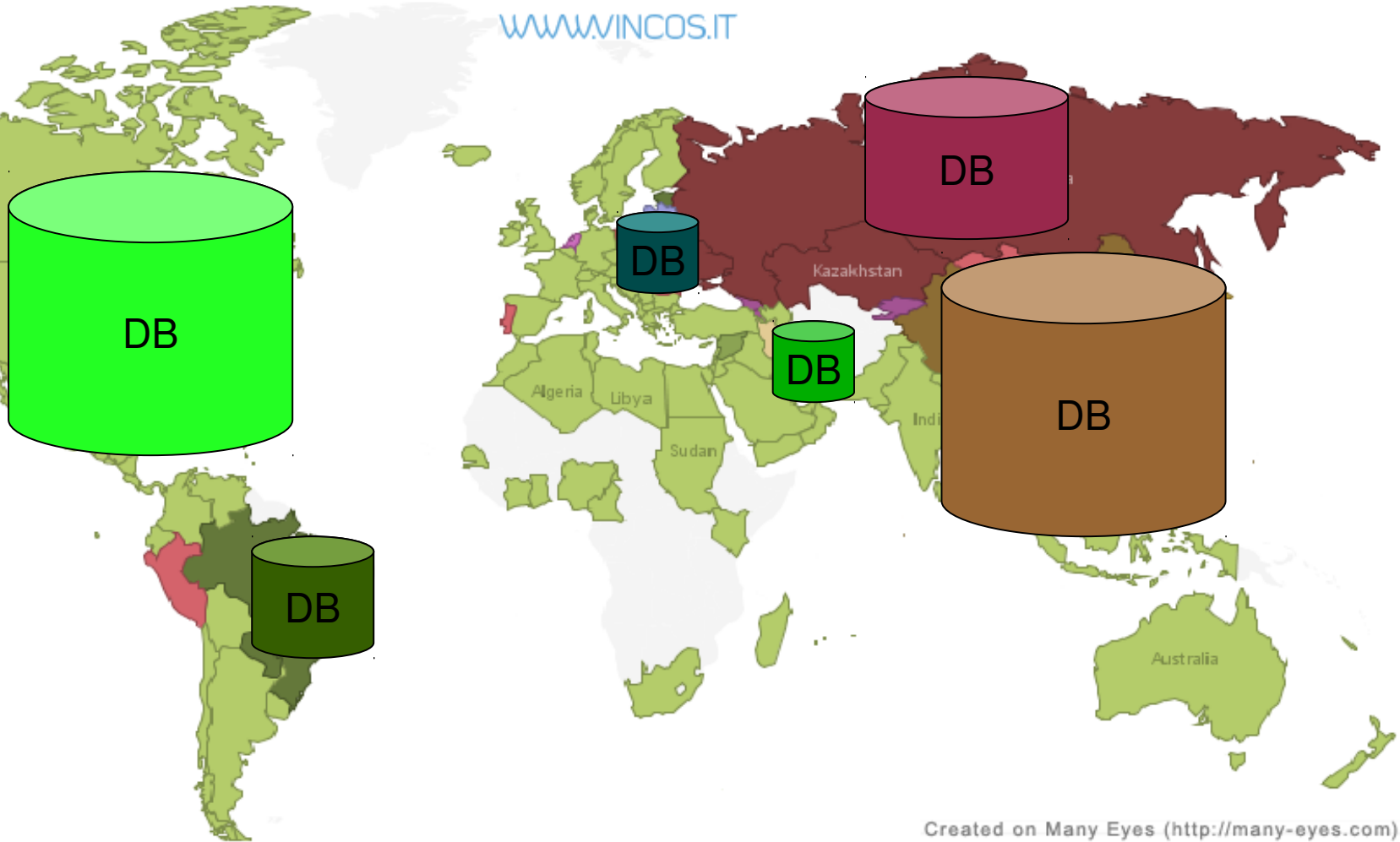
WWW.VINCOS.IT



# WORLD MAP OF SOCIAL NETWORKS

WWW.VINCOS.IT

- Draugiem
- Facebook
- Friendster
- Hi5
- Hyves
- Iwiw
- Maktoob
- Mixi
- Nasza-klasa
- Odnoklassniki
- One
- Orkut
- QQ
- V Kontakte
- Wretch
- Zing



