Università
di Modena
e Reggio
Emilia

Mobilware 2010

Agents and Pervasive Computing Group

A Self-Organizing Approach for Building and Maintaining Knowledge Networks

Gabriella Castelli, Marco Mamei, Franco Zambonelli

Università di Modena e Reggio Emilia, Italy

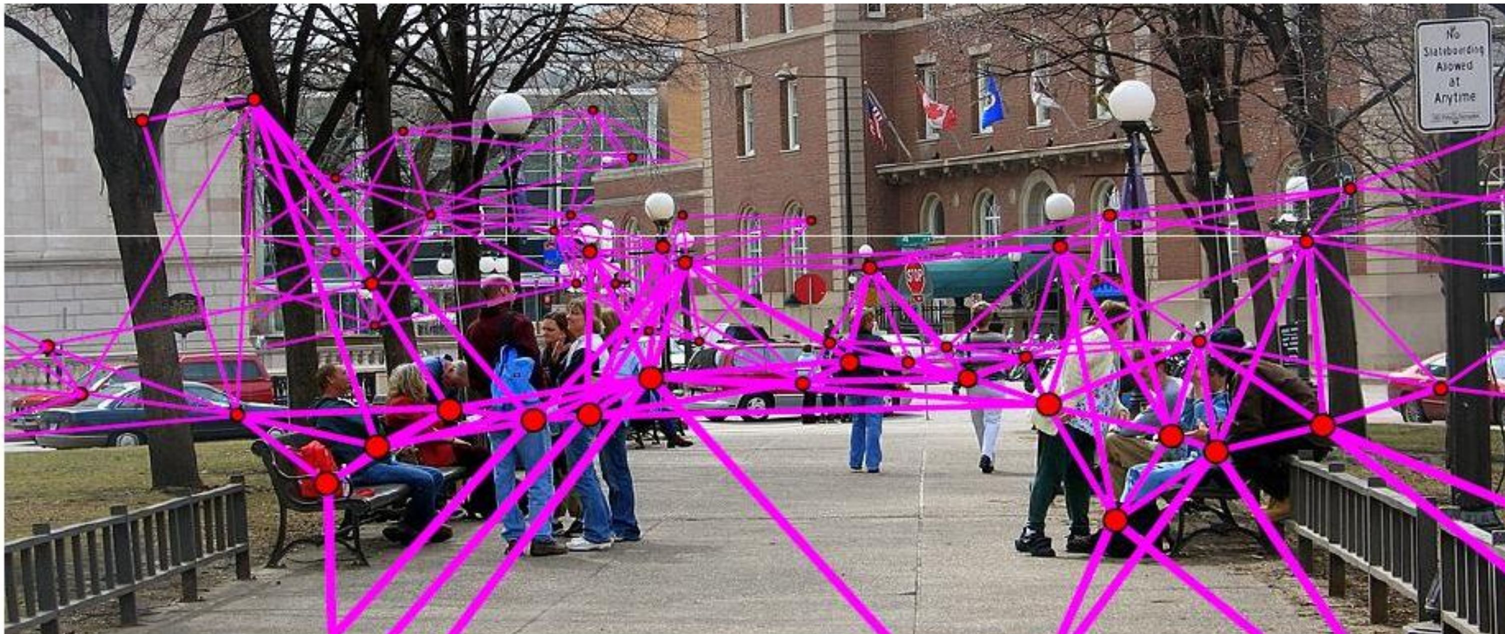
Outline

- Scenario and Motivation
- Context-Awareness Vs Situation Awareness
- Engineering challenges
- The W4 Context Model & API
- W4 Knowledge Networks
- Experimental Results
- Conclusion and Future Works

