Pervasive Service Ecosystems

Mirko Viroli
mirko.viroli@unibo.it

Alma Mater Studiorum—Università di Bologna, Cesena

Academic Year 2011/2012

"Self-Aware Pervasive Service Ecosystems"

SAPERE European Project

- European call for funding projects: FP7-ICT-2009.8.5
- Call name: Self-awareness in Autonomic Systems
- Period: 1/10/2010 - 30/9/2013
- Consortium: (i) Università di Modena e Reggio Emilia, Italy
  (ii) Università di Bologna, Italy
  (iii) University of Geneve, Switzerland
  (iv) University of St Andrews, Dublin, UK
  (v) Johannes Kepler Universitaet Linz, Austria

Impact on the Cesena site

- A "fulcrum" of challenging EU-wide research activities,
- An opportunity for students (projects, thesis, post-laurea)

Outline/Goals of this seminar

- Deepen the "Future Pervasive Computing" scenario
- Describe the SAPERE Project
- Show what is the structure of an European Project
- Present main research ideas/results/challenges
- Propose collaborations

Outline

Challenges of Future (and emerging) Pervasive Computing

1 — Increasing introduction of pervasive devices

Thanks to Alois Ferscha and JKU unit in SAPERE for this picture — "(Thanks to JKU for the picture)" for short in the following

Challenges of Future (and emerging) Pervasive Computing

1 — Increasing introduction of pervasive devices

- The world will be more and more populated of computing devices
  - GPS, tablets, smart phones, tags, cameras, displays, sensors, actuators
  - Note that number of such devices follows an exponential law
- Internet, Telecom, TV networks will become integrated
  - Think at Skype, Digital TV
- The network will become a true virtual counterpart of the world
  - concerning events, sociality, business, logistics, physical structure
  - it will be increasingly dense up to be understood as a continuum
  - with obvious ethic and social consequences..

Challenges of Future (and emerging) Pervasive Computing

2 — Prosumption of large masses of data and services

- Devices will produce large masses of data
- Users will themselves inject their data in the system [11]
  - Think at Facebook, Youtube
- We will expect that any environment properly react to our preferences/situation
- Not just data, we are already facing universes of applications
  - E.g. Android Market may become a standard deployment tool
  - There is an increasing "long tail" in the market [3]
- Distinction between data and (atomic) services will blur

Challenges of Future (and emerging) Pervasive Computing

3 — Software will grow increasingly and be an "eternal beta"

- The standard notion of software is (slowly) shifting
  - No longer a big monolithic engineering artifact
  - ..but rather a mash-up of services and data [6]
- similarly, the development cycle will become much different
  - No longer analysis-design-development-deployment
  - ..but rather it starts with initial services, and through years..
  - ..new simple services are injected
  - ..existing services are updated by small changes
  - ..even the availability of data and users will make the difference
- there is a feeling that a system is never in final state, but always in a "beta" state, that will improve next months
  - Open-source projects (even OS), social networks, ..

Challenges of Future (and emerging) Pervasive Computing

4 — Opennes, self-*, contex-awareness will be mandatory

- Opennes: we won't know which services, data, users, devices will be available soon, the infrastructure should work independently of this
- Self-* features should naturally emerge
  - Self-adaptiveness: tuning behaviour to ongoing changes
  - Self-organisation: find a better/new organisation (spatial/temporal)
  - Self-optimisation: be able to garbage services/data
  - Self-awareness: intrinsic identification of situations
- Context-awareness: data and services will be relative to the position/location in which they reside
  - centralisation of data and software will be abandoned
  - (or, will clouds take over?)

The challenge of Pervasive Computing

Oppurtunities for industry

- We are far from supporting what we have in mind
- Only specific solutions to specific problems so far (even by academia)

A lot of work is going on

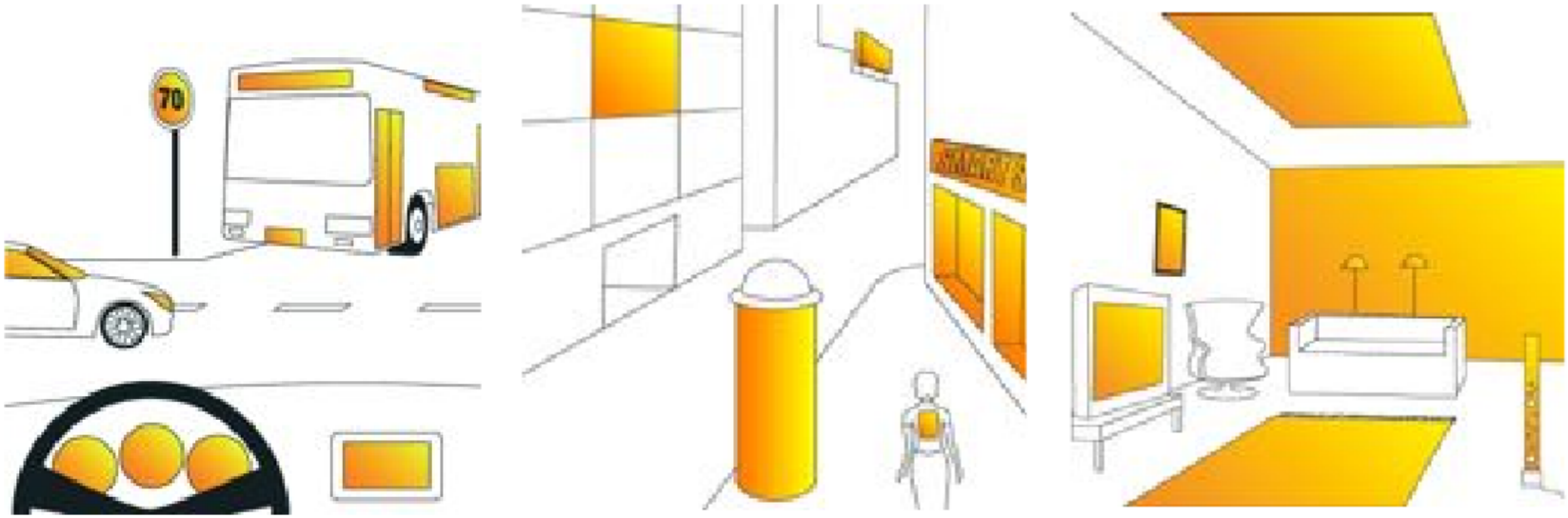
- Research contexts: SOA, P2P, Grid, Cloud, Self-org, Coord
- Argue there are *two next big things* in Information Technology
  - Pervasive Computing
  - Bio-ICT convergence (e.g. nature-inspired computing)

Outline

The scenario of display ecosystems

(Thanks to JKU for the picture)

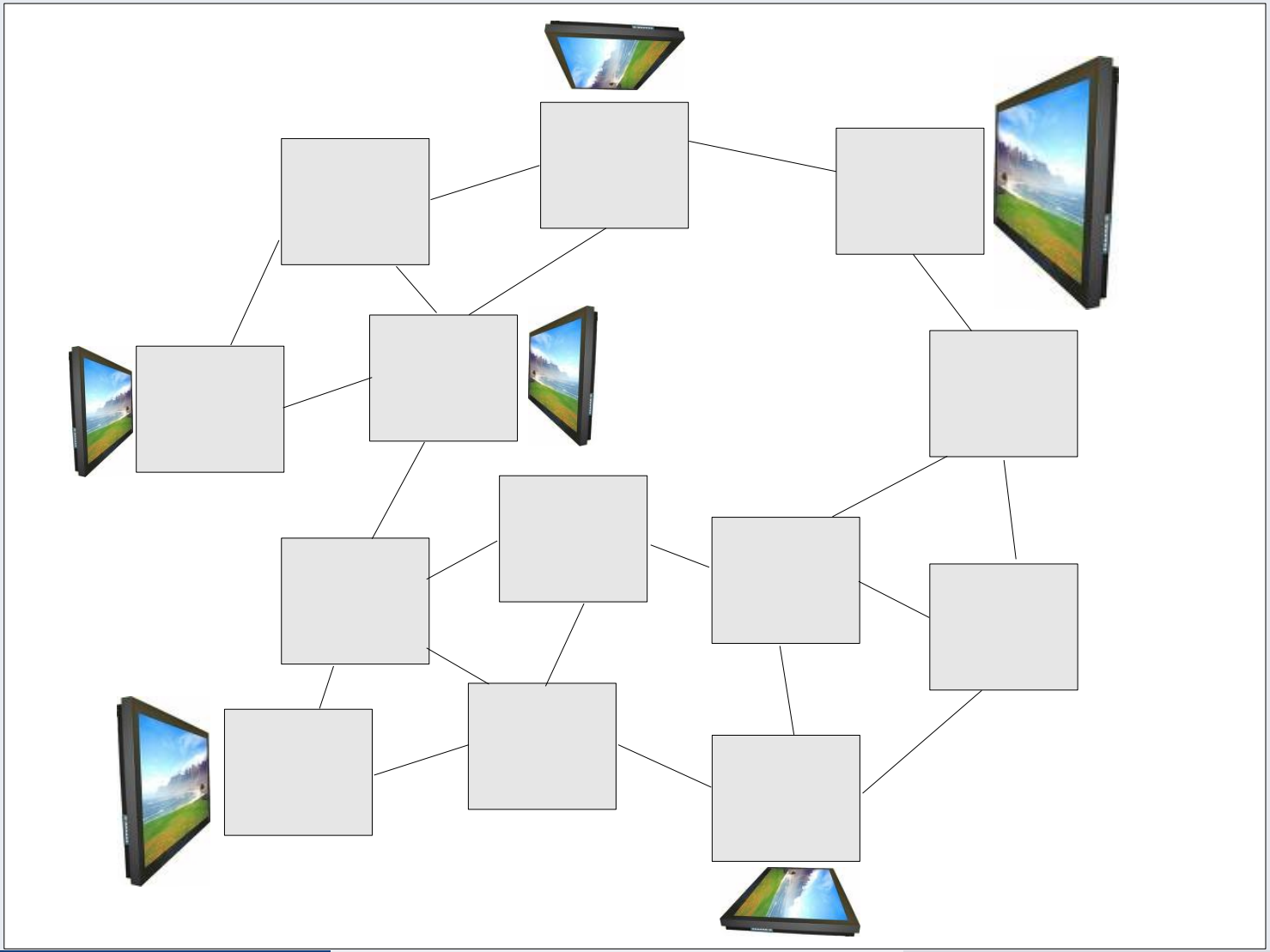
Displays are (and will increasingly be) pervasive