CC BY-ND

Created on Many Eyes (<http://many-eyes.com>) © IBM

# Social Network Sites are Large Databases...

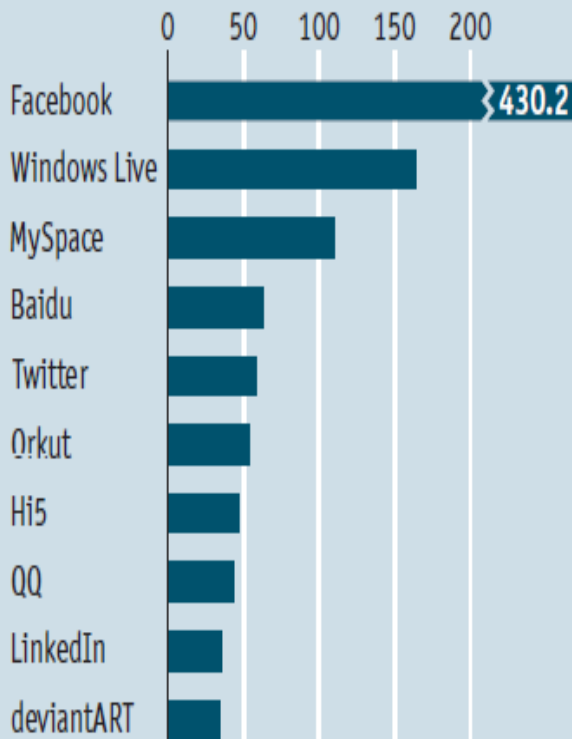
- ...with some peculiar features.
  - Global phenomenon, with specific sites expressing cultural diversities:
    - Facebook (>300 400.000.000 users)
    - QQ (>350.000.000 users).
  - Contain relevant information:
    - Terroristic attack, Mumbai, 2008.
    - Twitter revolution, Iran, 2009.
  - Enable relevant applications:
    - Politics, marketing, ...



# Some figures

## 1 Who will be my friend?

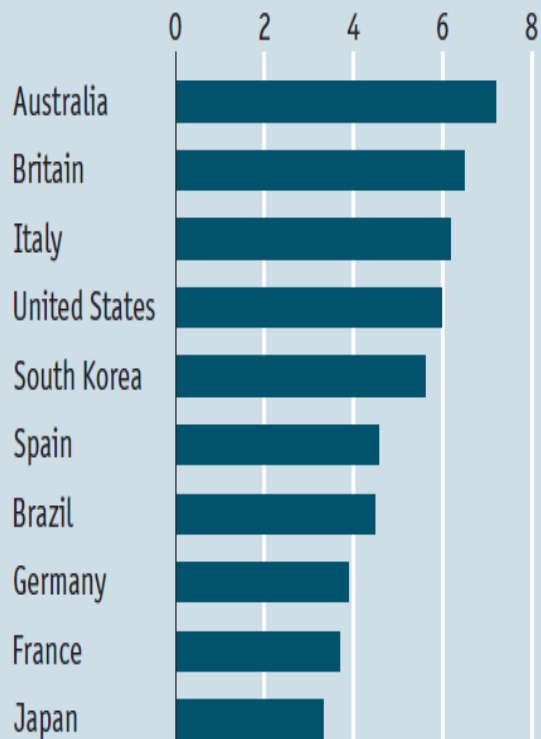
Social-networking sites, total unique visitors  
October 2009, m



Source: comScore

## 2 Sociable types

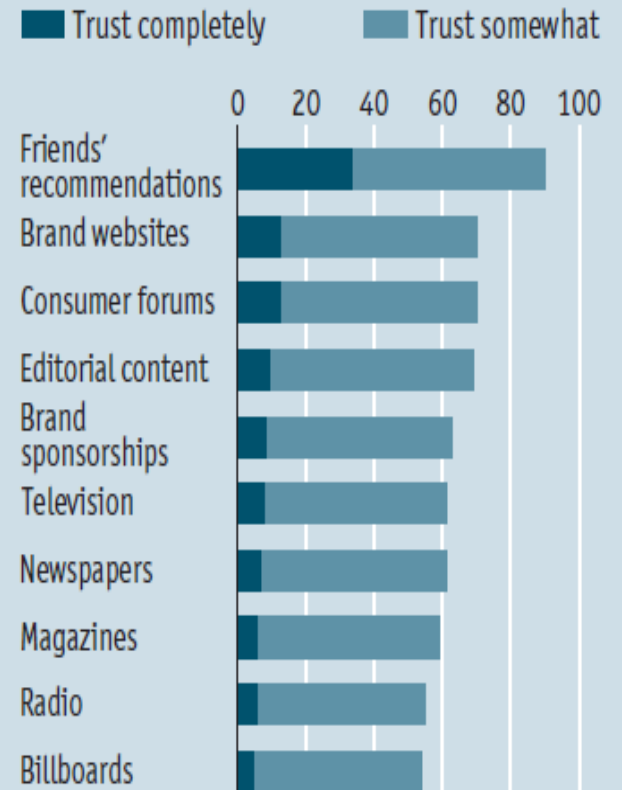
Average time spent on social-networking sites  
October 2009, hours per user