Università di Modena e Reggio Emilia

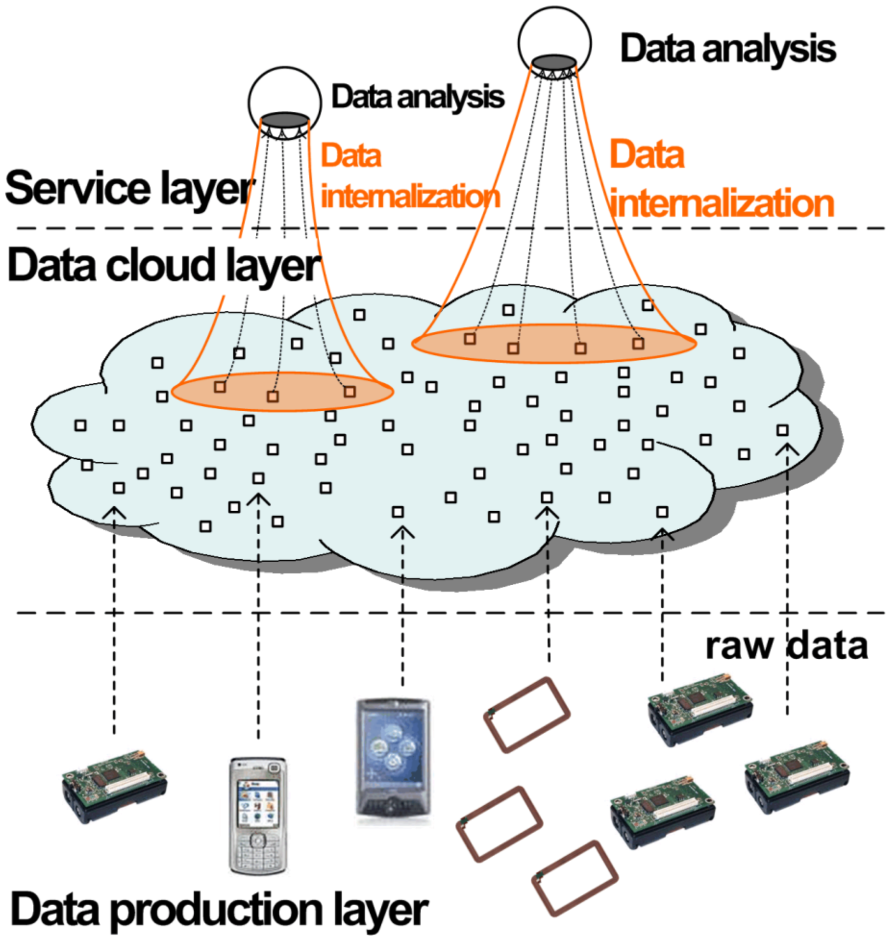
Agents and Pervasive Computing Group

Pervasive Scenario



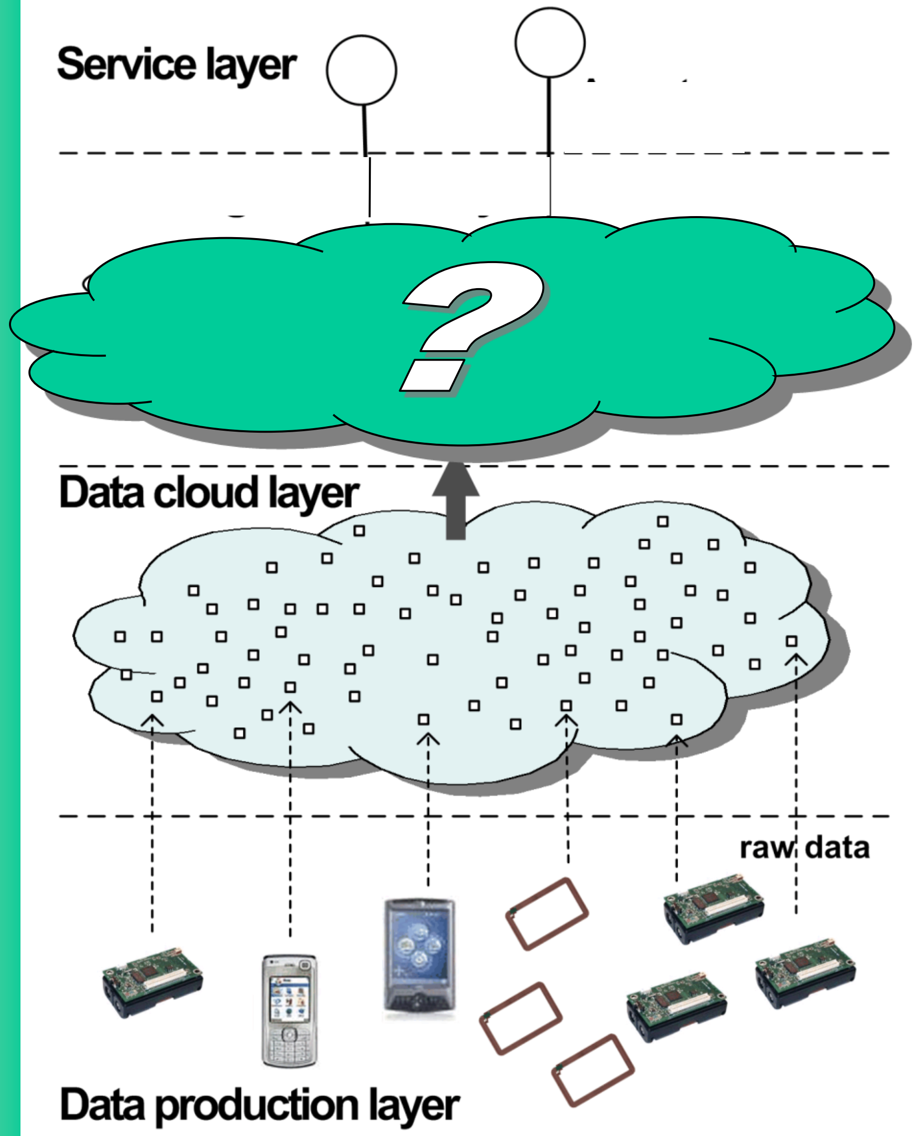
- Computer-based systems and sensors will be soon embedded in everywhere:
  - all our everyday objects
  - all our everyday environments (house, offices, cities)
- A truly pervasive network generating increasing amount of information: WSN, RFID, smartphones & GPS, Web 2.0