(Thanks to JKU for the picture)

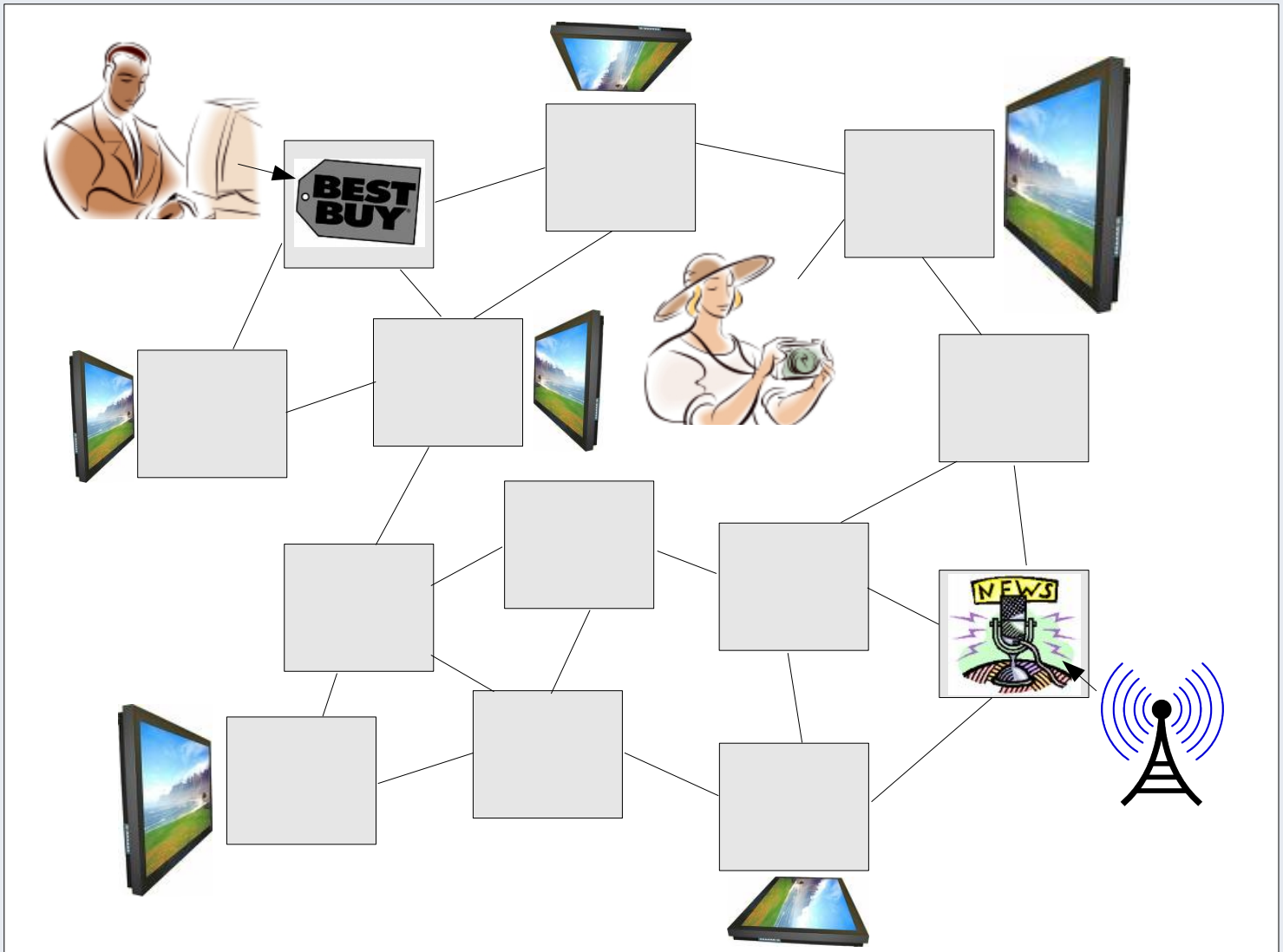
A scenario of pervasive displays

A pervasive network with displays spread around



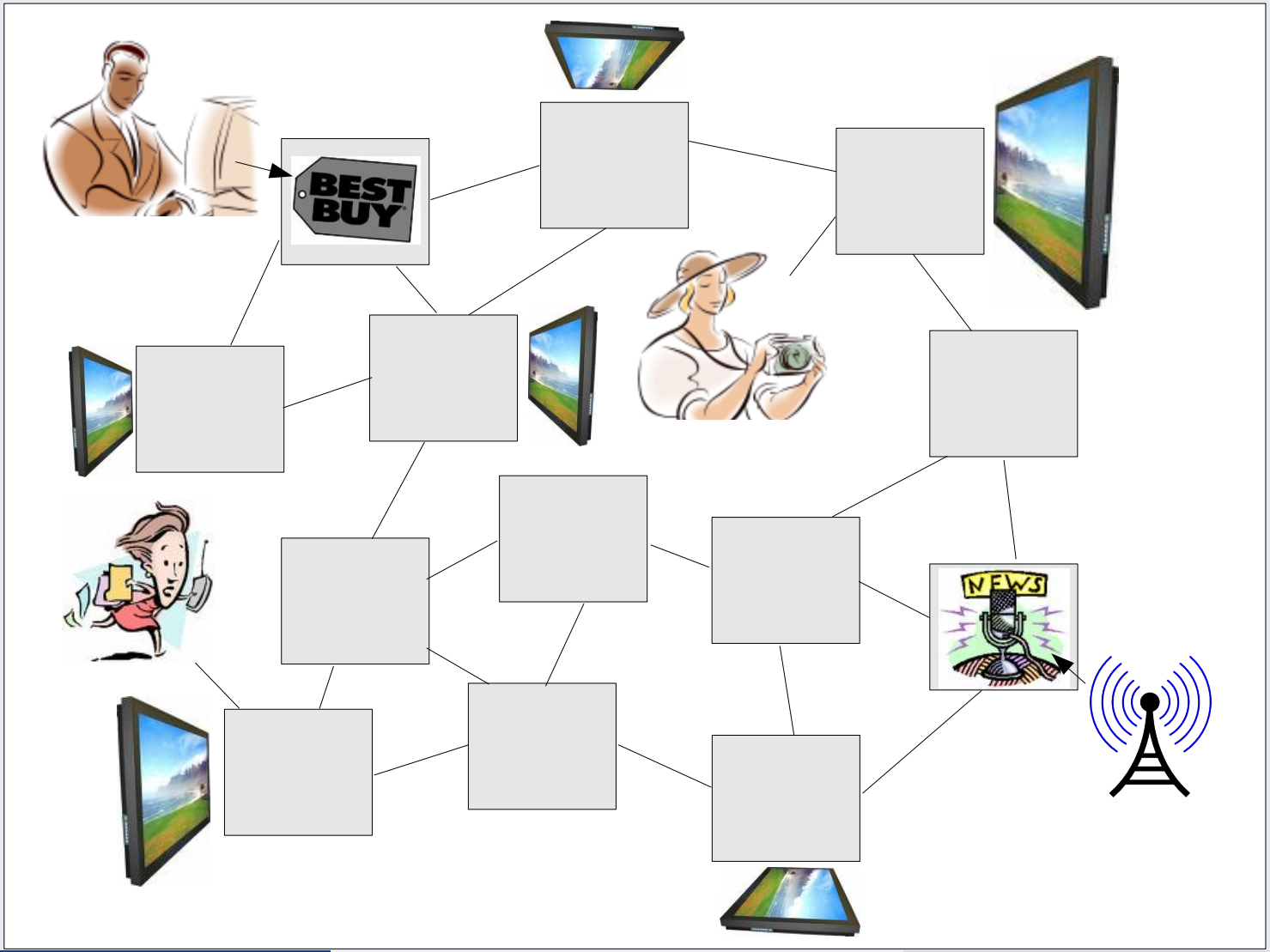
A scenario of pervasive displays

Visualisation services (news, ads, social data) get injected



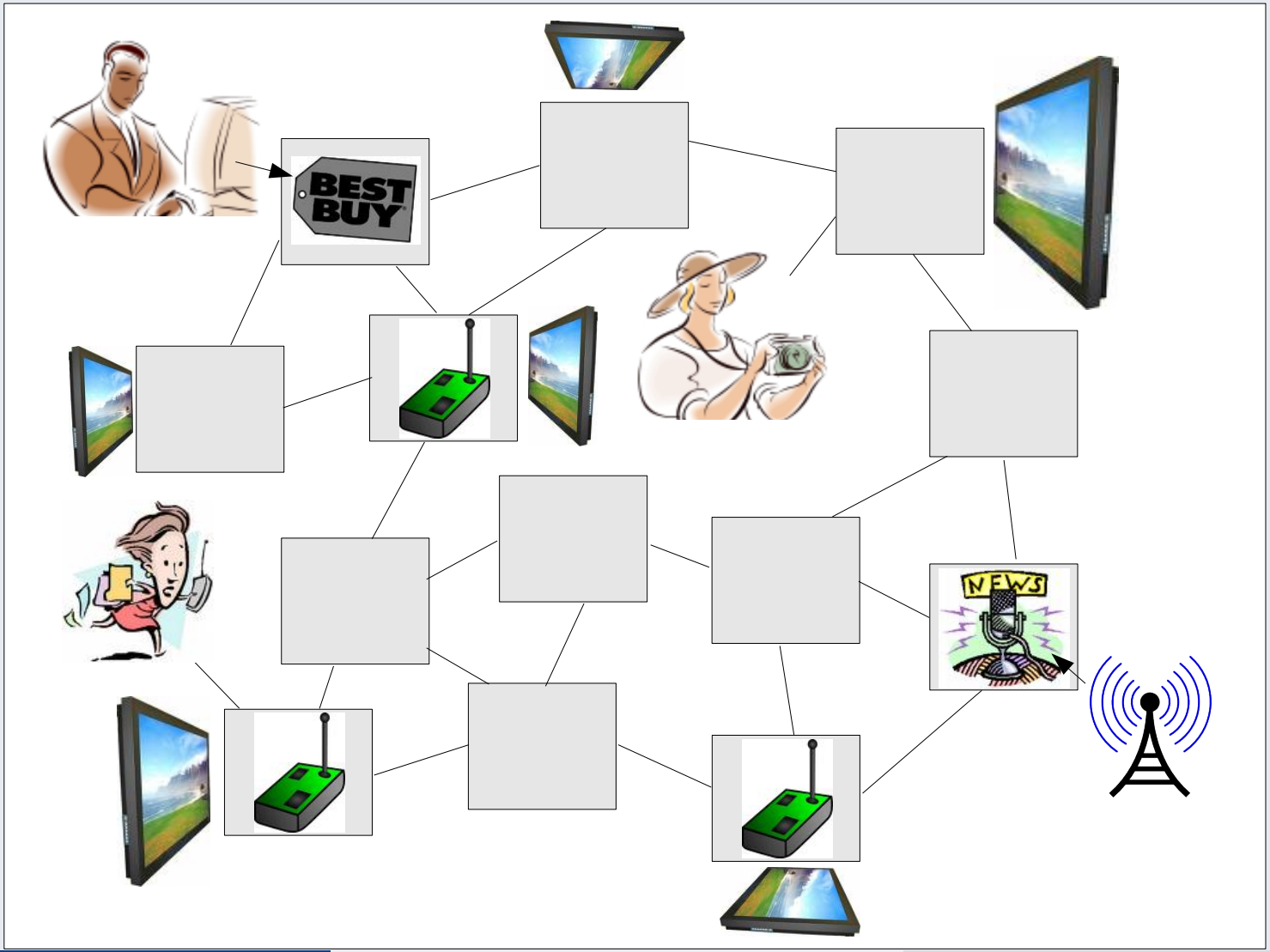
A scenario of pervasive displays

Displays should adapt visualisation to users nearby



A scenario of pervasive displays

Sensors provide contextual-information to improve adaptation



Some interesting services and features

- Displays show information based on majority of people around
- Alerts are shown as a given person passes nearby