Source: Nielsen

## 5 In friends we trust

Global consumer trust in advertising, Q1 2009  
% of respondents



Source: Nielsen

# Social databases

- To support social network sites, relational databases are enough.
  - To enable *advanced applications*, we need specific data models and query languages:
    - \_ EXAMPLE 1: Real-time conversation-aware search engines.
      - Should consider user popularity, networks of messages on related topics, ...
    - \_ EXAMPLE 2: Message propagation prediction.
      - Should consider strength of relationships, probability of replying, ...
  - Social data structures are often based on networks.
- Datalog: good candidate.
- \_ Clean semantics.
  - \_ Recursive queries.
- However, social databases and applications require a number of features.
  - In this talk we examine the requirements of a Social Query Language (SocQL), to encourage a discussion on the features Datalog should have to be used in this real and relevant context.

FriendFeed is the

# Our case study: Friendfeed

in FriendFeed

Everyone

**Go To Talk Legal Channel**High Court Punts on Redskins Dispute - <http://blogs.wsj.com/law...>

17 seconds ago from WSJ.com: Law Blog - Comment - Like - Share

The NFL's Washington Redskins on Monday got a bit of good news from the U.S. Supreme Court, which declined cert filed by Native American activists who claim the Redskins' team name is so offensive that it does not deserve trademark protection.

- Recently acquired by Facebook.
  - Aggregates social content from Facebook, Twitter, Blogs.
  - Allows microblogging, but also complex conversations.
- contains data representative of different social behaviors.
- Small (about 5 million messages per week)
- good for a case study.
- Public APIs.
- most of data accessible.

# Data extraction

- API monitored from Sep. 6, 2009 to Sep. 19, 2009.
- Stored all entry IDs.
- Retrieved all associated comments and likes, in XML, and exported to CSV files.
- Retrieved the network of followers (friends), starting from active users and reconstructing the graph of all connections.

# Social data model

**Entry**(PostID, PostedBy, Timestamp, Text, Language)

**Comment**(PostID, EntryRef, PostedBy, Timestamp, Text, Language)

**Like**(User, EntryRef, Timestamp)

**User**(ID, Name, Description)

**UserLink**(Follower, Followed)

# Social data model

**Entry**(PostID, PostedBy, Timestamp, Text, Language)

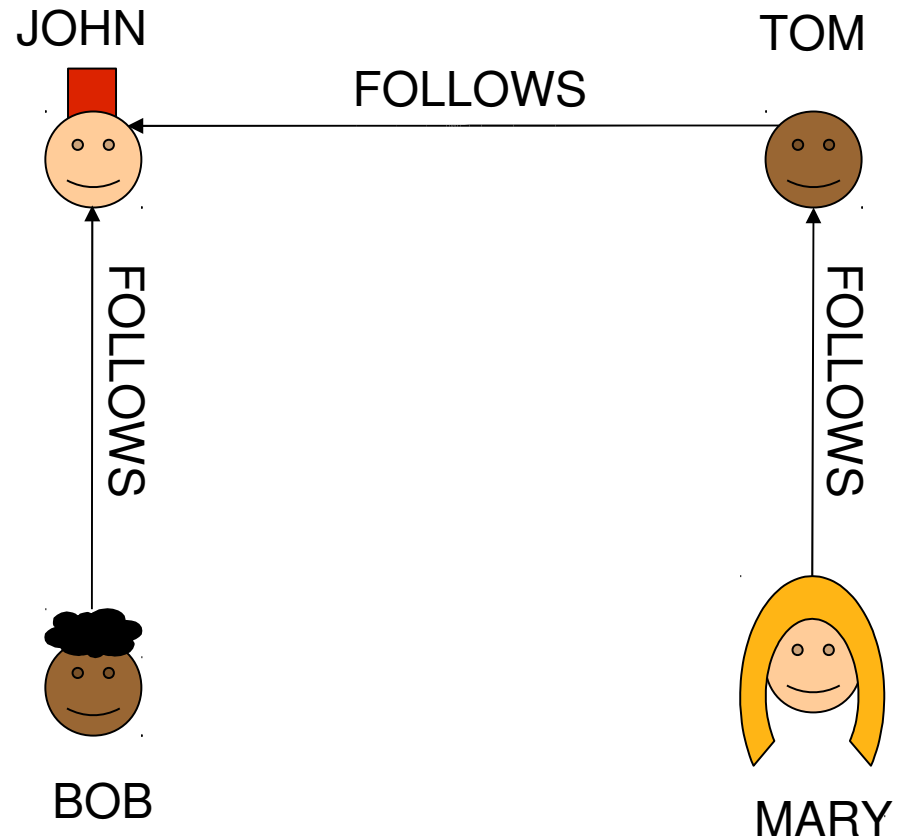
**Comment**(PostID, EntryRef, PostedBy, Timestamp, Text, Language)

**Like**(User, EntryRef, Timestamp)

**User**(ID, Name, Description)

**UserLink**(Follower, Followed)

- Network of users.