- The peculiar challenge is **turning such information into usable Knowledge.**

Context-Awareness



- Pervasive devices and sensors make available to service agents a sort of “data cloud layer”.
- A service needs to internalize data atoms from the cloud, analyze it to properly understand situations, and finally exploit such knowledge as needed for their own goals.

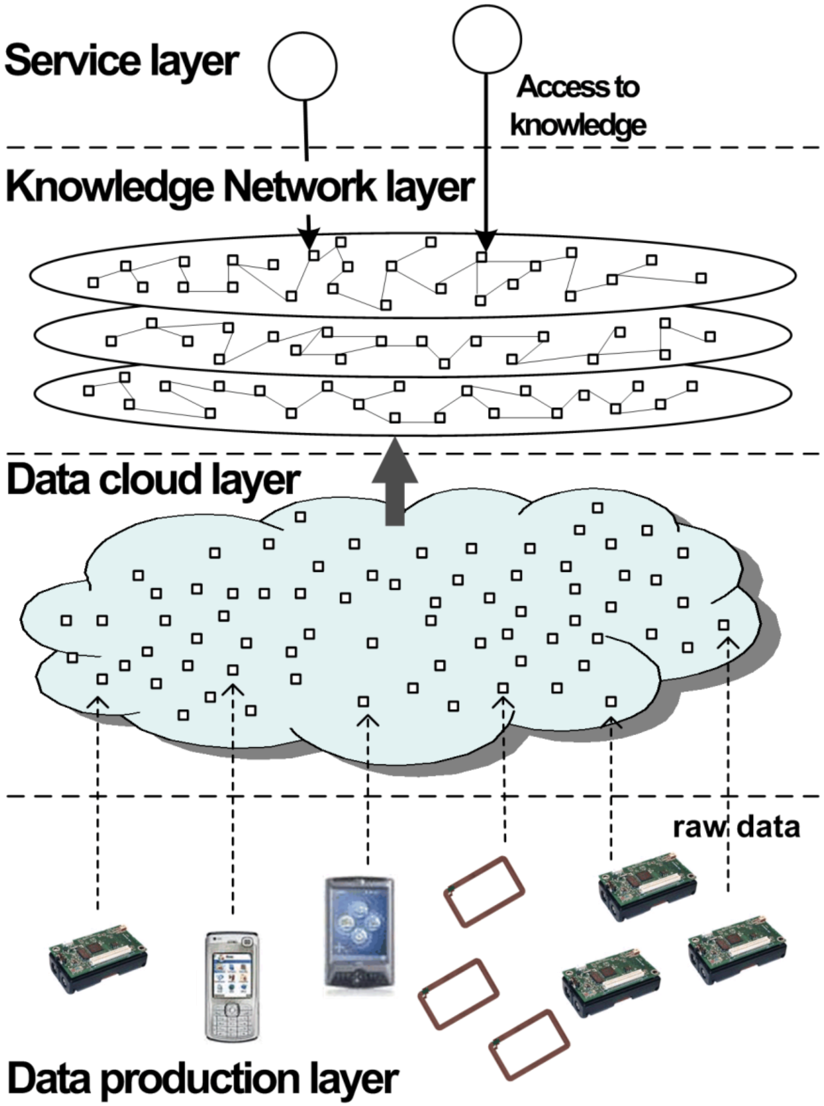
What do we need?