- Displays coordinate to avoid irritating users
- Displays coordinate to provide visualisation streams
- Adjacent displays show a common, bigger content
- Injection of a new display cause redirection there
- A display used as a shared map
- Using eye-glasses with eye-pointers for immersed interaction

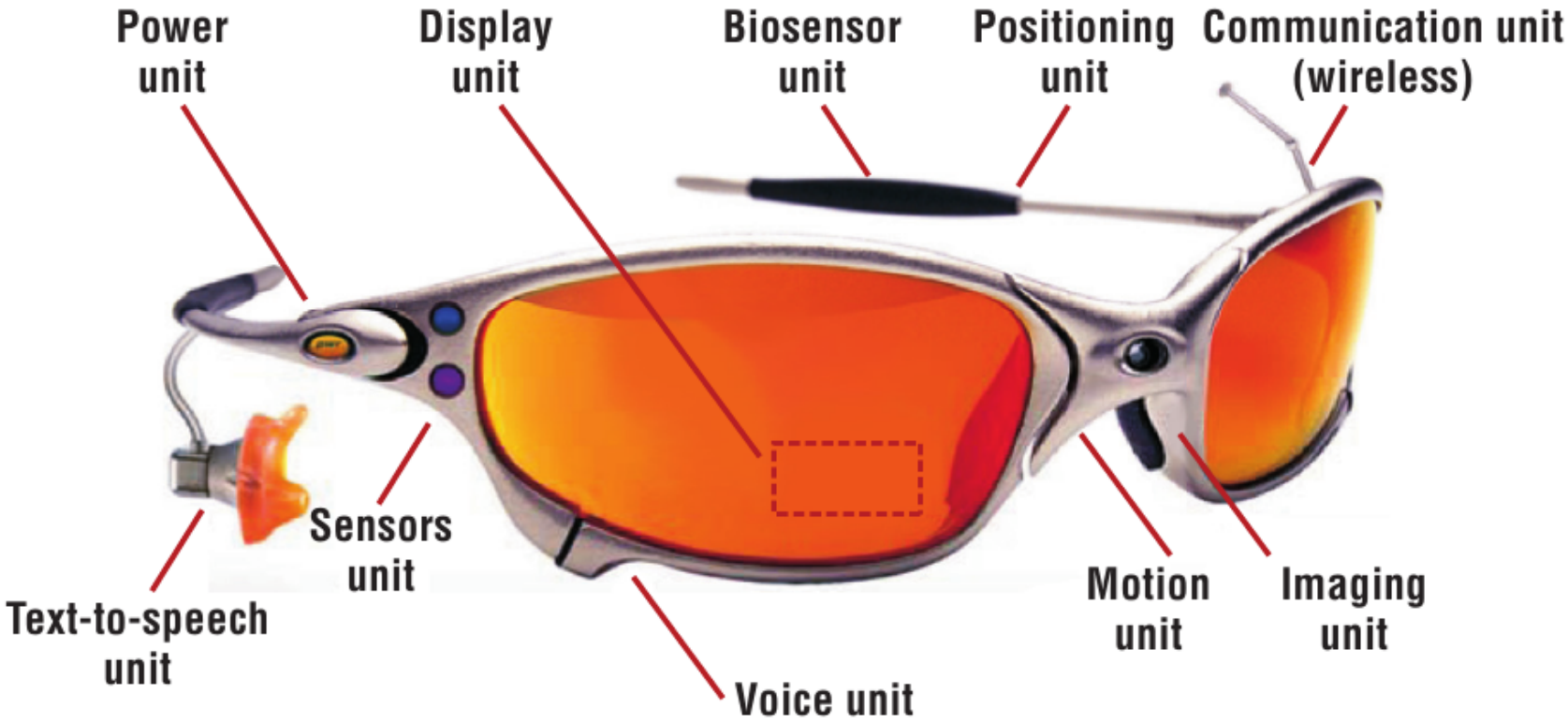
User-Display Coordination

(Thanks to JKU for the picture)

Enhancing displays

(Thanks to JKU for the picture)

New displays

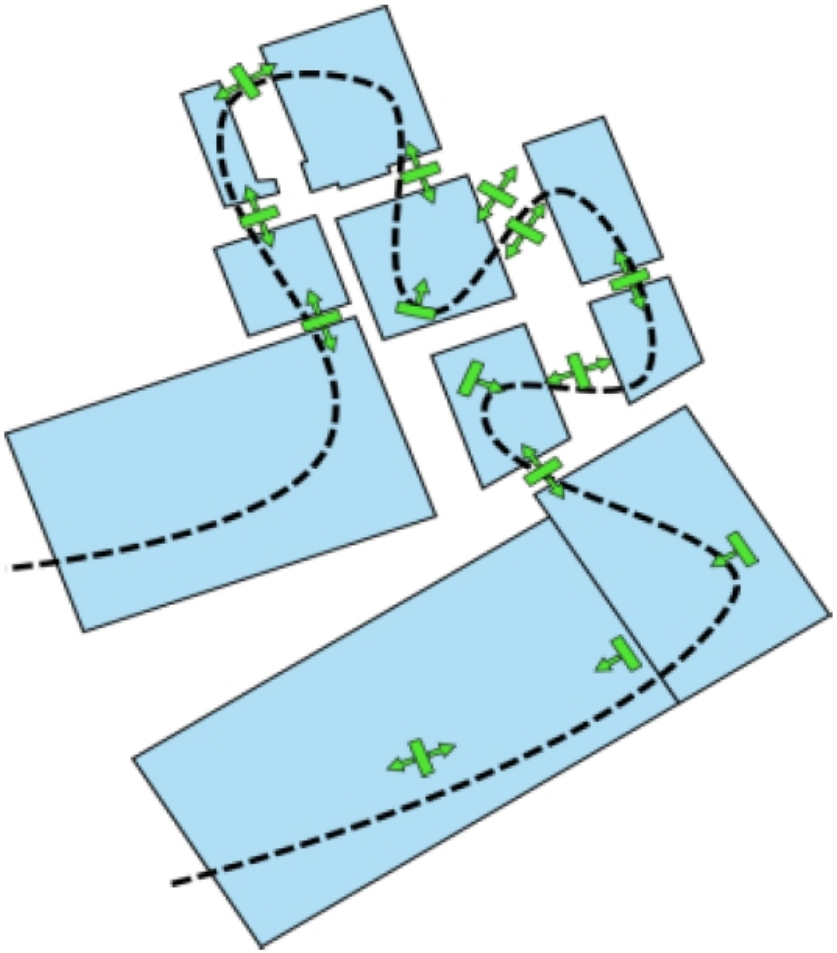


(Thanks to JKU for the picture)

Displays Coordination

(Thanks to JKU for the picture)

Content Coordination

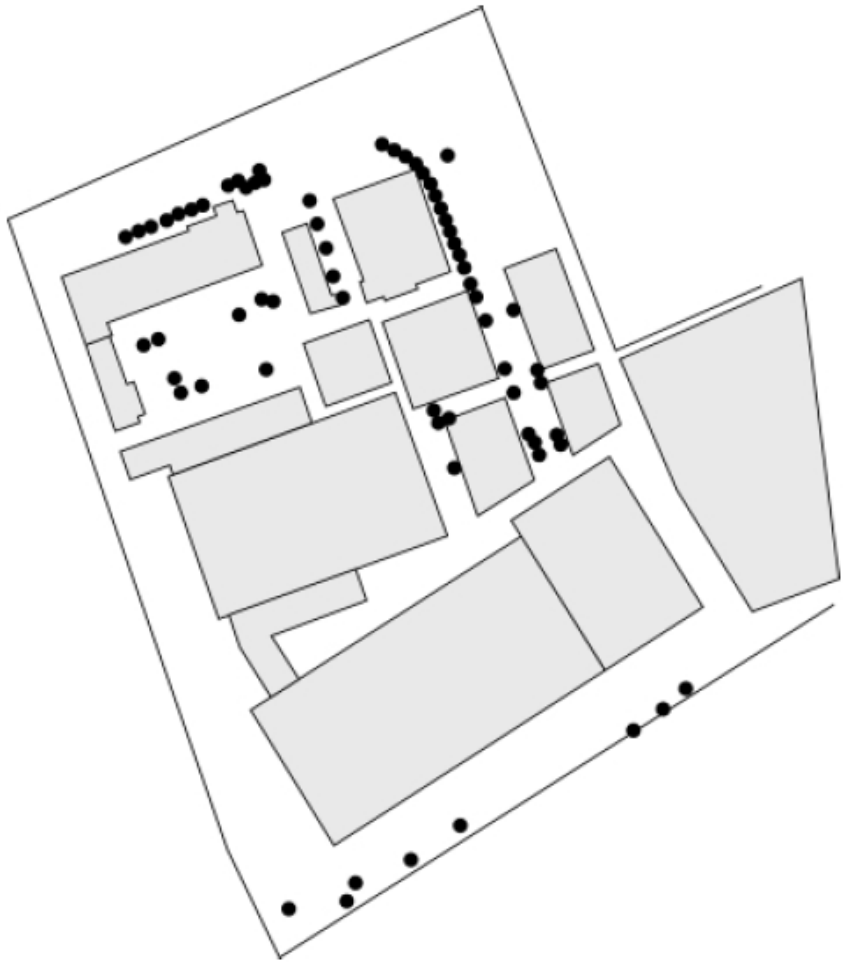


(Thanks to JKU for the picture)