# Social data model

**Entry**(PostID, PostedBy, Timestamp, **Text**, Language)

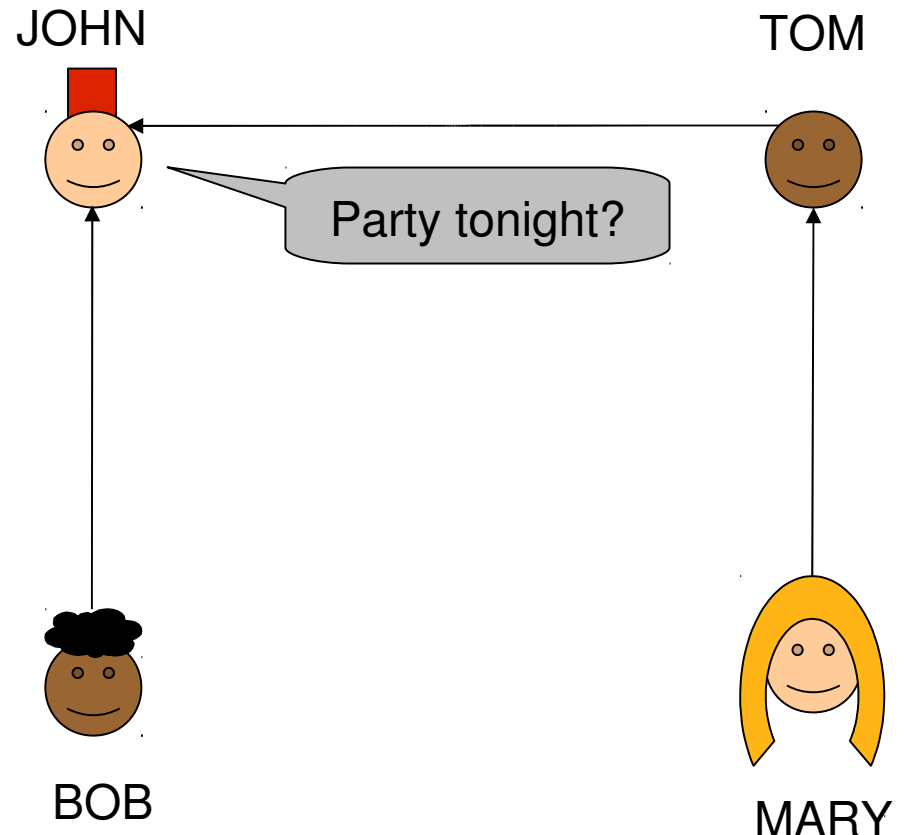
**Comment**(PostID, EntryRef, PostedBy, Timestamp, Text, Language)

**Like**(User, EntryRef, Timestamp)

**User**(ID, Name, Description)

**UserLink**(Follower, Followed)

- Network of users.
- Unstructured content.



# Social data model

**Entry**(PostID, PostedBy, Timestamp, Text, Language)

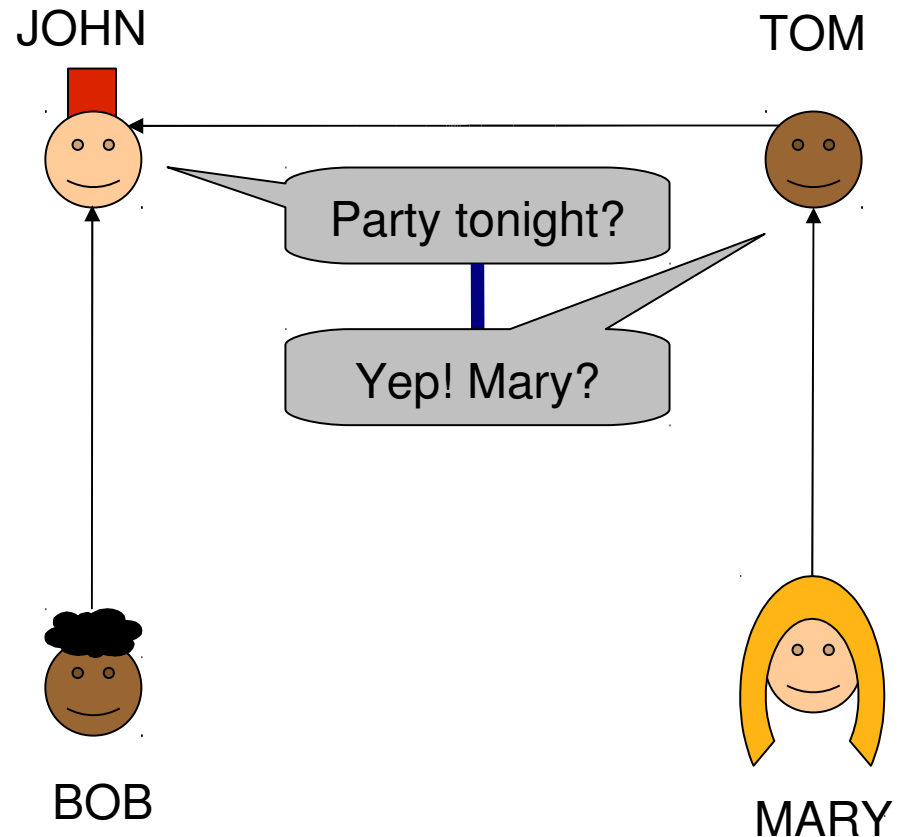
**Comment**(PostID, **EntryRef**, PostedBy, Timestamp, Text, Language)