We have to **engineer the data environment:**

- in order to reduce the computational costs for service agents
- to provide the knowledge of what is happening around ("the situation")

Situation-Awareness



- By exploiting a **knowledge network** layer, agents are no longer forced to access the raw data cloud layer.
- Knowledge organization and analysis is externalized in the middleware, and agents are given access to pre-digested information, with a notable complexity reduction.

Università di Modena e Reggio Emilia

Agents and Pervasive Computing Group

Challenges

- **Data Model**:
  - a simple, general-purpose and uniform model to represent contextual information as individual data atoms as well as their aggregates.
- **Access to data**:
  - the very goal of knowledge networks is to digest data from any possible contextual data source and to provide knowledge to agents.
- **General Approaches for data aggregation and networking**.
  - It should be a “live layer” continuously and autonomously analyzing information to aggregate data atoms, relate existing knowledge atoms with each other, and extract meaningful knowledge from the available data.
- **Application-specific views**:
  - specific agents may require the dynamic instantiation within the knowledge networks of application-specific algorithms for knowledge analysis.

The W4 Context Model

- Someone or something (*Who*) does some activity (*What*) in a certain place (*Where*) at a specific time (*When*)

- **Who** is the subject. It is represented by a string with an associated namespace that defines the "kind" of entity that is represented.
  - "person:Gabriella", "tag:#567"
- **What** is the activity performed. It is represented as a string containing a predicate-complement statement.
  - "work:pervasive computing group", "read:temperature=23".
- **Where** is the location to which the context relates.
  - (longitude, latitude),
  - "campus", "here"
- **When** is the time duration to which the context relates
  - 2006/07/19:09.00am - 2006/07/19:10.00am
  - "now", "today", "yesterday", "before"

W4 API

- knowledge atoms will be stored in a **shared data space**, we took inspiration from tuple-space approaches:

```
void inject(KnowledgeAtom a);
KnowledgeAtom[ ] read(KnowledgeAtom a);
```

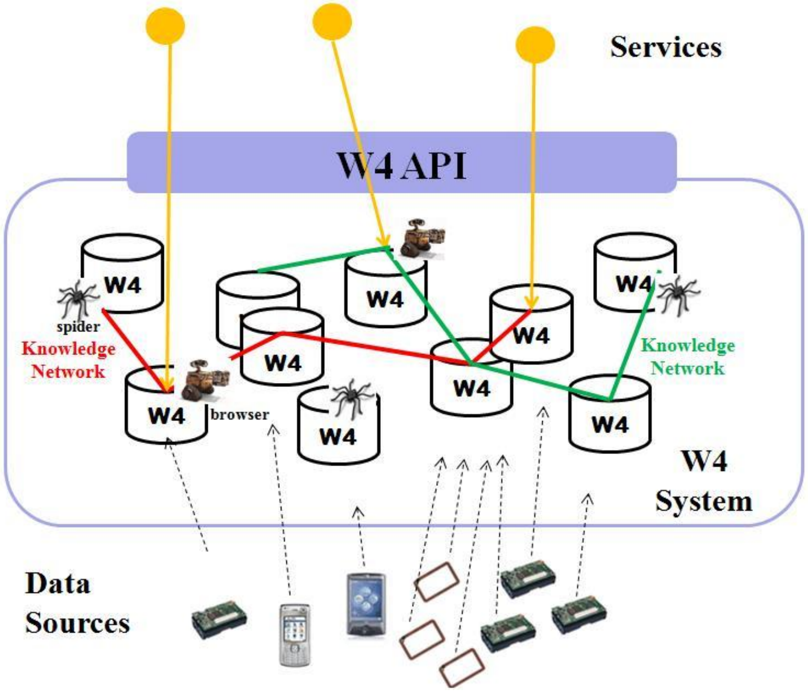
- Pattern matching on read operation depends on the semantics of the W fields