Single-user steering

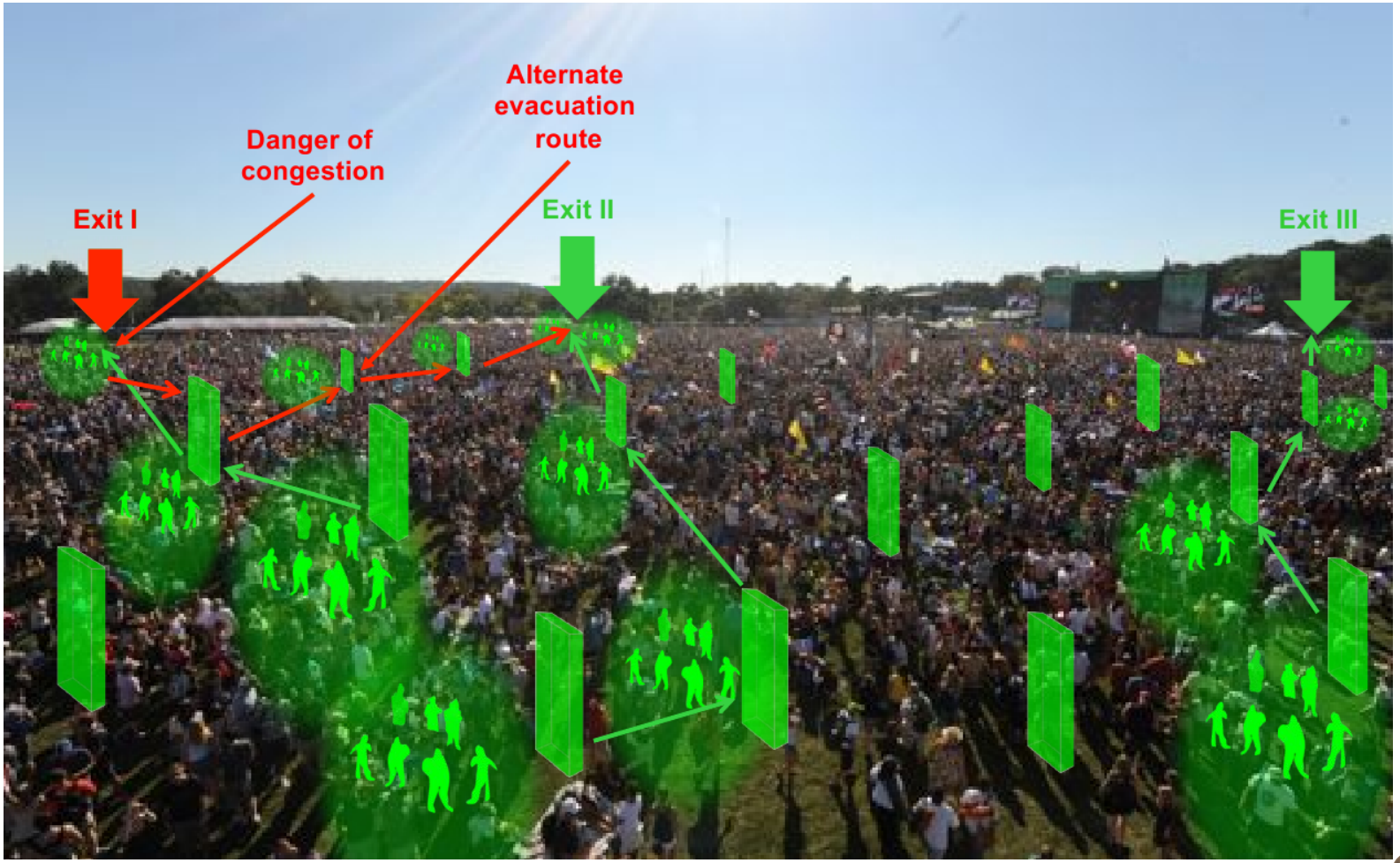
(Thanks to JKU for the picture)

Crowd steering in complex environments



(Thanks to JKU for the picture)

Crowd engineering



(Thanks to JKU for the picture)

Crowd-based application scenario

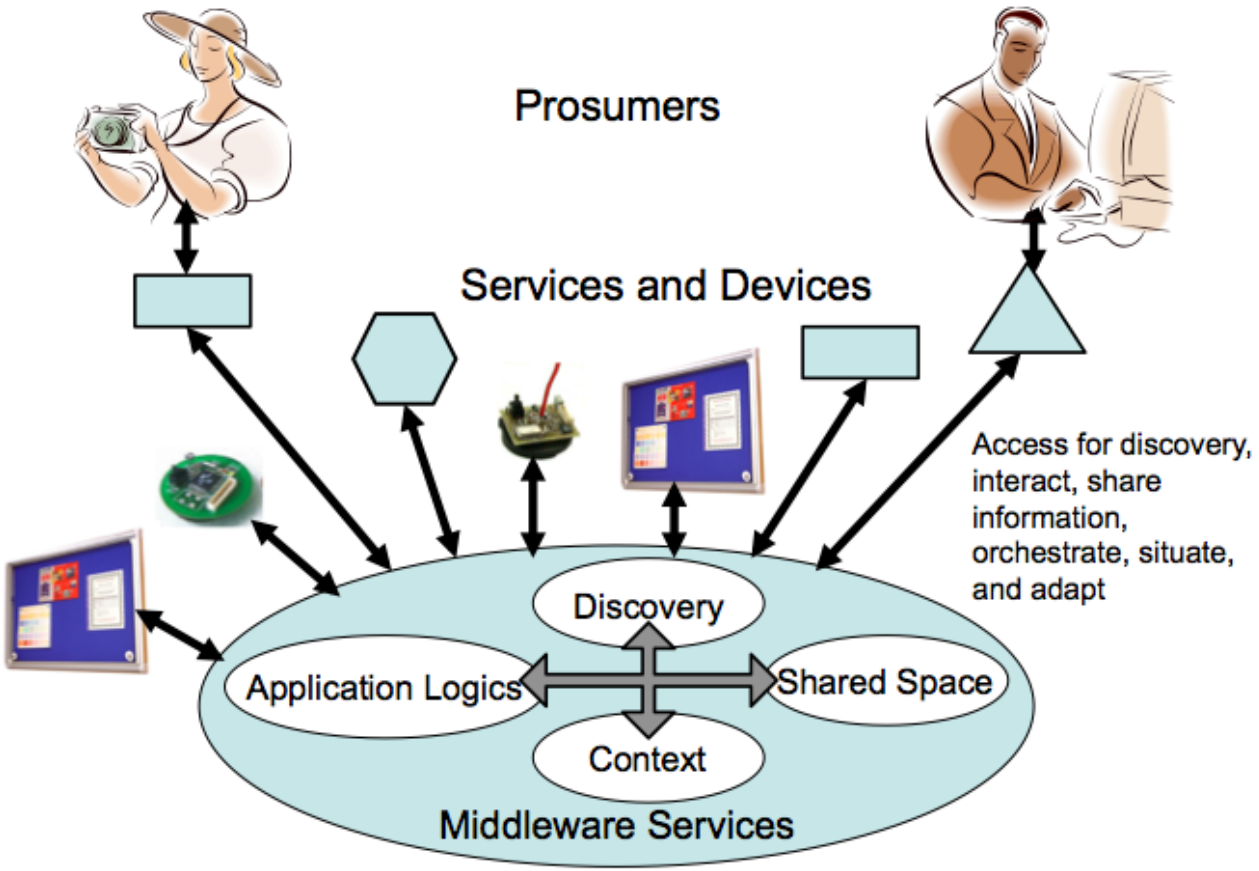
Other cases

- Single User Steering via Public Displays
- Crowd Balancing
- Evacuation
- Intention Driven User Steering
- Adaptive Advertisement
- Collective Guidance

Abstract of SAPERE

The objective of SAPERE is the development of a highly-innovative theoretical and practical framework for the decentralized deployment and execution of self-aware and adaptive services for future and emerging pervasive network scenarios. The framework will be grounded on a foundational re-thinking of current service models and of associated infrastructures and algorithms. In particular, getting inspiration from natural ecosystems, the project will demonstrate and experiment the possibility of modelling and deploying services as autonomous individuals in an ecosystem of other services, data sources, and pervasive devices, and of enforcing self-awareness and autonomic behaviours as inherent properties of the ecosystem, rather than as peculiar characteristics of its individuals only.

