inContext: A Pervasive and Collaborative Working Environment for Emerging Team Forms

Hong-Linh Truong, Schahram Dustdar, Dino Baggio, Stephane Corlosquet, Christoph Dorn, Giovanni Giuliani, Robert Gombotz, Yi Hong, Pete Kendal, Christian Melchiorre, Sarit Moretzky, Sebastien Peray, Axel Polleres, Stephan Reiff-Marganiec, Daniel Schall, Simona Stringa, Marcel Tilly, HongQing Yu

truong@infosys.tuwien.ac.at

SAINT'08, 1 Aug 2008,
Turku, Finland

inContext FP6-034718
www.in-context.eu

- ✓ **Coordinated by TU Wien (AT)**

University of Leicester

West Midlands LGA local government association

DERI GALWAY

Microsoft Innovation Centre

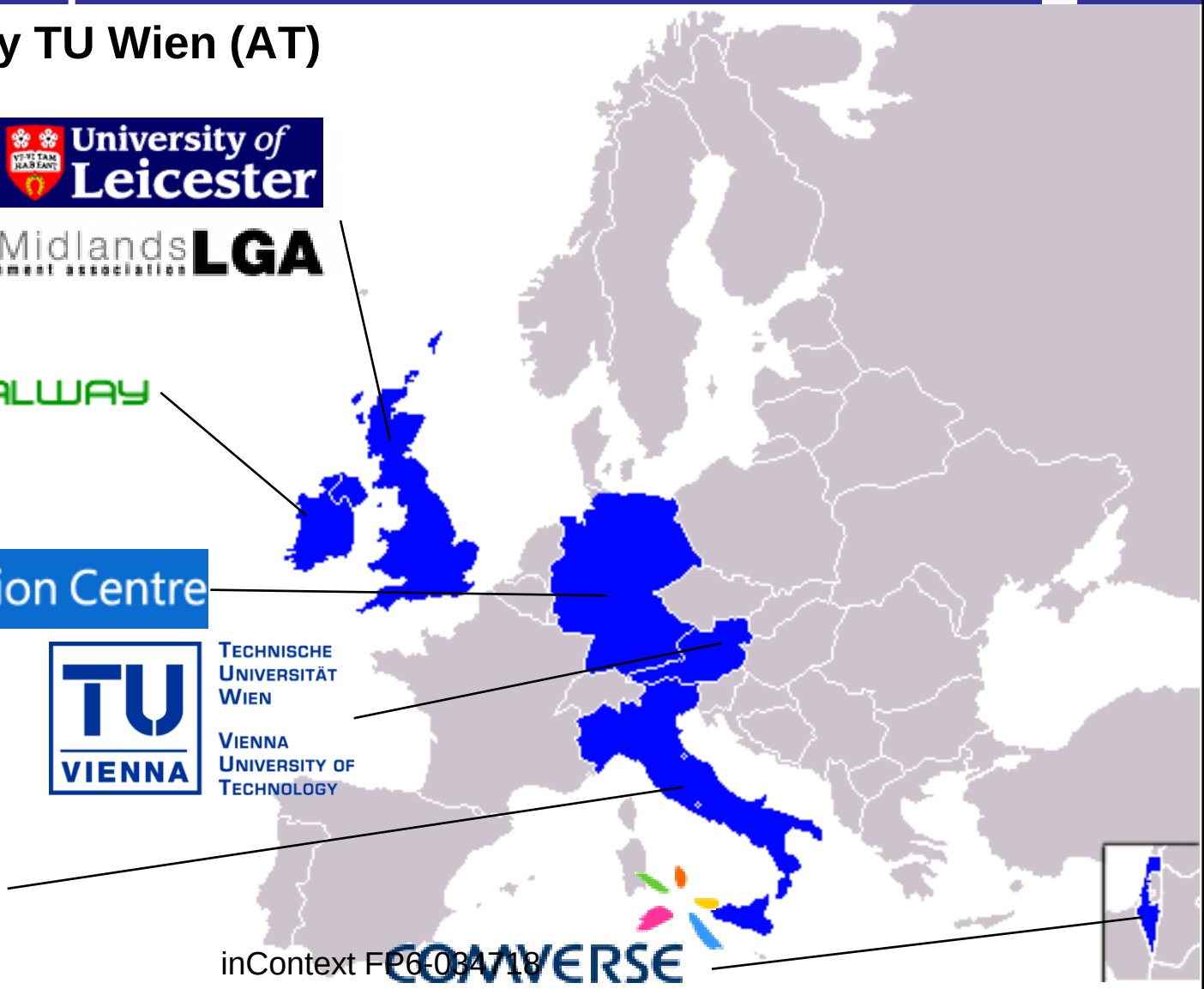
TU VIENNA — Technische Universität Wien — Vienna University of Technology