- In this way, agents can acquire contextual data by simply acting upon W4 shared tuples spaces

Università di Modena e Reggio Emilia

Agents and Pervasive Computing Group

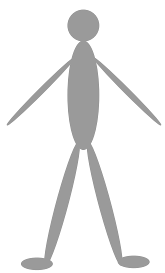
W4 Middleware Architecture



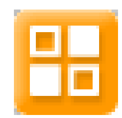
- I developed a W4 tuple space implementation on top of LighTS Tuple Space
- The W4 Middleware is implemented in Java. Java classes provides the support for the semantics introduced and the corresponding pattern matching operations.
- Fully implements the Data Model and the Knowledge Networks algorithms

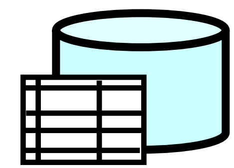
W4 Knowledge Networks

- The **data generation process** itself is the simples task of knowledge networking
  - in that it includes and the actions for relating individual atoms to increase their informative values.



UID = #567





Tag DB

Who: user:Gabriella
What: walk: 4km/h
Where: lngY, lat X
When: now

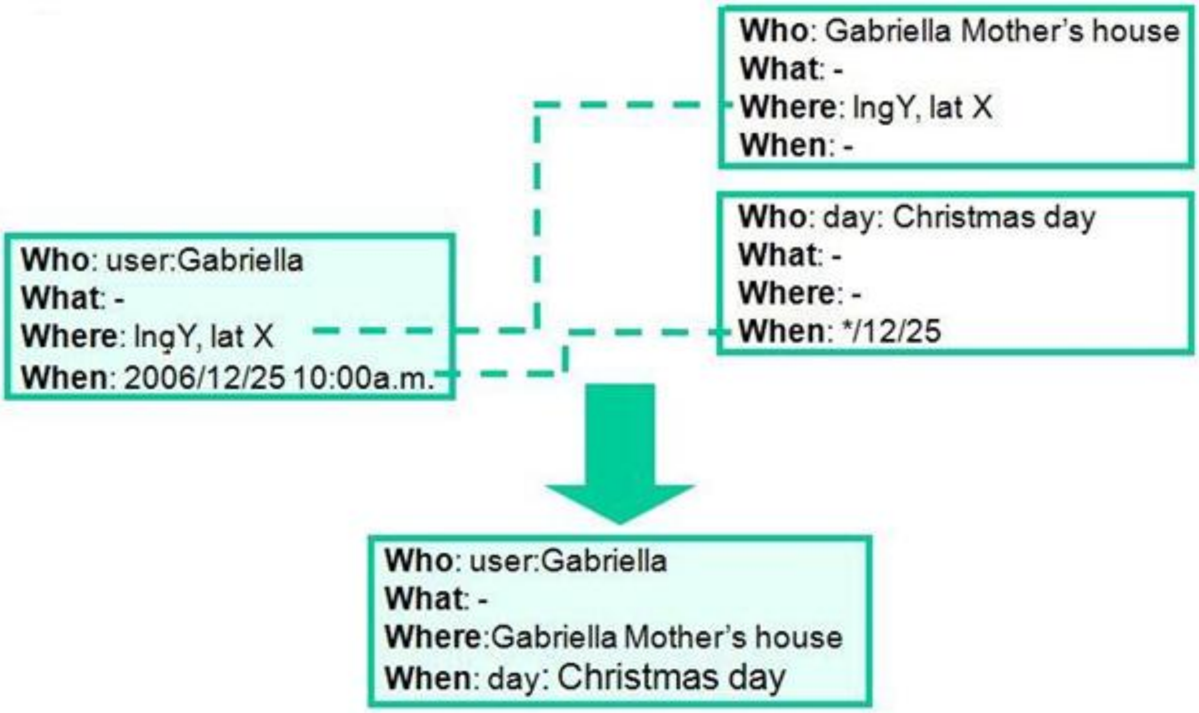
Who: tag:statue of Ariosto
What: -
Where: lng Y, lat X
When: now