A standard, centralised SOA solution



A standard, centralised SOA solution

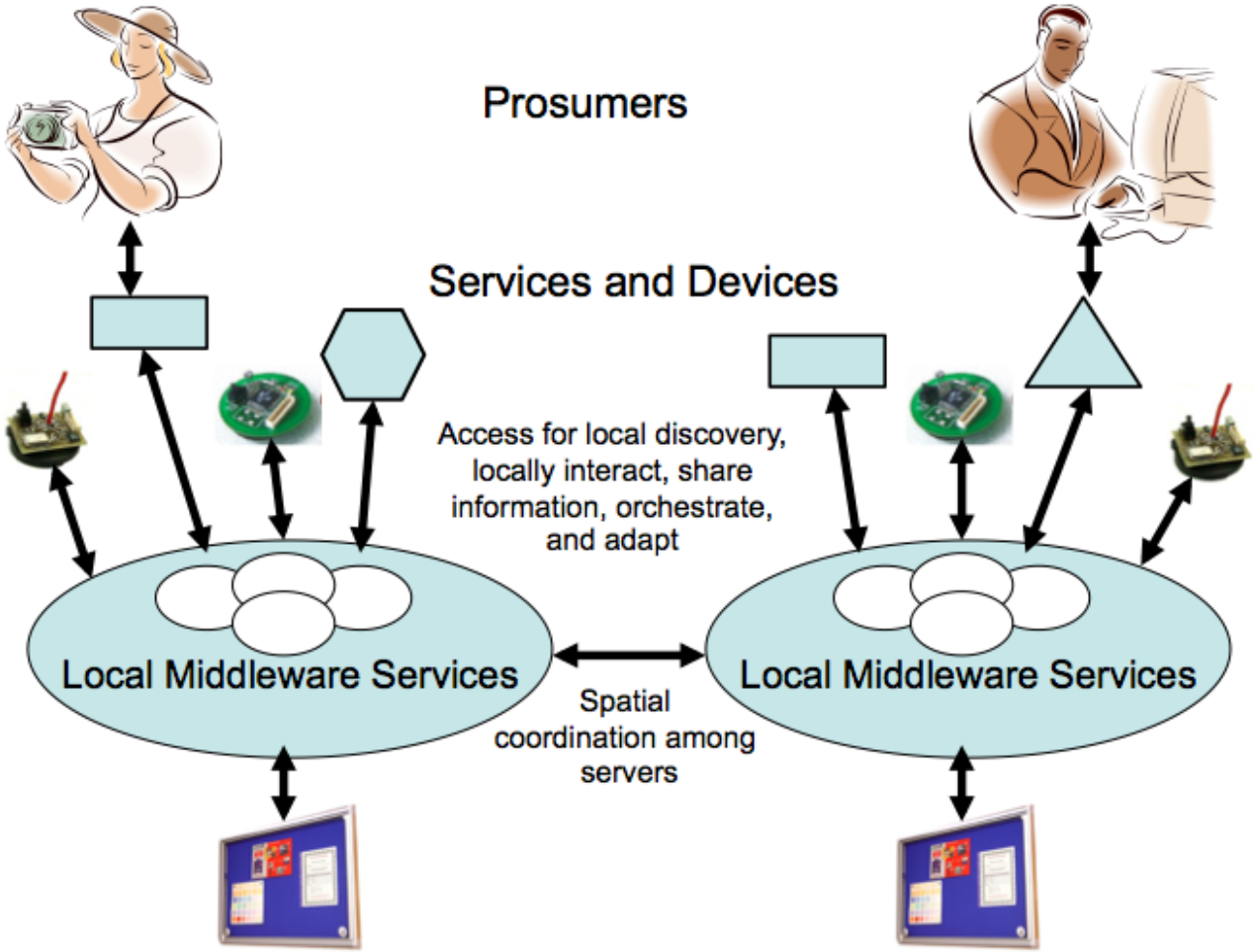
A centralised solution

One service for:

- Discovery: what components are available in the system?
- Context: where are components? (behaviour specialisation)
- Orchestration: coordinating components
- Shared space: depositing/retrieving local information
- Adaptation: reacting to contingencies

All components interact through such middleware services

De-centralising the SOA solution

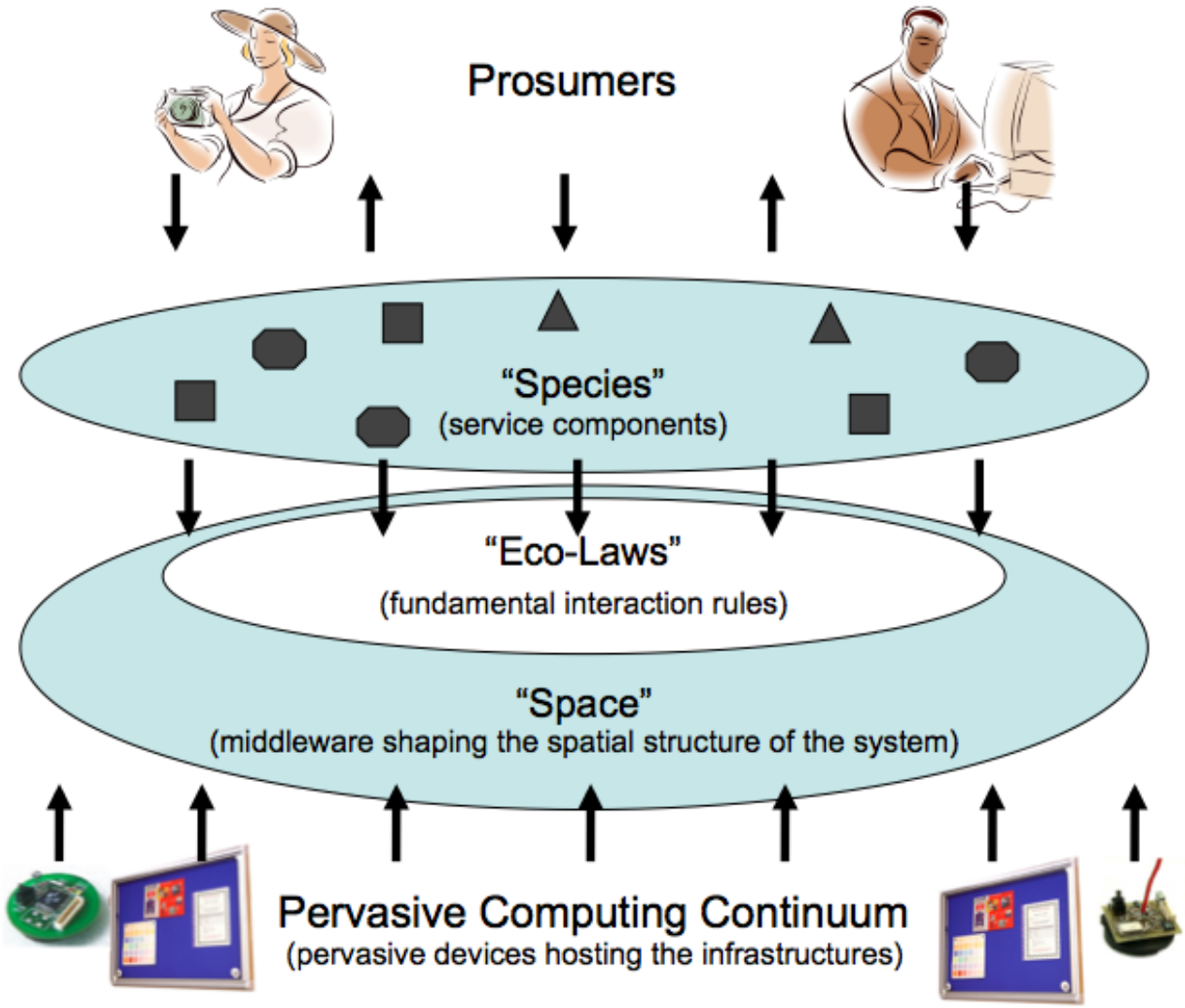


De-centralising the SOA solution

Duplicating middleware services in each location

- Better deals with situated components
- Simplifies contextualisation, discovery, and orchestration
- The role of shared spaces becomes more important
- Adaptation is still complex and crucial

Eco-inspired SOA solution



Eco-inspired SOA solution [16]

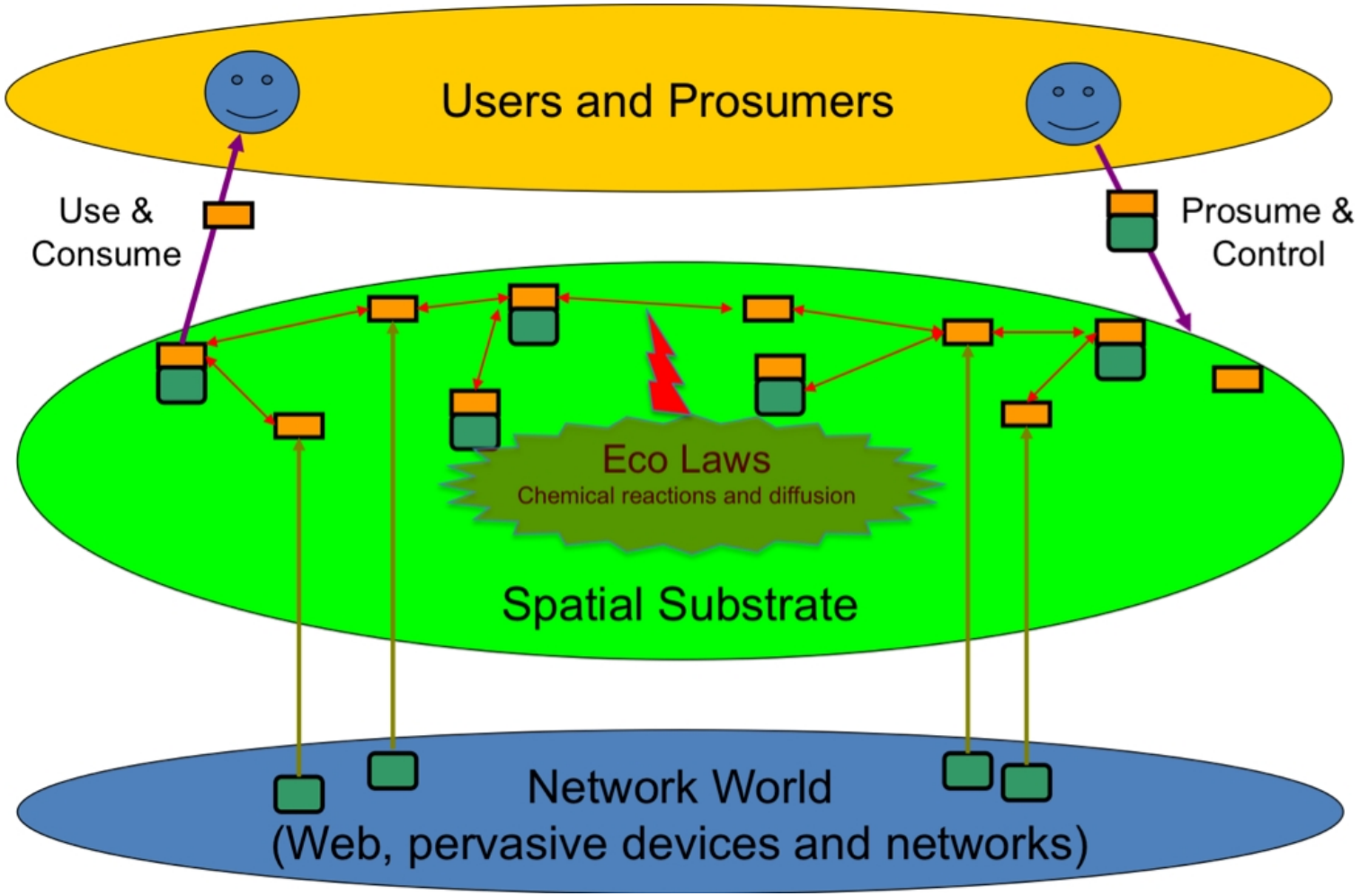
Fully decentralised middleware services

- Locations become very small and form a huge dynamic set
- Contextualisation, discovery, and orchestration almost vanish
- Middleware service just as a single space
- In overall we have a network of spaces with service "tags"
- Adaptation is achieved by simple rules combining tags