**Like**(User, EntryRef, Timestamp)

**User**(ID, Name, Description)

**UserLink**(Follower, Followed)

- Network of users.
- Unstructured content.
- Chains of messages (conversations).



# Social data model

**Entry**(PostID, PostedBy, Timestamp, Text, Language)

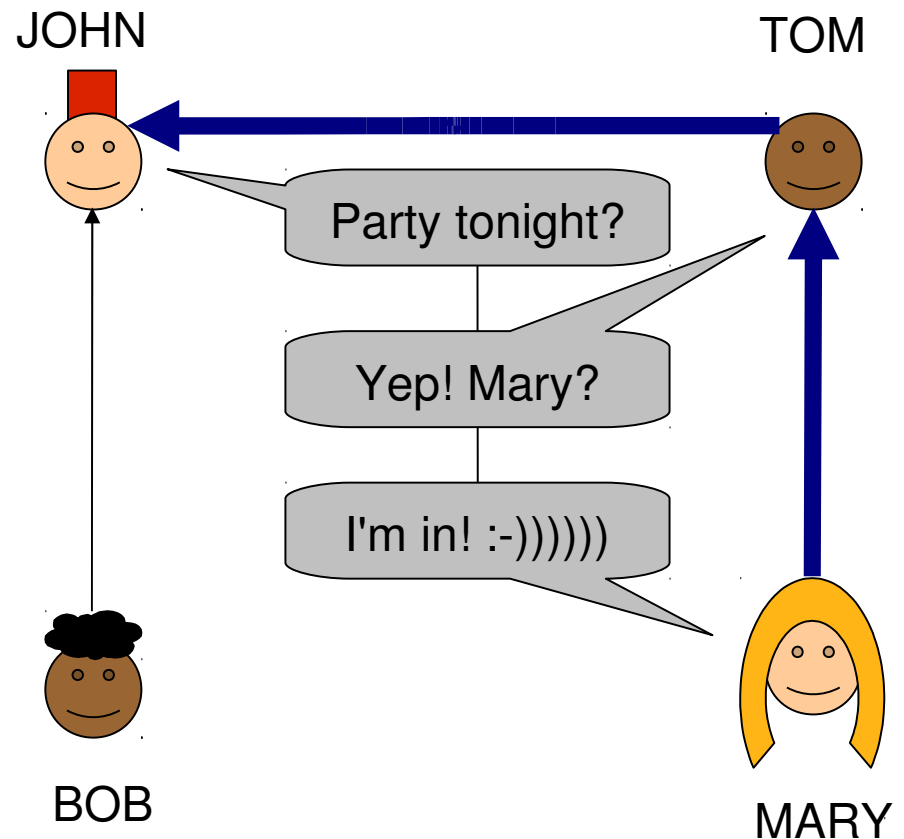
**Comment**(PostID, EntryRef, PostedBy, Timestamp, Text, Language)

**Like**(User, EntryRef, Timestamp)

**User**(ID, Name, Description)

**UserLink**(Follower, Followed)

- Network of users.
- Unstructured content.
- Chains of messages (conversations).
- Transitive connections.