Who: user:Gabriella
What: walk: 4km/h
Where: statue of Ariosto
When: now

W4 Knowledge Networks (2)

- Producing new information by navigating the knowledge network and **combing and aggregating existing information** into new data atoms is a 2 step process:
  - **Knowledge Linking:** identification of all possible correlations between knowledge atoms
  - **Knowledge Generation:** generation of new knowledge atoms

E.g.,



Self-Organizing Approach

- A self-organizing approach is clearly required by:
  - the decentralized nature of pervasive computing systems
  - the overwhelming amount of generated data
- **Knowledge Linking**: **Spider agents** continuously surf W4 Tuple Spaces in order to retrieve tuples that fulfill the specific relationship, those tuples are virtually linked together thus creating a W4 knowledge network.
- **Knowledge Generation**: **Browser agents** generate new knowledge atoms merging related atoms.
- **Using the Knowledge Networks:** Even when new data are not generated, the web of links between atoms can be fruitfully used during querying to access and retrieve contextual information more effectually.

Experimental Results

- We want to evaluate the effectiveness and feasibility of the W4 approach

- **W4 System Prototype** based on LighTS, fully implements the Data Model and the Knowledge Networks algorithms
- Data come from a **simulated distributed environment** (a University Campus) realized with Repast framework with 100 Tuple Spaces and a number of users moving in the campus and performing their day by day activities.

- We compared the results with:
  – Exhaustive Search
  – Hash Based Tuple Space

Università di Modena e Reggio Emilia

Agents and Pervasive Computing Group

Experimental Results: Efficiency

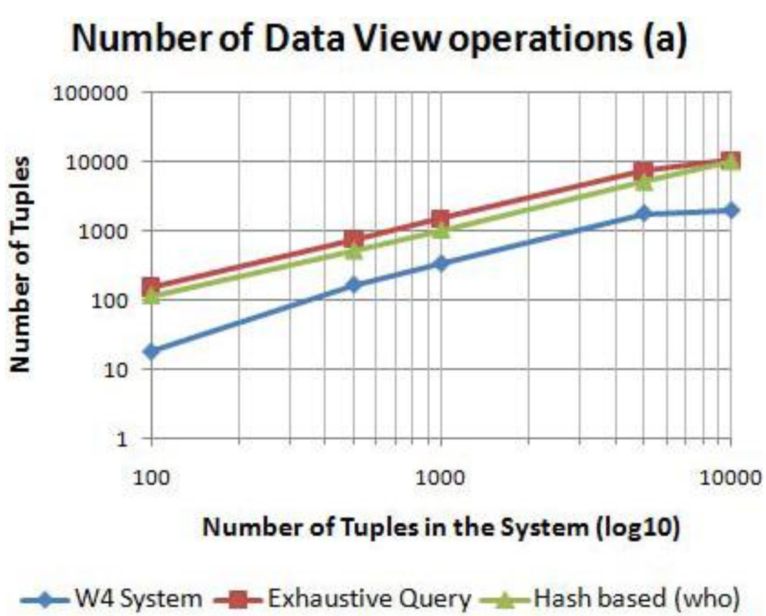
- how the services improve their **data access costs** exploiting the w4 knowledge networks instead of accessing isolated pieces of information
- We submitted to the system the following complex question:

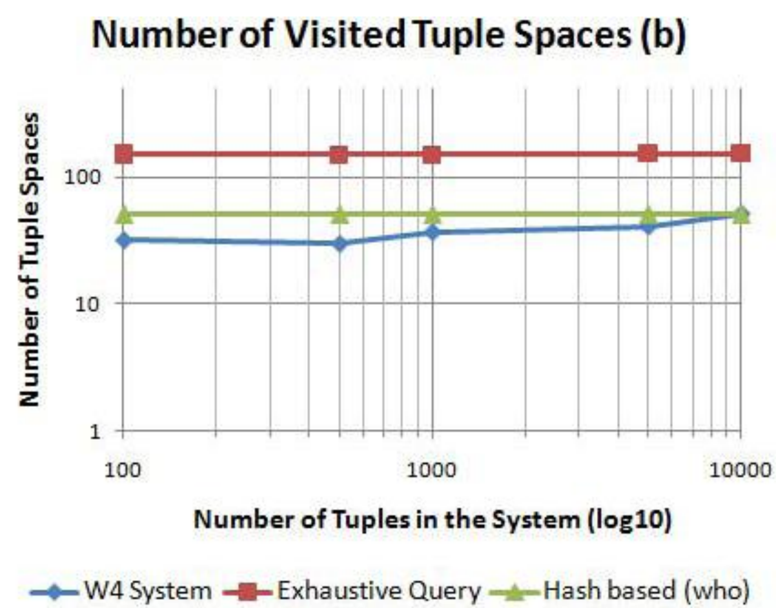
  "Retrieve all the users that were near agent A5 @ time 500"

Who: user:A5
What: *
Where: *
When: 500

then

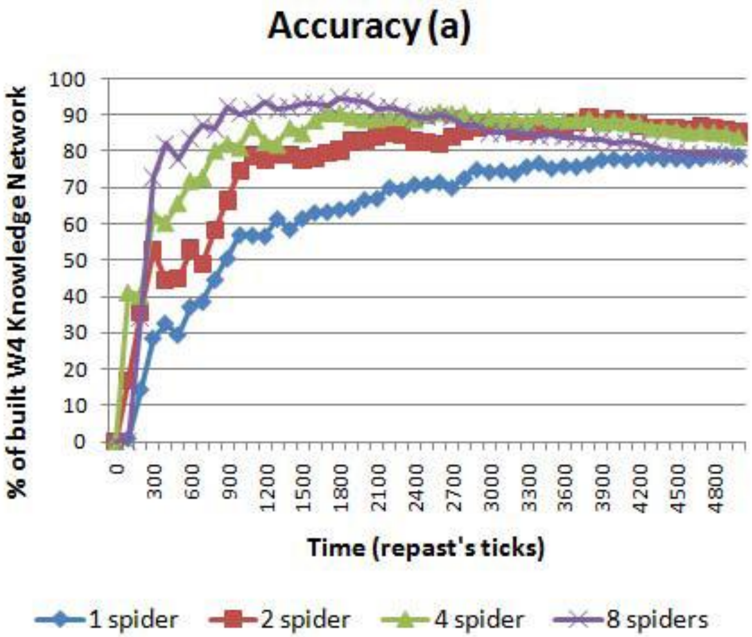
Who: user:*
What: *
Where: <location found>
When: 500