Drawing a bridge with natural ecosystems

We have a set of spatially situated entities interacting according to well-defined set of natural laws enforced by the spatial environment in which they situate, and adaptively self-organizing their interaction dynamics according to their shape and structure

Abstract Architecture



Pervasive Ecosystems [15]

Example Patterns

Data/devices/services are added by injecting their tags, then:

- they connect to other tags, supporting agent-to-agent interaction
- they could diminish until their population extinguishes
- they compete with other "species", and may survive
- they compose with patches injected to improve them
- they diffuse around
- they move where the context is favourable
- they aggregate with other copies, forming an overlay

Eco-laws and Live Semantic Annotations

Live Semantic Annotations (LSA)

- A unified description for devices, data, services
- Is about interface, status, and behaviour of a component
- It provides semantic information, and it is dynamic

Eco-Laws

- They resemble chemical reactions
- They take some reagent LSA, and provide some product LSA
- They can diffuse an LSA in the neighborhood
- They can aggregate LSAs like in chemical bonding
- They form a small & fixed set of natural eco-laws

Outline

What is SAPERE

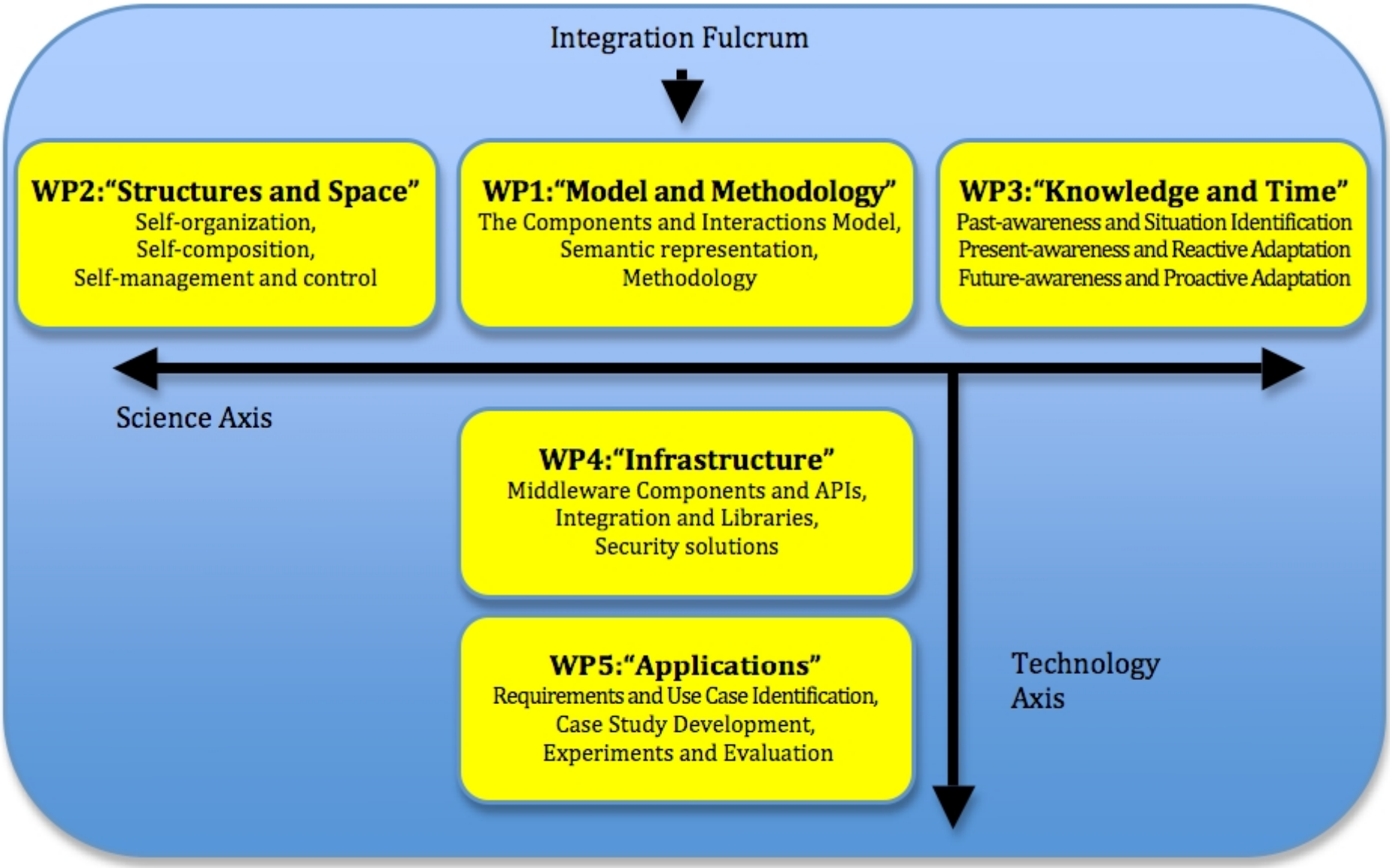
A European Project

- European call for funding projects: FP7-ICT-2009.8.5
- Call name: Self-awareness in Autonomic Systems
- Call type: Future and Emerging Technologies (FET)
- Funding Scheme: STREP (Specific Targeted Research Project)
- Period: 1/10/2010 - 30/9/2013
- Overall research grant: $\approx$ 2.5MEuro
- Consortium
  - Università di Modena e Reggio Emilia – Franco Zambonelli
  - Università di Bologna – Mirko Viroli
  - University of Geneve – Giovanna di Marzo
  - University of St Andrews, Dublin – Simon Dobson
  - Johannes Kepler Universitaet Linz – Alois Ferscha

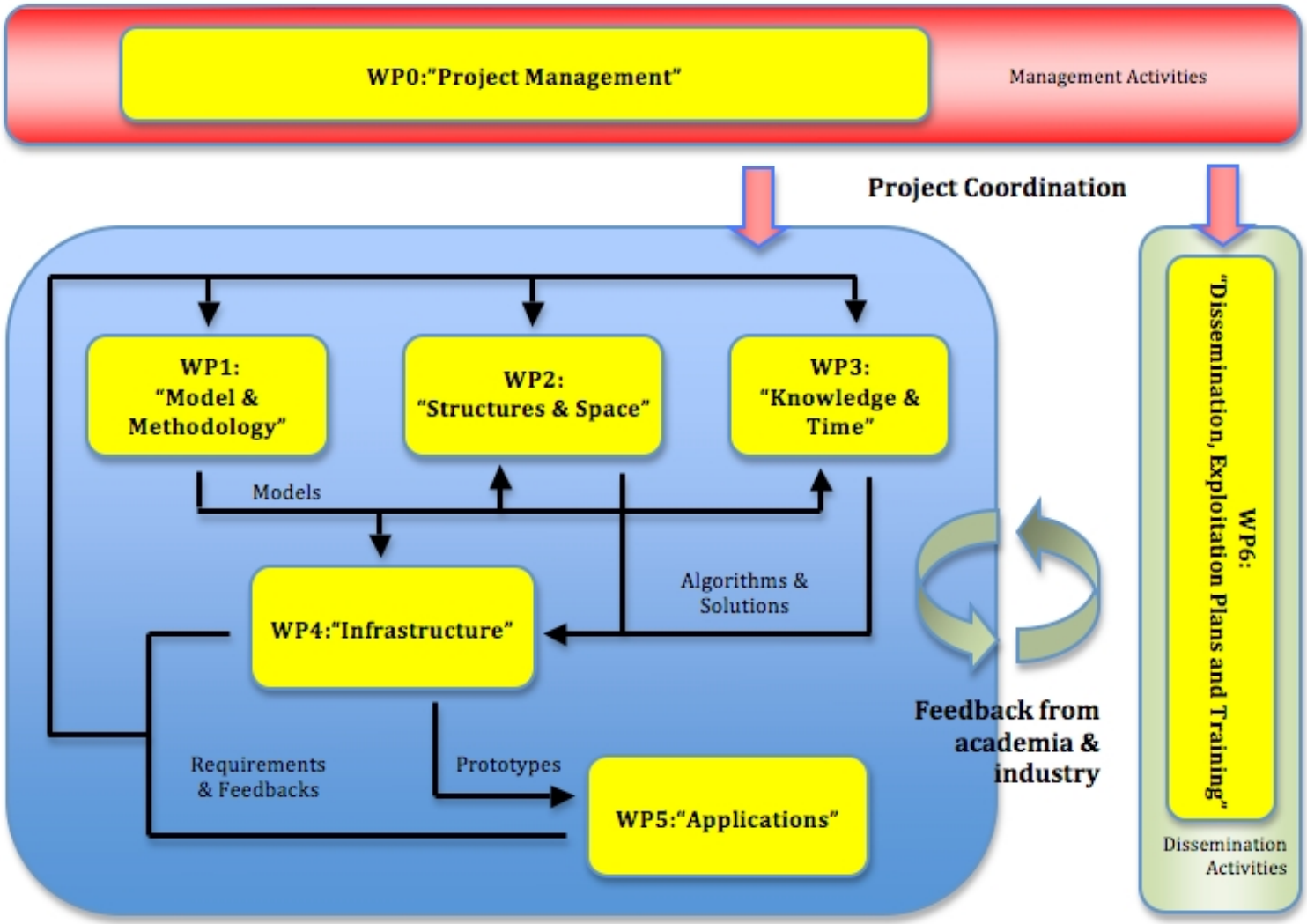
Elements of a Project Document (in general)