# Social data model

**Entry**(PostID, PostedBy, Timestamp, Text, Language)

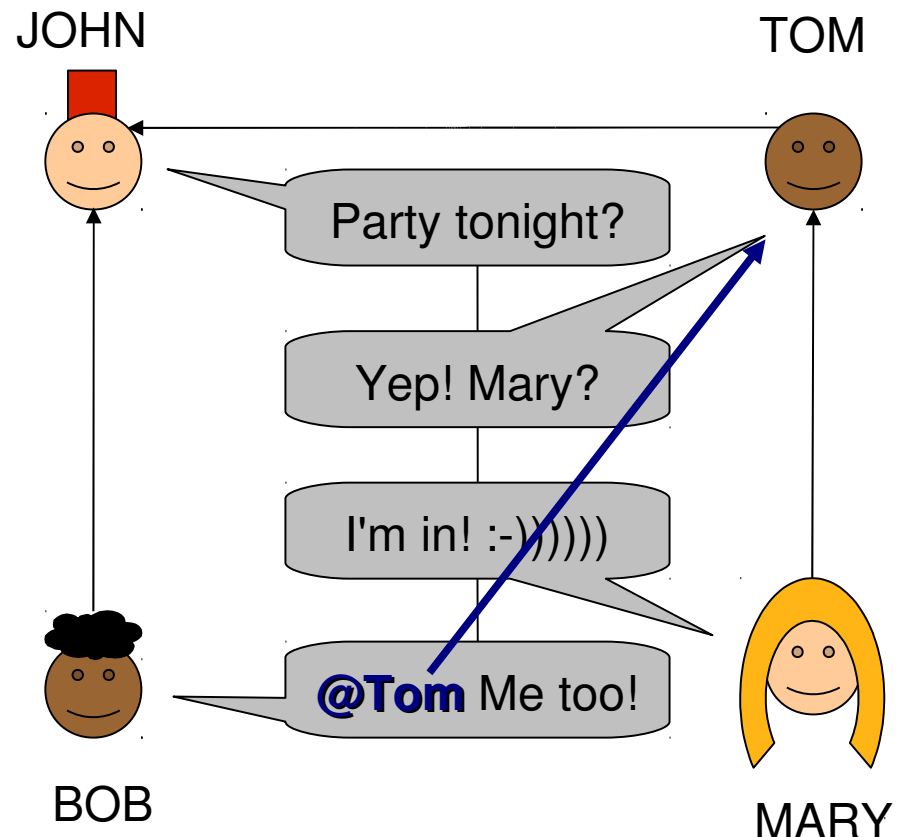
**Comment**(PostID, EntryRef, PostedBy, Timestamp, **Text**, Language)

**Like**(User, EntryRef, Timestamp)

**User**(ID, Name, Description)

**UserLink**(Follower, Followed)

- Network of users.
- Unstructured content.
- Chains of messages (conversations).
- Transitive connections.
- Implicit references (through data).



# Data requirements: size

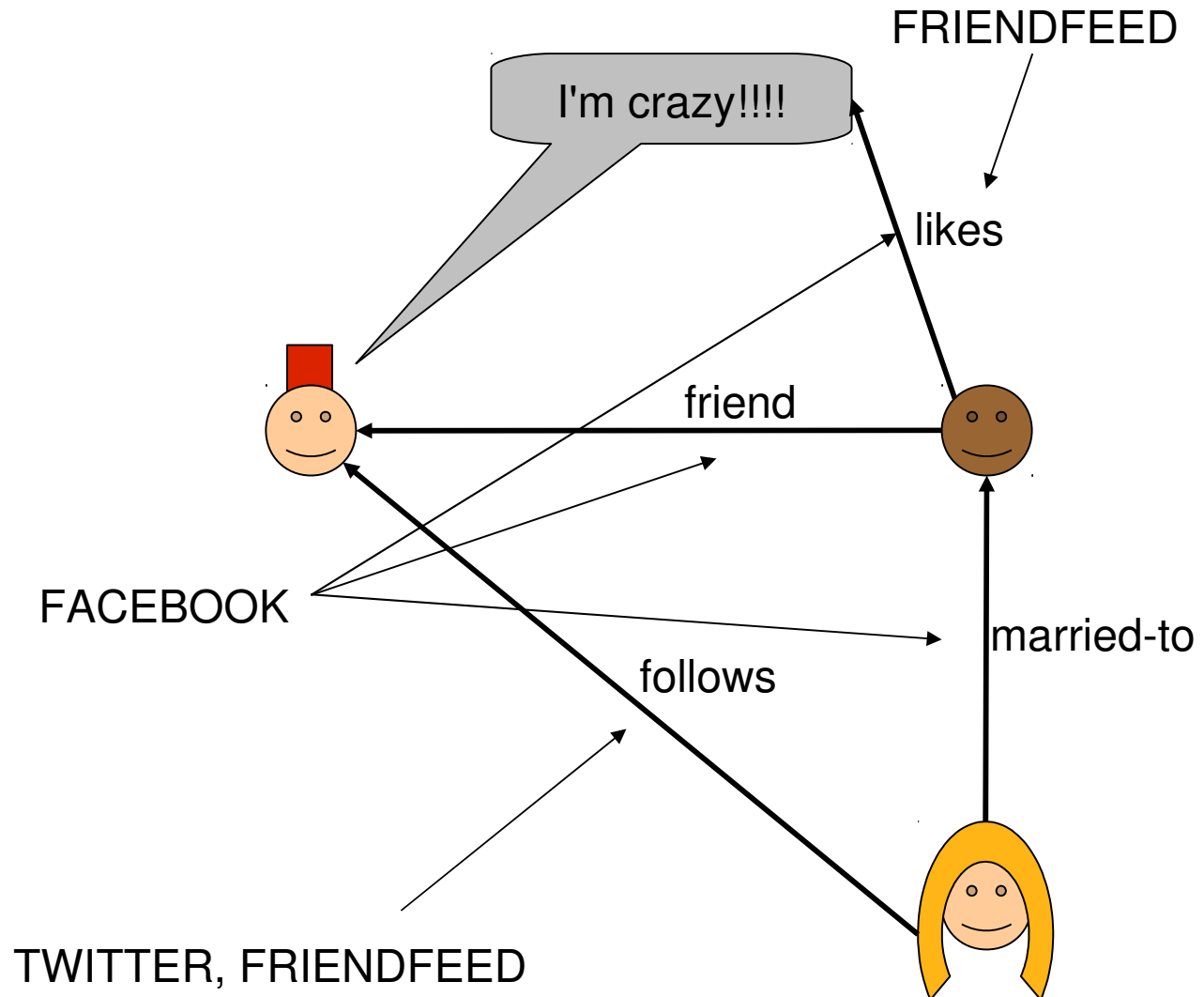
- Order of  $10^{6-7}$  records per week.
- About 500.000 users.
- About 1 GB (text only) per week.
- In addition to multimedia content.
- (Friendfeed has 1/1000 users w.r.t. Facebook...).
- Datalog implementations should be based on existing relational technologies and systems.