softeco sismat information technology

hp

Electrolux

COMVERSE



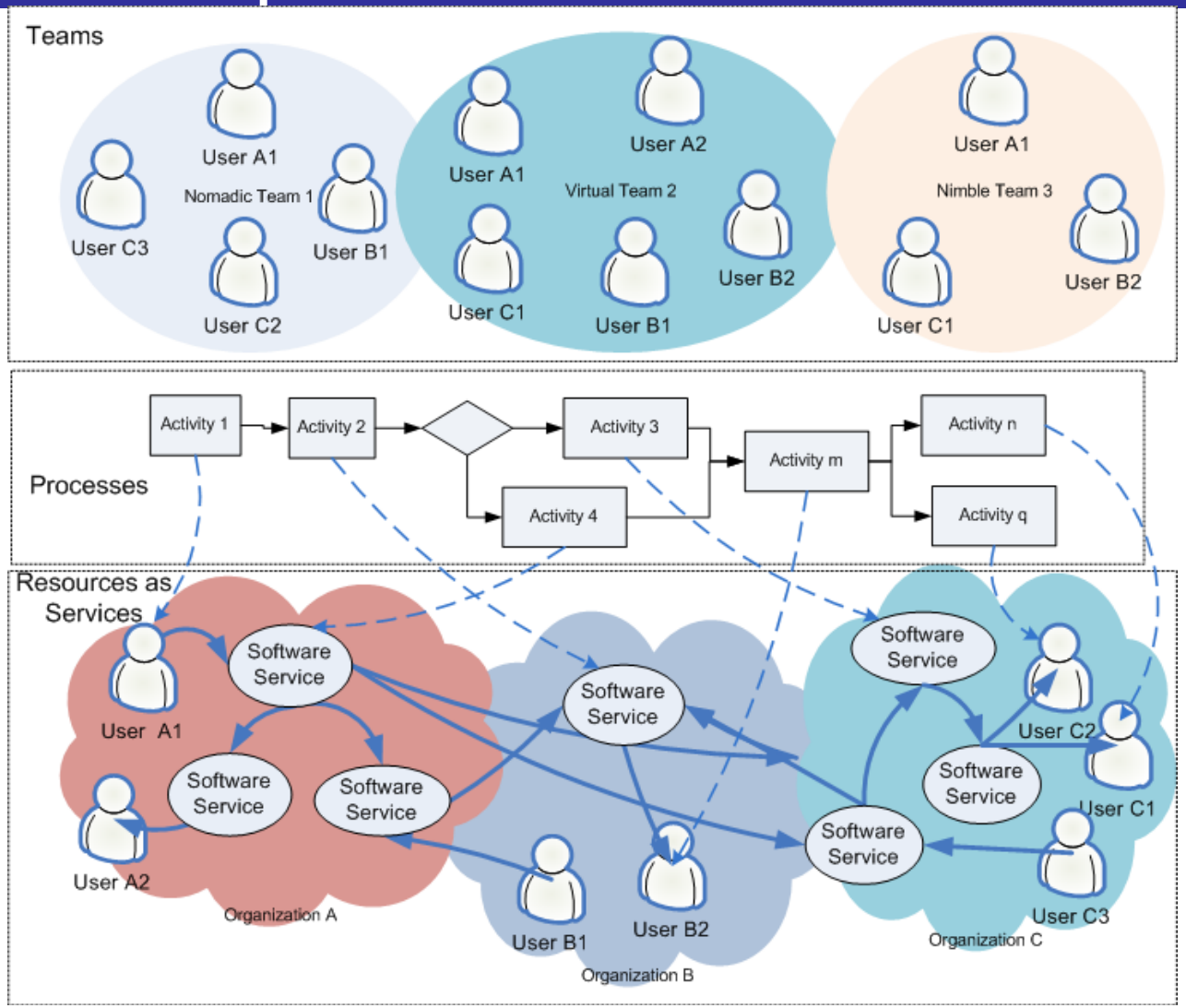
Talk outline

- ✓ Motivation
- ✓ Approach
- ✓ The inContext Environment
- ✓ Context Management
- ✓ Interaction Mining
- ✓ Service Management
- ✓ Tools and Experiments
- ✓ Conclusion and Future Work

Motivation: New emerging team forms

- ✓ The way people collaborate has been changed substantially: Multi-objective and nomadic working style and ad-hoc collaborations
  - Working different objectives and projects at the same time
  - Moving from places to places during the collaboration
  - Using a variety of devices and infrastructures
- ✓ Many new emerging team forms
  - Nimble: short-lived collaboration to solve emerging problems
  - Virtual: spanning different goegraphical place and having diverse professionals
  - Nomadic: collaboration with mobility capabilities

Motivation: teams, activities and services



Motivation : the problem

✓ Traditional collaborative working environments

- Collaboration tools and services are not integrated into a unified system
- Users have to manually select individual tools/services
- Context and interaction have not been well utilized
- See our report for European Space Agency at https://www.vitalab.tuwien.ac.at/autocompwiki/index.php/Current_and_Future_Technologies_for_Collaborative_Working_Environments_study

→ Collaboration tools/services are hardly reusable

→ Services cannot be adapted according to team context and interaction

→ Existing CWEs are not able to support emerging teams in highly dynamic environments

Motivation: questions

- ✓ How to integrate diverse collaboration tools and services built with different technologies and provided by different organization?
  - To avoid monolithic/proprietary applications and to support the composition
- ✓ How collaboration services are adapted to the collaboration context of emerging team forms ?
- ✓ How to reduce human intervention in CWEs ?

→ The inContext aims at providing solutions for these questions by providing *context and interaction based collaboration techniques*

- ✓ How can we integrate different (free, commercial) collaboration services belonging to different organization?
  - Utilize service computing principle to loosely couple and aggregate diverse types of collaboration services
- ✓ How do we know the context of teams, their activities and operating environments?
  - Explicitly model context associated with emerging teams
  - Infer and enrich existing context to provide high-level information
- ✓ How do we monitor and quantify metrics and patterns associated with interactions inherent in collaborations
  - Employ interaction mining techniques to understand metrics and patterns associated with interactions