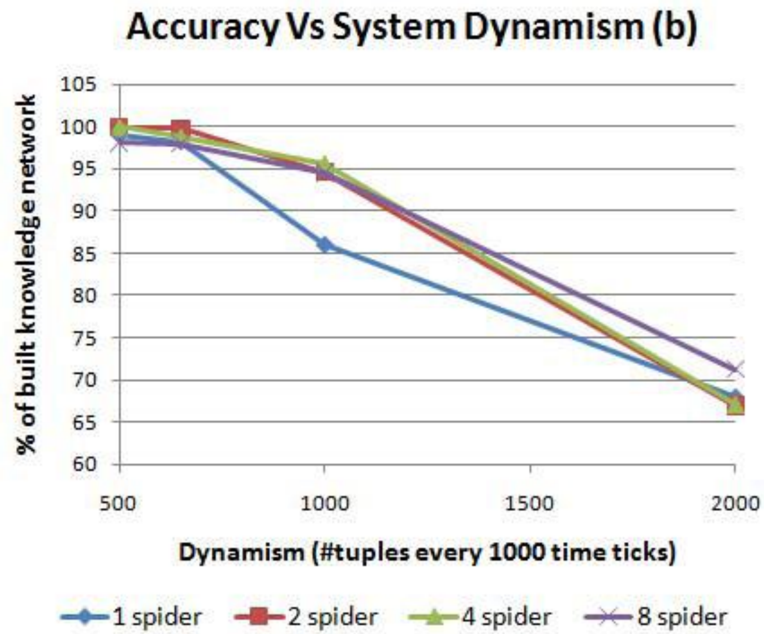




Experimental Results: Effectiveness

- in terms of **accuracy** of provided results when the knowledge networks algorithms are running, i.e. the fraction of the tuples that are relevant to the query that are successfully retrieved.





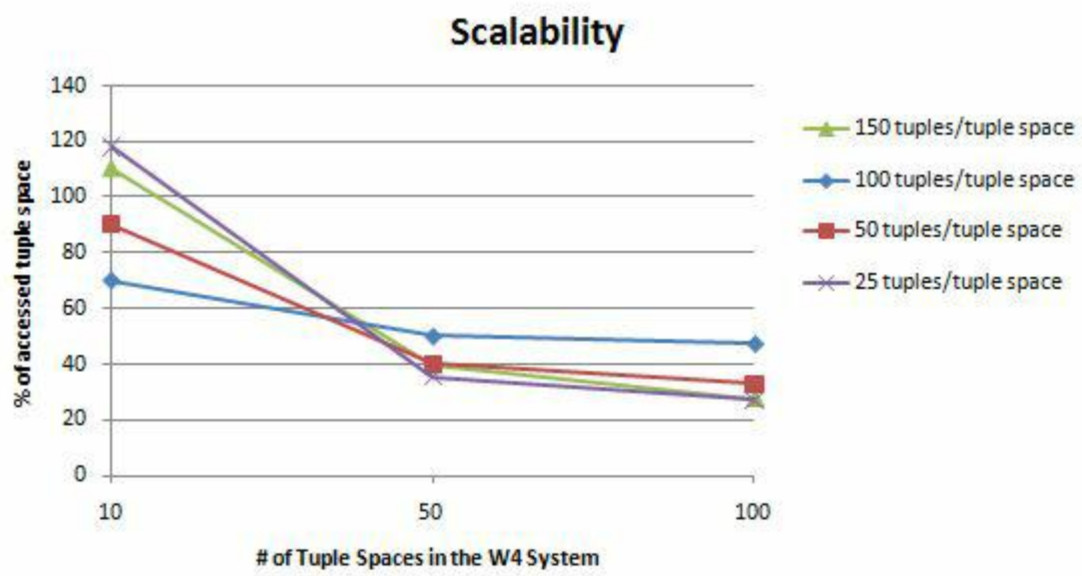
- Of course the indexing works as quicker as more spiders are involved in.
- The W4 system could be improved by taking into account the W4 tuples' injection rate in order to autonomously determine the right number of spiders running.

Università di Modena e Reggio Emilia

Agents and Pervasive Computing Group

Experimental Results: Scalability

- To test the system scalability we performed another set of experiments fixing the number of tuples per tuple space, and varying the number of tuple space in the W4 system and measured the percentage of tuple spaces accessed to solve the query.



- The performances improve when the number of tuple spaces in the system increases.
- Indeed the W4 approach makes sense in a distributed environment rather than in a centralized and static one.

Conclusion & Future Works

- Future pervasive computing scenarios invites considering how to turn the overwhelming amount of pervasive information into usable knowledge.
- The W4 Data Model and the W4 Knowledge Networks try to tackle this challenge.
- There are still open issues:
  - More experiments should be done, in particular to estimate the overhead costs.
  - There is a need for a "garbage collection" to reduce the number of stored tuples.

Università di Modena e Reggio Emilia

Agents and Pervasive Computing Group

Thank for your attention!