# Data requirements: structured, semi-structured and unstructured

- Relations.
- Text and multimedia.
- Multiple graphs: users, messages.
- Graphs are the main motivation behind a Datalog-based social query language.
- However, atoms can be complex (more later on this...).
- Timestamps.

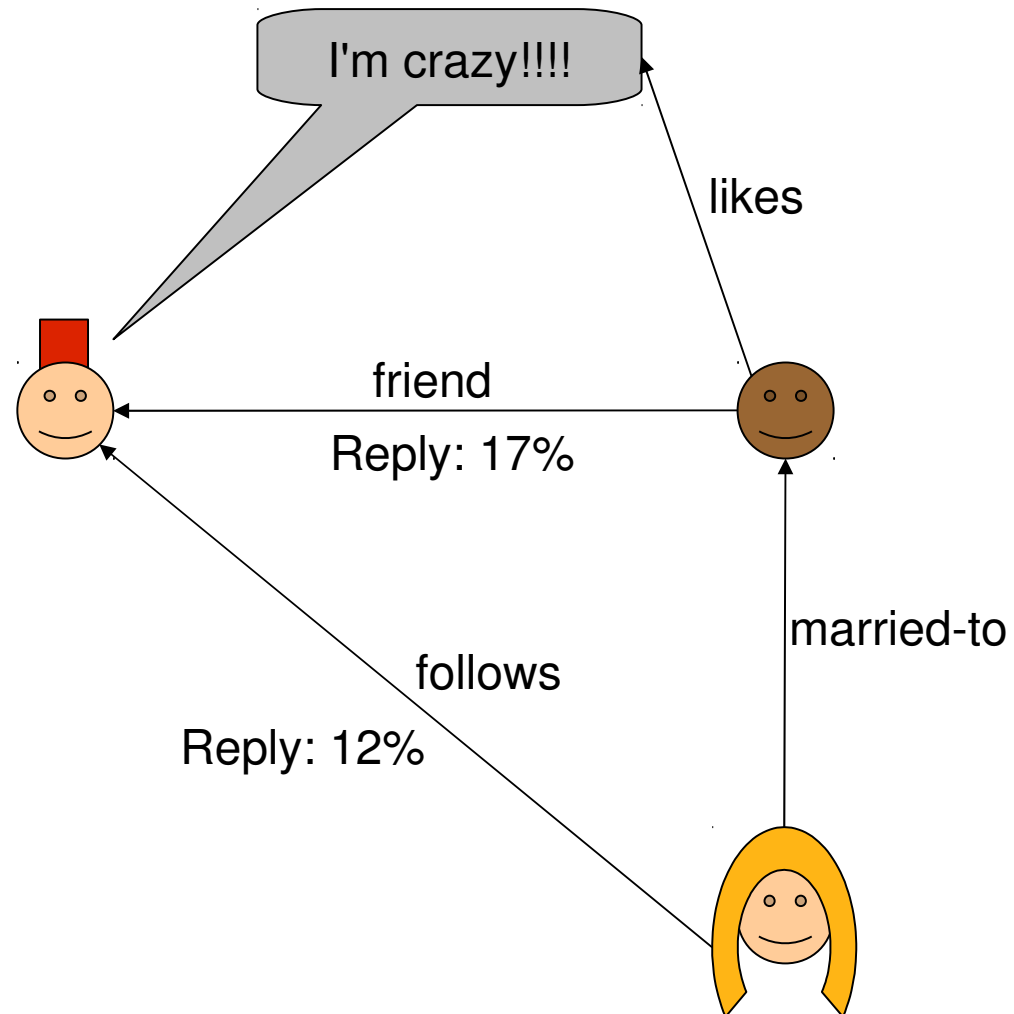
# Data requirements: labeled and weighted edges

- Different kinds of edges.
  - \_ Different behaviors.
  - \_ Different relationships.