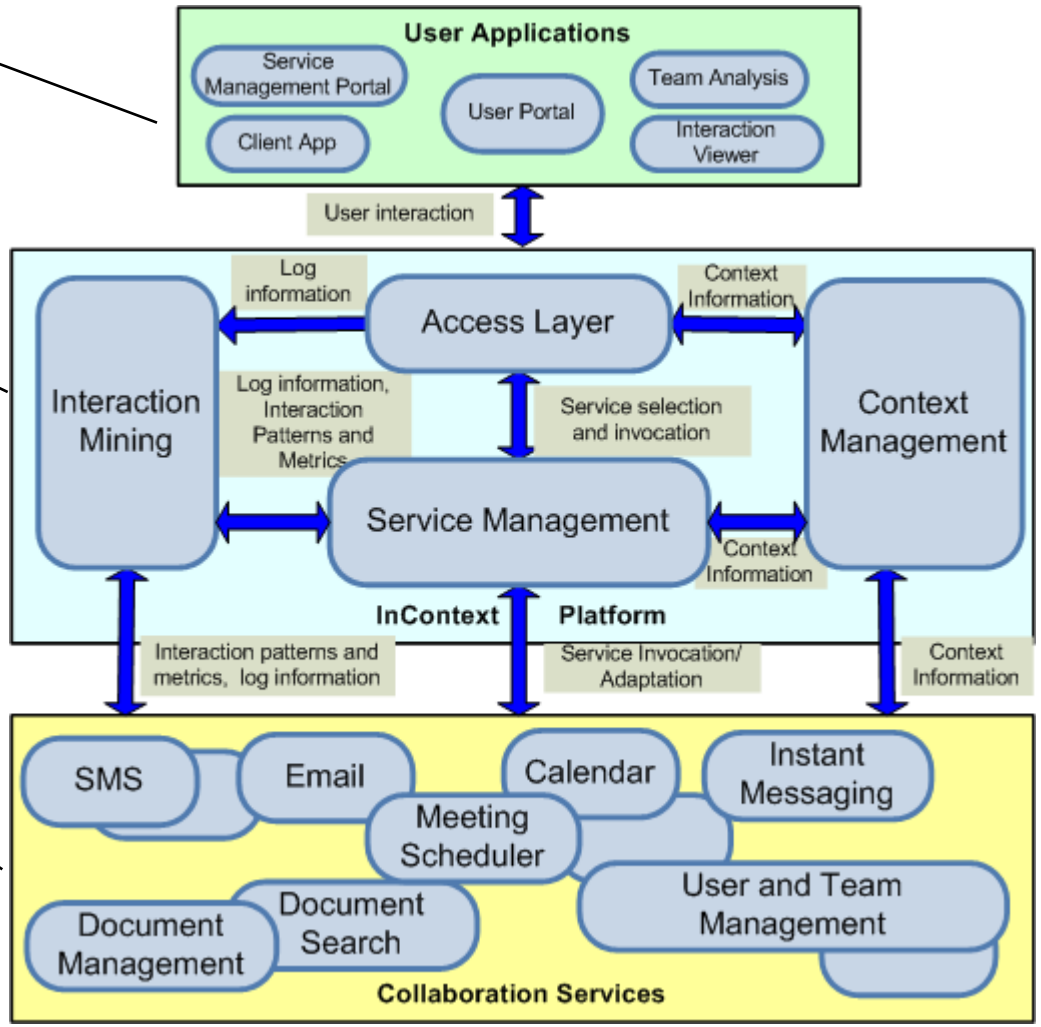
→ This talk gives you an overview of our approach

The inContext Environment

Providing different types of end user applications for different platforms and devices

Providing context information, metrics and patterns, perform service selection and adaptation

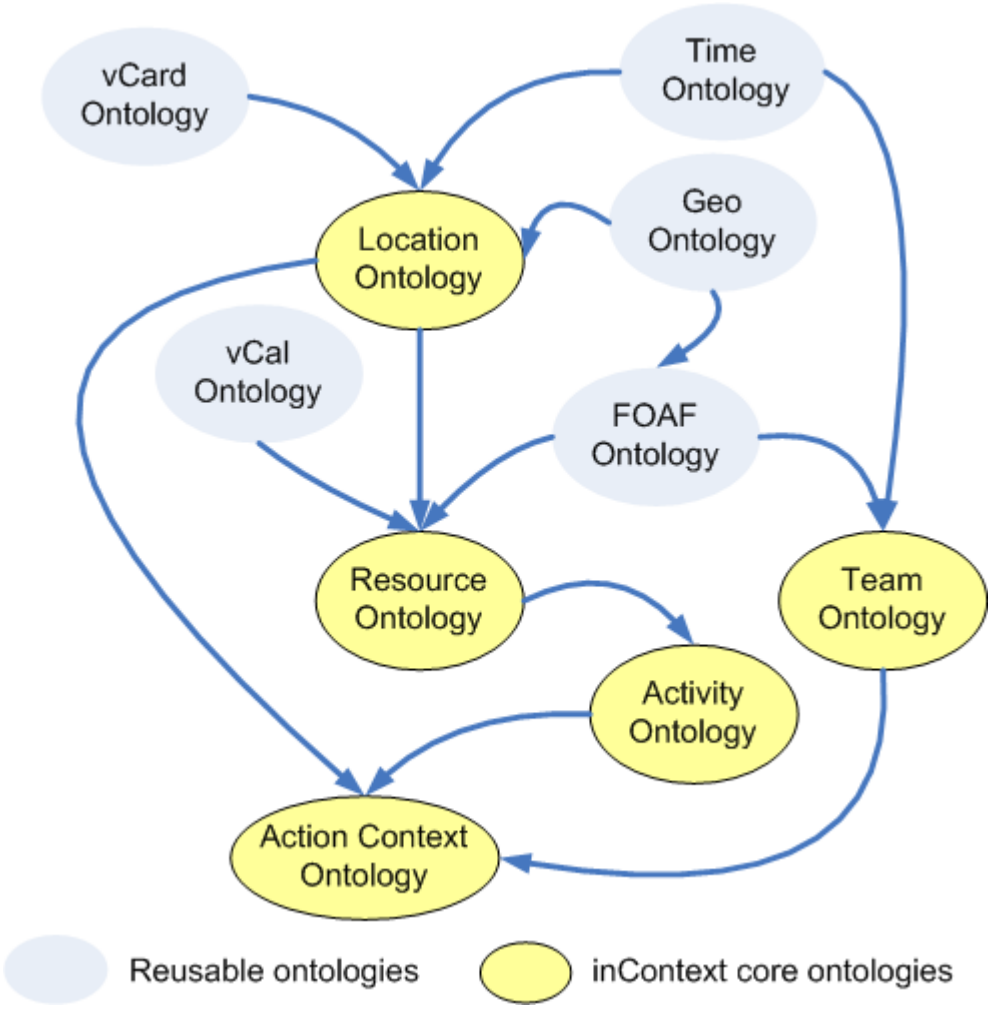
Providing basic operations normally required in collaborations