- Abstract
- Objectives
- Novelty and Contribution w.r.t. existing works
- Workplan (division in tasks, timing, efforts)
- Technical descriptions
- Deliverables
- Measure success indicators
- Dissemination activities
- Financial aspects

Strategy


Integration Fulcrum
WP2:"Structures and Space"
Self-organization,
Self-composition,
Self-management and control
WP1:"Model and Methodology"
The Components and Interactions Model,
Semantic representation,
Methodology
WP3:"Knowledge and Time"
Past-awareness and Situation Identification
Present-awareness and Reactive Adaptation
Future-awareness and Proactive Adaptation
Science Axis
WP4:"Infrastructure"
Middleware Components and APIs,
Integration and Libraries,
Security solutions
WP5:"Applications"
Requirements and Use Case Identification,
Case Study Development,
Experiments and Evaluation
Technology
Axis

Activities


WP0:"Project Management"
Management Activities
Project Coordination
WP1: "Model & Methodology"
WP2: "Structures & Space"
WP3: "Knowledge & Time"
Models
Algorithms & Solutions
WP4:"Infrastructure"
Requirements & Feedbacks
Prototypes
WP5:"Applications"
Feedback from academia & industry
WP6: "Dissemination, Exploitation Plans and Training"
Dissemination Activities

WP1: Model & Methodology – Leader UNIBO

Task T1.1 – The Components and Interaction Models

- Abstract (representation independent) model of eco-laws
- Services structure and interactions
- Studying analysis tools for behaviour verification

Task T1.2 – Semantic representation

- Shape of LSAs
- Studying analysis tools for logic reasoning

Task T1.3 – Methodology

- Finding a SE methodology
- Conceiving tools for development/analysis

WP1: Deliverables

D1.1 – Early Operational Model (M12) – Editor: UNIBO

- First version of abstract computational model
- First version of live semantic annotation framework
- Early demonstration of the operational model at work

D1.2 – Complete Operational Model (M20) – Editor: UNIBO

- ...

D1.3 – Final Operational and Semantic Model (M24) – Editor: UNIBO

- ...

D1.4 – Early Report on Methodology (M24) – Editor: STA

- ...

D1.5 – Methodology and Analysis Suite (M32) – Editor: UNIBO

- ...

D1.6 – Final Report on Methodology and Suite (M36) – Editor: UNIBO

- Complete and refined engineering methodology documentation
- Assessment of methodology and of associated tools.

Efforts

Partic. no.	Partic. short name	WP0	WP1	WP2	WP3	WP4	WP5	WP6	Total person months
1	UNIMORE	**15**	12	0	22	**34**	16	2	101
2	BIRKBECK	1	15	**36**	0	21	0	1	74
3	STA	1	13	8	**36**	0	15	1	74
4	UNIBO	1	**36**	20	0	20	0	4	81
5	JKU	1	0	16	16	0	**40**	2	75
Total		19	76	80	74	75	71	10	405

UNIBO details

- Leader of WP1: "Model and Methodology"
- Leader of WP6: "Dissemination, Exploitation, .."
- Involved in WP2: "Structures & Space"
- Involved in WP4: "Infrastructure"

Outline

Key Features – and corresponding architectural elements

Toleration of diversity: Live Semantic Annotations (LSA)

- Each individual (agent) is represented by one (or more) LSAs
- They uniformly reflect/affect the agent inner state

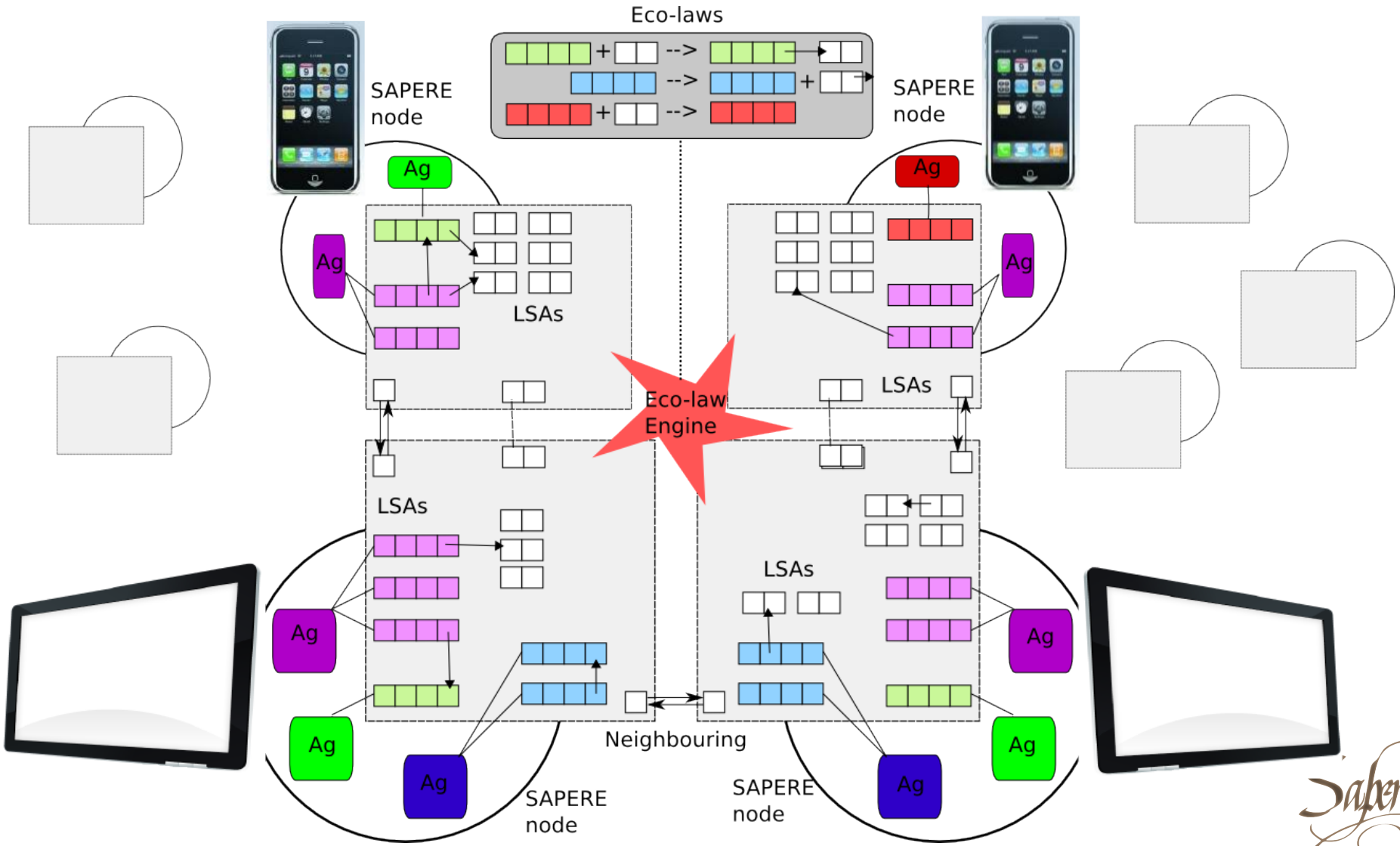
Situatedness: Spaces and Bonding

- LSAs are spread in LSA-spaces, each hosted by a device
- LSAs can connect to other via bonds, allowing mutual observation

Adaptivity: Eco-laws

- They create/change/delete/move LSAs, promoting agent interaction
- They can enact self-organisation, i.e., structures robust to changes

Operational Architecture [14]



A natural metaphor

We chose chemistry as a reference metaphor [16]

- LSAs as chemicals, eco-laws as reactions
- LSA-spaces as "ambients" supporting diffusion
- All tightly bridged with Semantic Web Technologies (RDF/SPARQL)

Main ingredients

Situation	Agents are situated in SAPERE nodes
Action	Agents manifest through local LSAs
Context	LSAs can bond to others around
Observation	An agent observes the world via bonds from his LSAs
Reaction	Eco-laws manipulate LSAs semantically
Diffusion	Eco-laws can relocate LSAs

LSAs (serialised as RDF [2])

Structure of LSAs

- A pair of a unique LSA-id, and a semantic description (SD)
- SD as a set of multi-valued (ontology-based) properties, as in RDF
- Overall: an RDF-like graph (Resource Description Framework)
  - RDF as a set of subject, predicate, object triples
  - Strings or URIs (Uniform Resource Identifiers)

Example: The LSA of a crowd-sensor in a museum scenario

```
lsa:crowdsensorlsa1123
    eco:#time "2011-05-30T11:00:00";
    eco:#loc sid:node34164@room132;
    eco:type museum:crowdsensor museum:contextlsa;
    museum:sensingtime "2011-05-30T11:31:24";
    museum:crowdlevel "0.93";
```

Eco-laws

Basic facts

- Chemical rules over LSA patterns: `P+...+P --r--> Q+...+Q`
- Patterns mean pre-/post-conditions, applied at rate r
- Patterns are template of LSAs
  - Constrained variables written `?V(filter)`
  - Can check values for presence (keyword "+", assumed by default), absence ("−"), unique existence ("=")

Triggering a display ?DIS visualisation of ?ADV because of ?USR

```
?DIS eco:type museum:display; museum:status ="ready";  +
?ADV eco:type museum:ad; museum:content ?C;            +
?USR eco:type museum:usr; museum:profile ?P(?P matches ?C);
--r-->
?DIS museum:status ="showing"; museum:service ?ADV;
+ ?ADV + ?USR
```

Eco-laws to SPARQL [1]

Triggering a display ?DIS visualisation of ?ADV because of ?USR

```
?DIS eco:type museum:display; museum:status ="ready";  +
?ADV eco:type museum:ad; museum:content ?C;            +
?USR eco:type museum:usr; museum:profile ?P(?P matches ?C);
--r-->
?DIS museum:status ="showing"; museum:service ?ADV;
+ ?ADV + ?USR
```