# Data requirements: labeled and weighted edges

- Different kinds of edges.
  - \_ Different behaviors.
  - \_ Different relationships.
- Weights to indicate properties necessary to analyze the data and make predictions.
  - \_ Strength of connection.
  - \_ Probability of replying (w.r.t. topic).



# Operational requirements: recursive traversal

- Many properties do not regard just direct connections.
  - E.G., user popularity is important to rank messages and conversations.
  - A user with even a single follower who is very popular will be popular as well (her messages will be seen by many users when commented).
- Datalog already supports this kind of queries.
  - Who may see a message posted by Bob?

# Operational requirements: aggregation

- Queries may involve sub-graphs.
- E.G., the network of MY friends.
  - Return the probability that a message on *technology* will be commented.
- Aggregate functions are needed to compute these queries.
- Aggregations should work on numbers, and also edge types.
  - How many connections of each kind (follower, followed, ...) do I have?

# Operational requirements: keyword search and conversation retrieval

- Content is important.
  - However, a lot of content can (sometimes) be considered noise: :-), :-D, lol, rotfl, ok, ciao...
  - Often messages are short, and thus different from documents in traditional IR.
- User's relevance is important as well in ranking.
- The same people exchanging the same message at different times may have different rankings.
  - Return all conversations about Berlusconi, ordered by their relevance.

# Operational requirements: data analysis and exploratory queries

- A lot of *basic* information is not explicitly exposed by the data model.
- Vague queries based on data *analysis* capabilities.
  - Return all my friends, grouped by their degree of friendship.
  - (involves graph clustering)
  - May return: (Bob, Mary [strong], John [weak])
- In addition, *exploratory* and *interactive* queries, typical of semistructured data models.
  - Return my followers, then show me their names. Select some of them, and expand their connections...

# Summary (1)

Data Model	
Large data sets	Order of $10^{6-7}$ records per week. It will be probably necessary to build social extensions of Datalog over existing relational systems.
Relations	This feature is already supported by Datalog.
Data graphs	The relevance of graph data can be one of the key motivations behind the adoption of Datalog as a social query language.
Arc weights	It is necessary to be able to manipulate weights, e.g., indicating the strength of a friendship relationship, according to some model to be defined.
Arc labels	As we have seen in our example data graph, different arcs should be treated differently depending on their labels. Labels may indicate an arc type, but also contain unstructured text.

# Summary (2)

Operations	
Recursive traversal	This feature, which is associated to the graph model, is easily supported by Datalog, and could constitute one of its strengths.
Aggregates (weights)	Dealing with weighted graphs, we may need to compute summary metrics of sub-graphs, involving the aggregation of floating point numbers.
Aggregates (labels)	In addition to the aggregation of weights, it is important to be able to evaluate aggregate metrics concerning labels. For example, counting all arcs of a given type.
Keyword search	Data graphs contain a lot of unstructured text. Therefore, queries should necessarily provide Information Retrieval capabilities.
Graph ranking	In addition to basic keyword search, queries involving IR capabilities should also take the structure of the graph under consideration.
Data Analysis	Being very complex, a social data model contains a lot of information not directly exposed using its data structures but hidden inside them. Therefore, to be able to extract the required information it is often necessary to execute <i>exploratory</i> queries, based on data analysis functions such as graph clustering or sub-graph matching.

# Conclusion

- We have identified the requirements of a generic query language for Social Web 2.0 applications (SocQL).
- Studying a real and complex social application.
- Datalog is a potential candidate, because of its clean representation of recursive queries.
- But it should provide additional capabilities.
- A big opportunity to leave academia...

# ***Datalog for the Web 2.0: the case of Social Network Data Management***



`follower(Luke, X)`

`close-friends(Danny, X)`

`close-friends(Matthew, X), profile-picture(X, P)`

`conversation(X, [President, Obama, Dalai, Lama])`



*Danilo Montesi, Matteo Magnani*

University of Bologna, Dept. of Computer Science

*montesi @ cs.unibo.it --- matteo.magnani @ cs.unibo.it*



Research project partially supported by Telecom Italia