- ✓ A reference implementation of Pervasive Collaboration Service Architecture (PCSA)
- ✓ PCSA addresses
  - Interfaces between diverse types of common collaboration services
  - Core services for supporting context- and interaction-based collaboration and their interfaces
  - Deployment strategies for different team forms and infrastructures

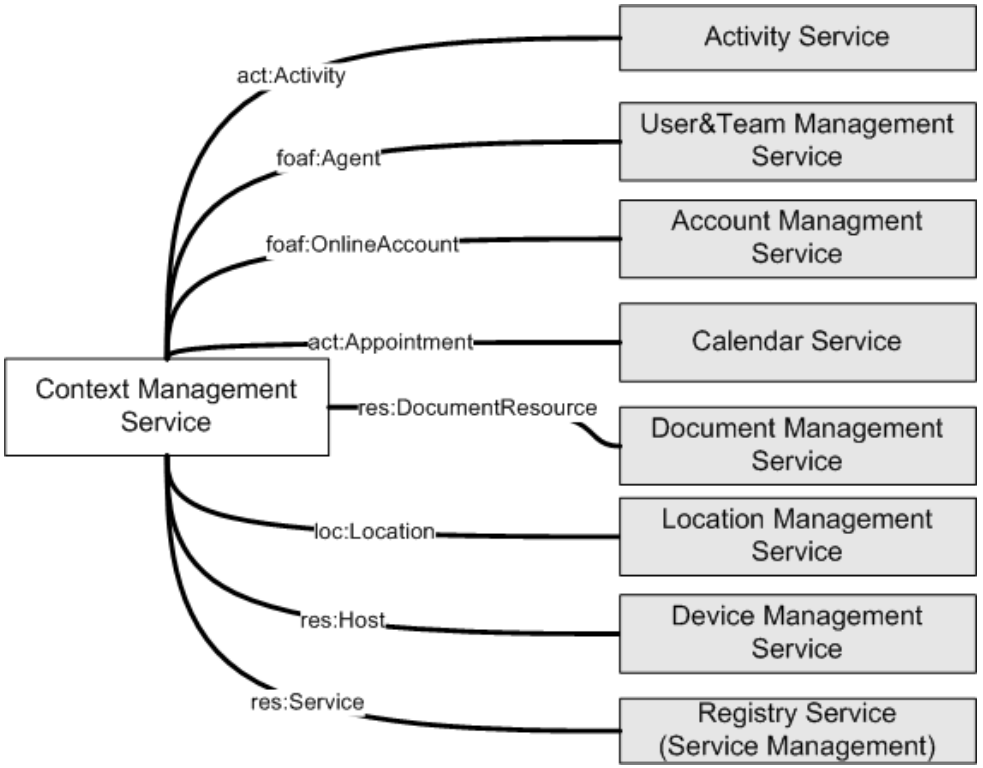
Context Management: Context model

- ✓ Context associated with team collaboration is much more complex than HCI or location-based services
  - Human, services, teams, activities, and interaction between human and services
- ✓ Existing context models are not enough
  - Reuse existing concepts and develop new ones
- ✓ inContext relies on RDF+OWL



Context Management: distributed storage

- ✓ Context information collected from different sources
- ✓ Centralized context store is not suitable
- ✓ Context information is stored in different services
  - Linked through a core model

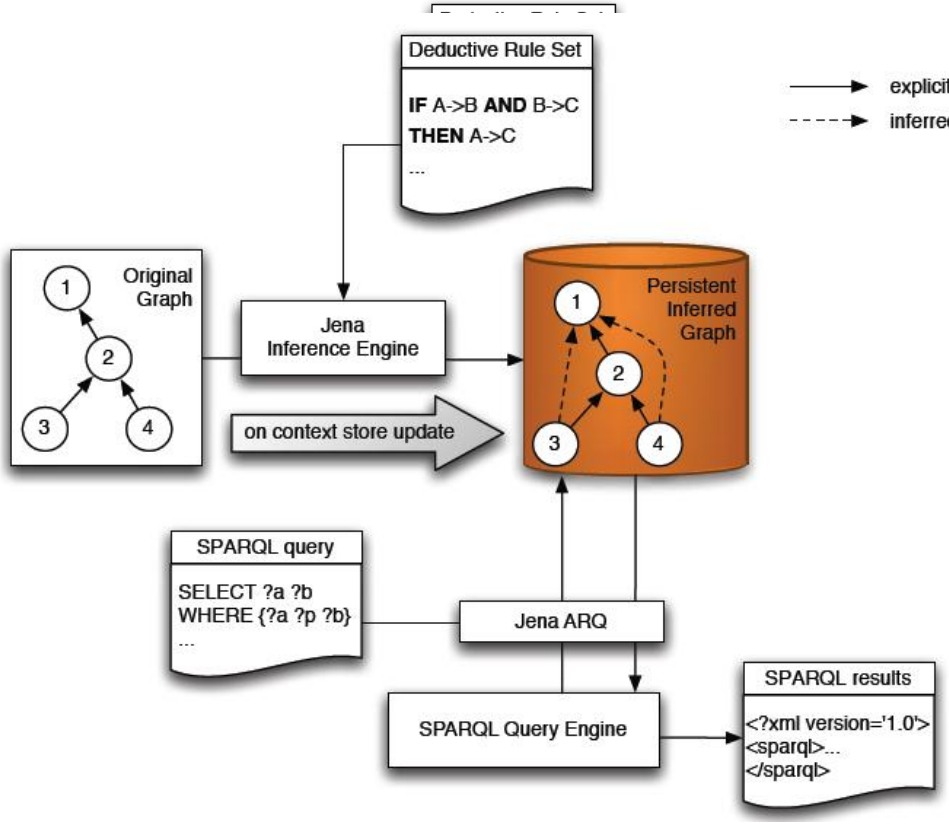


- ✓ Context information can be inferred based on rules
  - Provide insightful information about context associated with people, teams, services and activities
  - Based on SPARQL++
- ✓ Example: using reasoning techniques to find all civil engineers available at a particular site.

```
PREFIX team:<http://www.in-context.eu/team.owl#>
SELECT ?engineer
WHERE{
      ?engineer :hasProfile ?profile.
     ?profile :hasSkill ?skill.
     ?skill :name ?sname.
     ?engineer :locatedAt :''Genoa sea port''
FILTER regex(?sname,"civil engineer","i")
}
```

Reasoning Approach

- In-Memory Inferencing:
  Inferred model is created in the memory every time, when query finished, it will be dropped.
  - Flexible, ability to specific reasoning rules for different queries. Lack of efficiency, need to load entire model into memory
- Persistent Inferencing: A set of static rules are applied directly on the persistent graph (Database) at all time.
  - Query is more efficient. But reasoning rule set are immutable.



In-Memory Inferencing

Persistent Inferencing

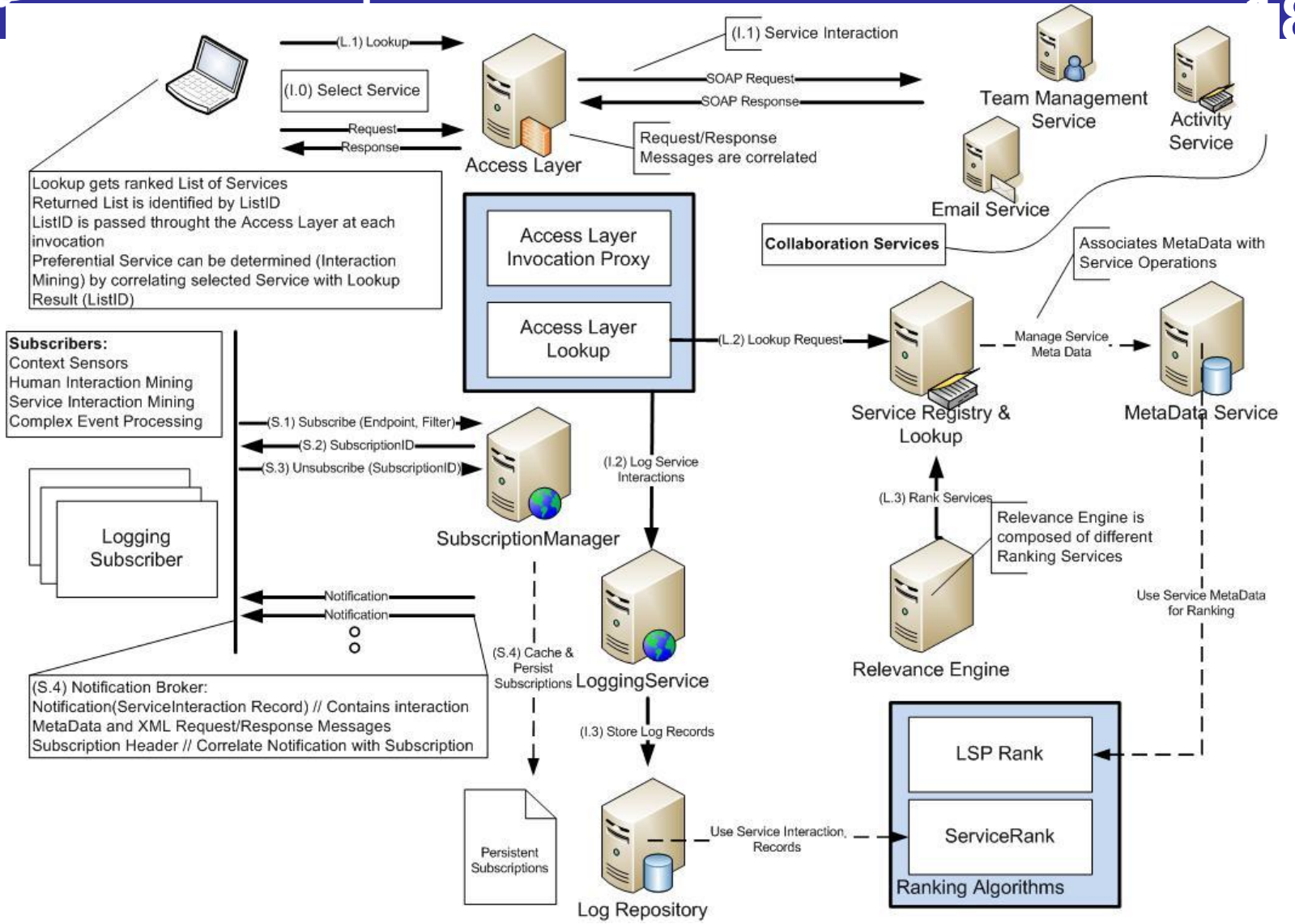
- ✓ Used to understand characteristics of team members, types of communication, performance of services
- ✓ Provide quantitative information associated with interactions for enriching context and selecting services
- ✓ Three types of interactions
  - Service-to-service
  - Human-to-service
  - Human-to-human
- ✓ Three levels of information
  - Individual (human or service), group (a team or a set of services), and the collaboration (all teams and services)

Interaction Mining: Examples of metrics and patterns

Interaction/level	Individual	Group	Collaboration
Service-to-service	Number of invocations, number of unavailability, number of failures, number of consumers	Usage distribution, usage mode (isolated or composite) patterns, service interactions network	Usage distribution, usage mode (isolated or composite) patterns
Human-to-service	Number of service invocations, usage mode (isolated or composite) patterns	Usage distribution, constant/-durable/limited duration usage patterns	Usage distribution, constant/-durable/limited duration usage patterns
Human-to-human	Number of callers/callees, number of interactions, number of assigned activities	Team size, total interactions, average number of callers/callees, interaction networks	Broker, proxy, master/slave, coauthoring patterns, interaction networks

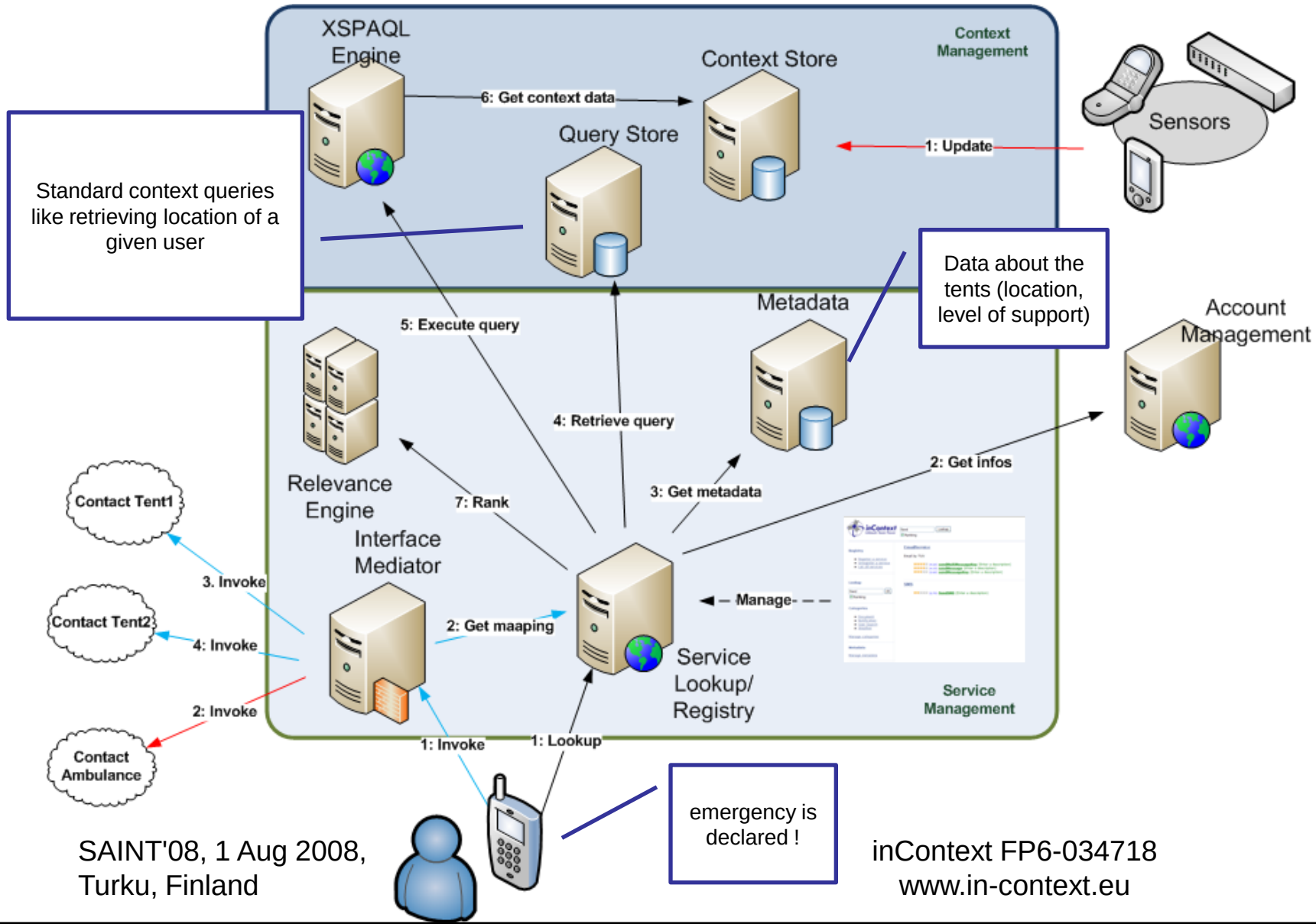
- ✓ Diverse collaboration services
  - Complement or compete
  - Are utilized differently, depending on the context
  - How to select the right service upon the context?
- ✓ Traditional service selection approach
  - Based on service-meta information, and possibly historical data of service usage
  - Not enough for emerging team work due to the lack of context consideration
- ✓ inContext approach: service selection based on four types of information
  - Context information, interaction information, and service meta-information

Service Management and Logging and Interaction Mining Infrastructure



- ✓ Service operations are associated with category
- ✓ Service-meta information includes a set of criteria of metrics and weighted factors
  - Cost, reliability, availability
  - Criteria can include SPARQL queries
- ✓ Multiple-steps in selecting a service
  - Using keyword matching to select the right service category
  - Ranking services based on meta-information, interaction information, and context information.
    - Also support a modified LSP algorithm and a service rank algorithm
  - Selecting the best service
- ✓ Service adaptation at runtime

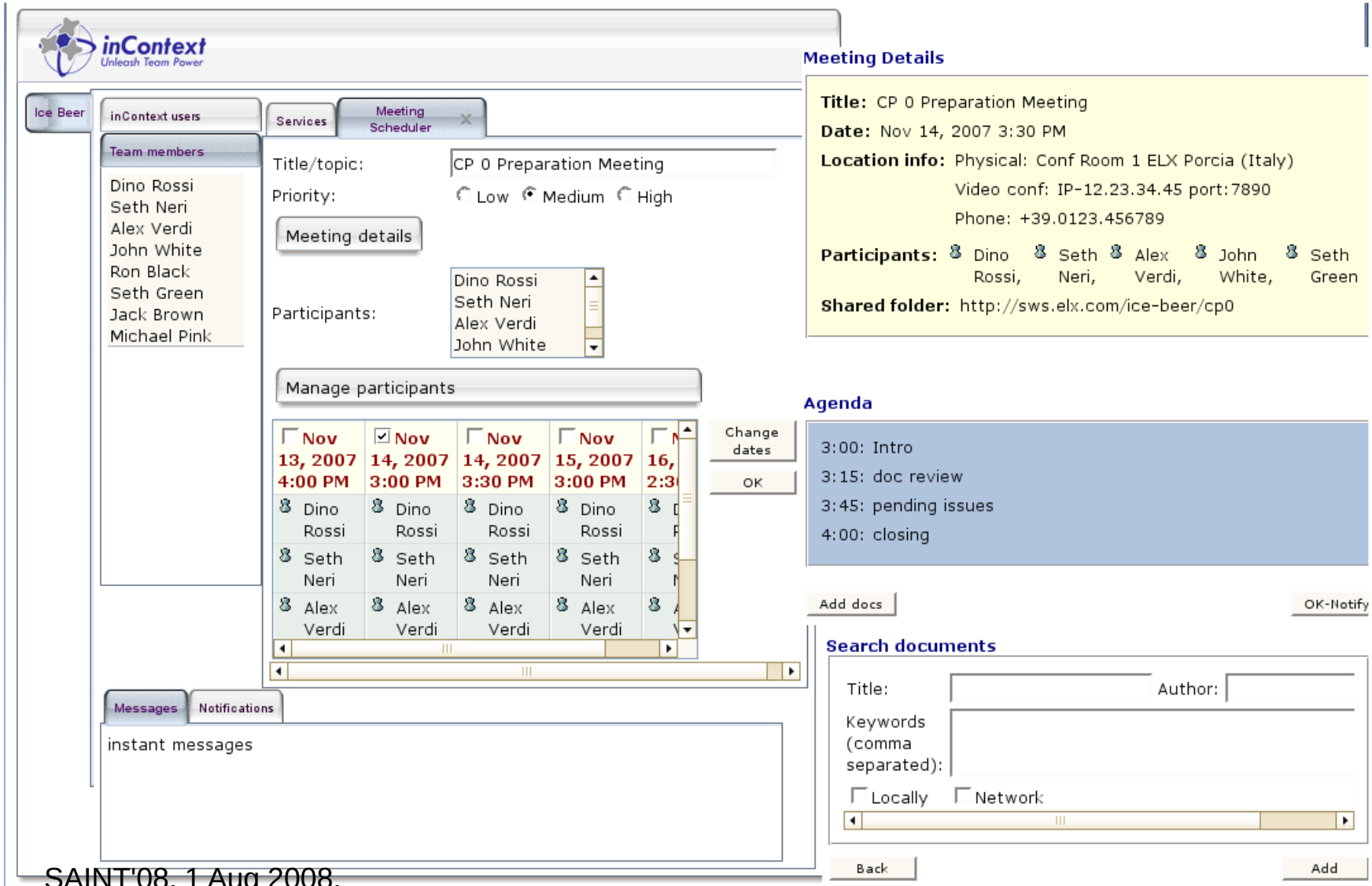
Example of Service Selections



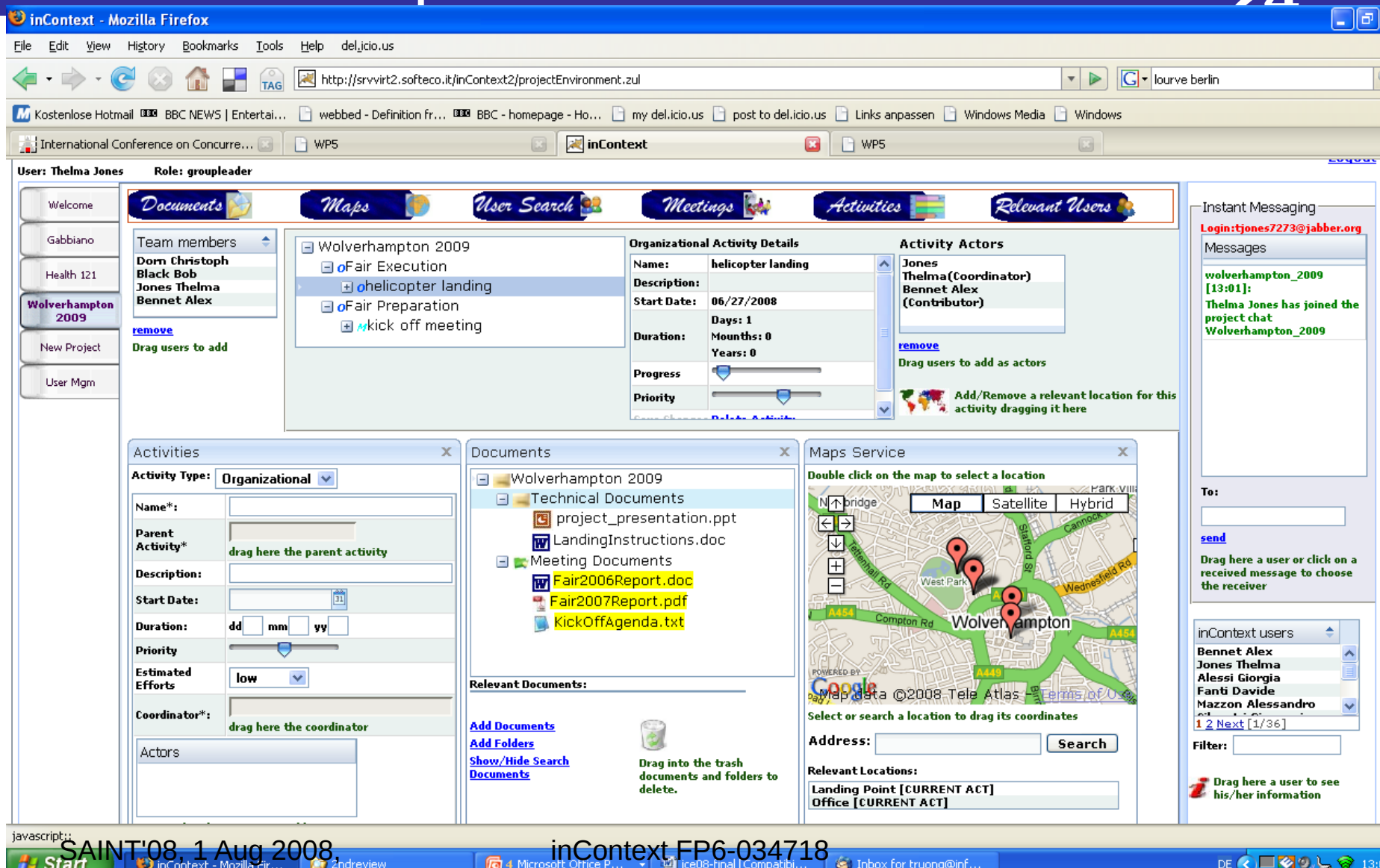
- ✓ Services are implemented in Java/AXIS/Tomcat and C#/.NET
- ✓ AJAX-based collaboration tools
  - Using ZK framework
- ✓ Collaboration services
  - Calendar, Email, Instant Messaging, Document Management, Document Search, Meeting Scheduler, SMS, Activity Management, etc.
- ✓ Some support for mobile devices
- ✓ Services deployed in Aachen, Genoa, Leicester, Milan and Vienna

Examples of Collaboration Tools

- ✓ Many collaboration tools can be built
  - By utilizing common collaboration services
  - By utilizing context-aware supporting services
- ✓ Electrolux case study: Meeting Scheduling collaboration tool: *support all relevant steps in preparing a meeting*
- ✓ Event Management Tool – Wolverhampton Fair case study from WMLGA: *support the organization, communication, cooperation and coordination of activities*
- ✓ Both tools utilize common collaboration services and composite services based on common ones


inContext
Unleash Team Power
Ice Beer
inContext users
Services
Meeting Scheduler
Team members
Dino Rossi
Seth Neri
Alex Verdi
John White
Ron Black
Seth Green
Jack Brown
Michael Pink
Title/topic:
CP 0 Preparation Meeting
Priority:
Low
Medium
High
Meeting details
Participants:
Dino Rossi
Seth Neri
Alex Verdi
John White
Manage participants
Nov 13, 2007 4:00 PM
Nov 14, 2007 3:00 PM
Nov 14, 2007 3:30 PM
Nov 15, 2007 3:00 PM
Dino Rossi
Seth Neri
Alex Verdi
Change dates
OK
Messages
Notifications
instant messages
Meeting Details
Title: CP 0 Preparation Meeting
Date: Nov 14, 2007 3:30 PM
Location info: Physical: Conf Room 1 ELX Porcia (Italy)
Video conf: IP-12.23.34.45 port:7890
Phone: +39.0123.456789
Participants: Dino Rossi, Seth Neri, Alex Verdi, John White, Seth Green
Shared folder: http://sws.elx.com/ice-beer/cp0
Agenda
3:00: Intro
3:15: doc review
3:45: pending issues
4:00: closing
Add docs
OK-Notify
Search documents
Title:
Author:
Keywords (comma separated):
Locally
Network
Back
Add

Event Management Tool



Some Videos

Meeting scheduling

- ✓ Meeting scheduling problem
  - Frequently required for team collaboration
- ✓ It is complex due to emerging team forms
  - Many constraints have to be implemented
- ✓ Three main steps in planning a meeting
  - Selecting suitable time and participant
  - Preparing document
  - Sending notification/changes
- ✓ Three steps can be fully automated in inContext by utilizing context reasoning, rules, and service selection

Experiment: example of rules for a meeting

Meeting priority and attendance rules

IF meeting priority = High THEN

….

ELSE IF meeting priority = Medium THEN

Attendance type = Any (Physical | Phone | Video)

Organizer attendance = Physical

Travel for meeting = False

Proxy participation = At the same level or one level below

Attendance Quorum = At least 1 for each L2 type

ELSE IF meeting priority = Low THEN

…

ENDIF

Notification rules

Always send MAIL with Full Details

IF present on Instant Messaging (IM) THEN

send summary as IM message

ELSE

send summary using SMS

ENDIF

E.g., Using reasoning techniques to automatically find possible time slots for the meeting

```
PREFIX iCal: <http://www.w3.org/2002/12/cal/ical#>
SELECT ?T
WHERE {<m1> :possibleTimeSlot ?T ; :priority "low".
      ?T time:hasBeginning ?TB; time:hasend ?TE.
 FILTER( COUNT{?P : { <m1> :invited ?P }} >=
     2 * COUNT{?P :
     { <m1> :invited ?P .
     ?P :hasCalendar ?C .
GRAPH ?C { ?E a iCal:Vevent;
     ical:dtstart ?B
     ical:dtstart ?E. }
FILTER( ( ?B >= ?TB && ?B <= ?TE )
  || ( ?E >= ?TB && ?E <= ?TE ) )
}
```

E.g., automatically find relevant documents

```
PREFIX res: <http://www.in-context.eu/resource.owl#>
PREFIX act: <http://www.in-context.eu/activity.owl#>
PREFIX
rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?resoure ?meeting
{
?meeting rdf:type act:Activity.
?meeting :shortname "review meeting"^^xsd:string.
?meeting :usesResources ?resource.
?resource rdf:type res:DocumentRepository.
}
```

E.g, Check online status of a participant named Rossi

```
PREFIX ctx: <http://www.in-context.eu/context.owl#>
SELECT ?x ?y
WHERE{
?a ctx:connectedBy ?x .
?x ctx:hasOnlineStatus ?y .
?y ctx:status ?z .
}
```

E.g., Send notification

- ✓ It turns out that we have to send SMS to Rossi
- ✓ Service Management ranks existing SMS providers
- ✓ Service Management sends the notification to Rossi through the best ranked one

- ✓ inContext: a novel pervasive and collaborative working environment
  - Support emerging team forms
  - Provide techniques for integrating existing collaboration services and for context- and interaction-based collaborations
  - Proof the concept with real world applications
- ✓ Multidisciplinary research: Web services engineering + ontology/semantics + collaborative computing
- ✓ Future work
  - Further development of the Pervasive Collaboration Services Architecture
  - Collaboration-aware adaptation and composition
  - Distributed users/teams managements, context policy and privacy issues

Thank for your attention!

Contact:

Hong-Linh Truong

Distributed Systems Group
Vienna University of Technology
truong@infosys.tuwien.ac.at
https://www.vitalab.tuwien.ac.at/autocompwiki/