Equivalent SPARQL(/SPARUL) formulation

```
SELECT DISTINCT * WHERE{
  ?DIS eco:type museum:display .
  ?DIS museum:status "ready" .  FILTER NOT EXISTS { ?DIS museum:status ?o . FILTER (?o!= "ready") }
  ?ADV eco:type museum:ad .
  ?ADV museum:content ?C .
  ?USR eco:type museum:user .
  ?USR museum:profile ?P .      FILTER(?P rdf:type ?C) .
}
REMOVE DATA {!DIS museum:status ?o}
INSERT DATA {!DIS museum:status "showing"}
INSERT DATA {!DIS museum:service !ASV"}
```

Motivations for using RDF/SPARQL

For the users

- RDF is a standard and open language for describing resources
- RDF can be relative to widely-used ontologies
- Ontologies allow one to check LSAs for correctness

For implementers

- Parsers for RDF/SPARQL already exist
- SPARQL-enabled engines exist [12], usable for scheduling/firing laws
- RDF/Ontologies intrinsically support forms of semantic matching

A more advanced eco-law: diffusion

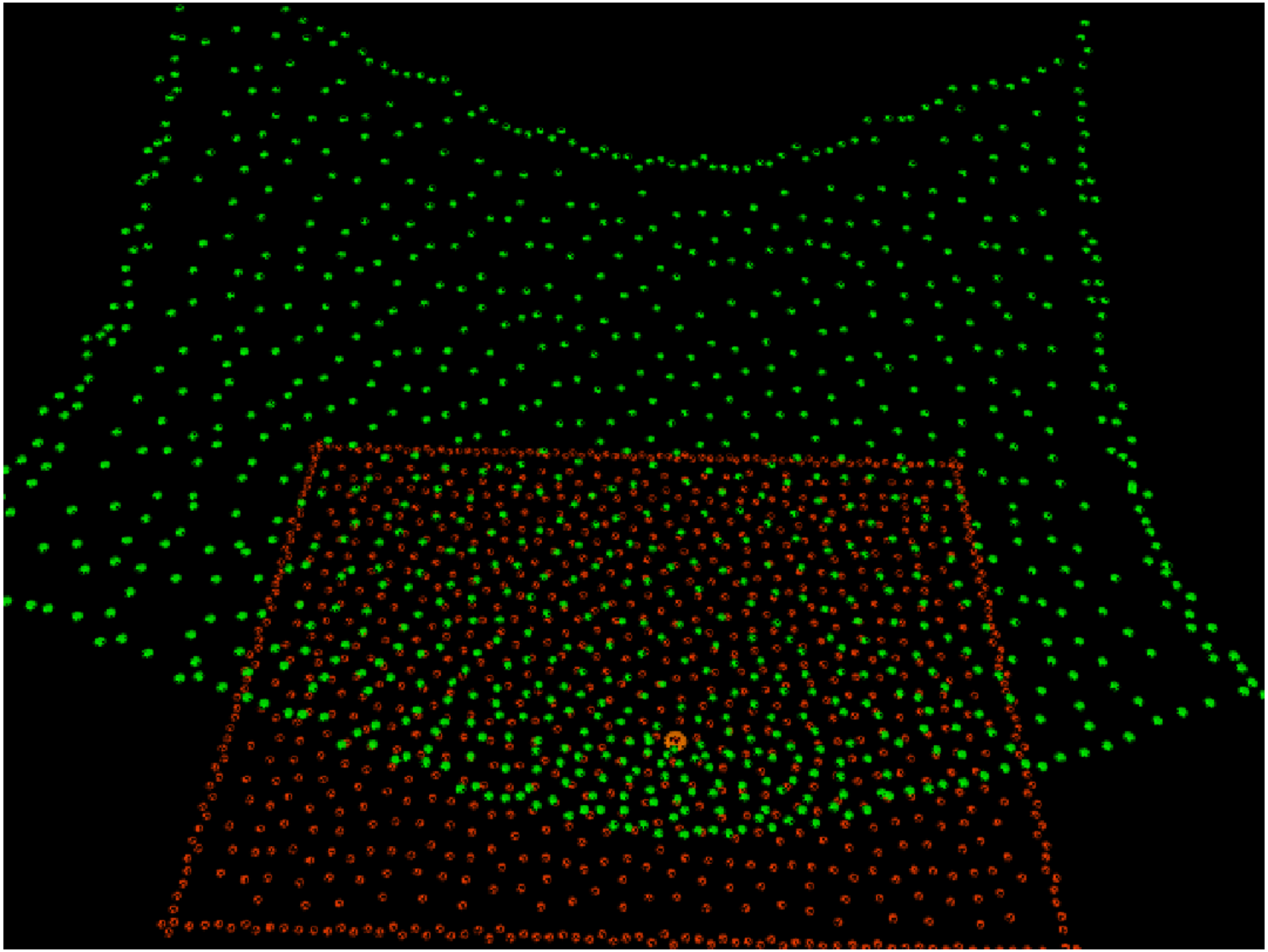
Goal

- It creates a *gradient* structure of LSAs with type `pump:field`
- Diffusion processes obtained by tweaking `eco:#loc` property
- It can be used to dynamically retrieve the gradient source
- (Should be coupled with aggregation eco-laws)

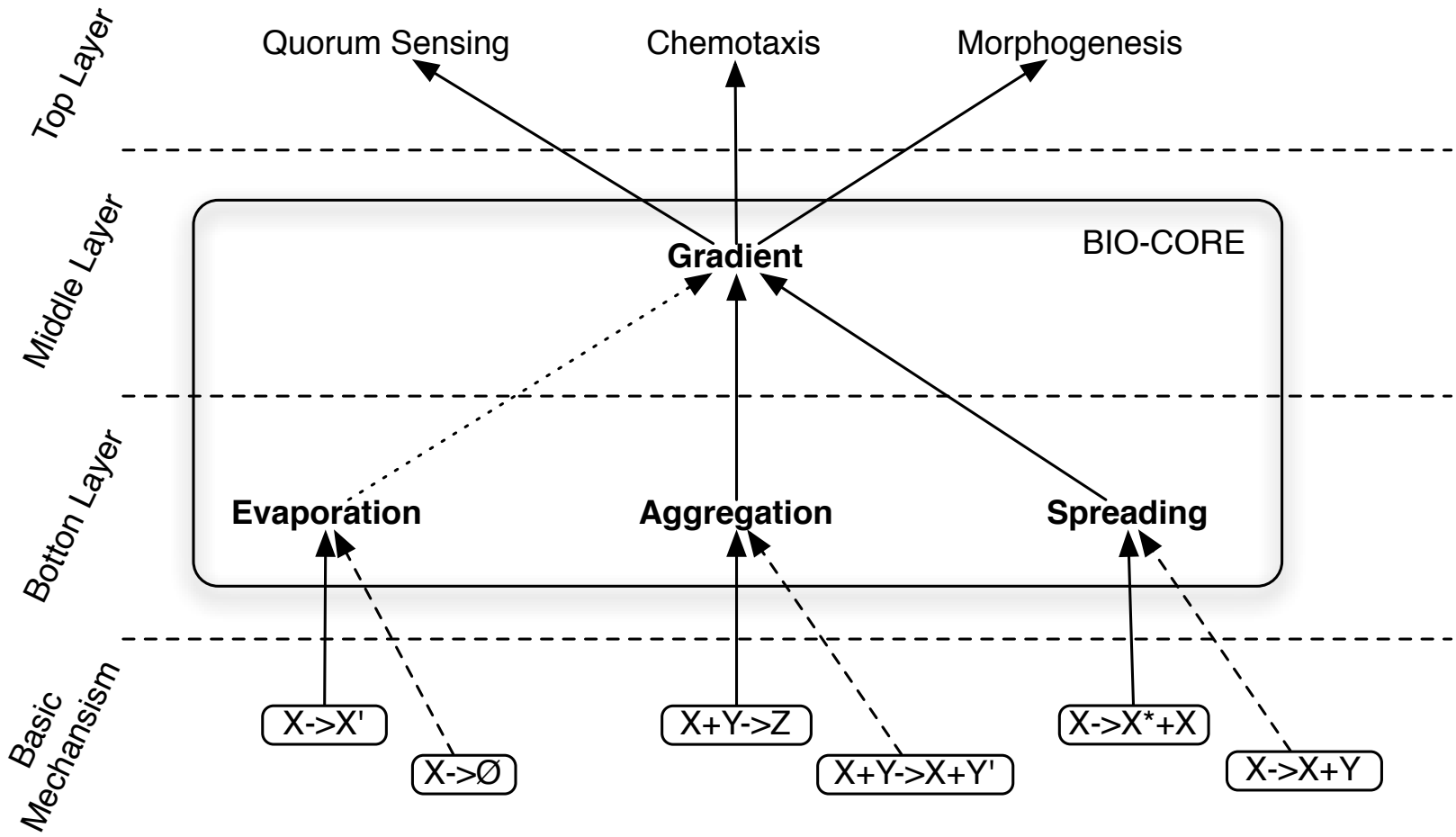
Diffuse in one neighbour a clone, with udpated distance to the source

```
?FIELD eco:type pump:field; eco:#loc ?L;
       pump:distance ?D; pump:range ?R; pump:diff_rate ?RT +
?NEIGH eco:type eco:neighbour; eco:neighbour_loc ?L1;
       pump:distance ?D1(?D1<?R-?D) +
?TIME eco:type eco:#time; eco:time ?T]
--?RT-->
?FIELD + ?NEIGH + ?TIME +
?CLONE(?CLONE extends ?FIELD) pump:distance ?D2(?D2 is ?D+?D1);
       pump:prev_loc ?L; pump:diff_time ?T;
       eco:#loc ?L1 ; eco:type pump:prefield;
       eco:type -pump:source -pump:field
```

A gradient structure (picture from [8])



Self-organisation Patterns [5]



A crowd steering example

An example of self-* ICT system

- Consider an articulated environment (e.g. a museum)
- The ground is full of nodes sensing people
- POIs appear, which could match people's profile/choice
- People is steered to the POI dynamically avoiding crowded areas

Eco-laws and a demo..

PUMP	A POI's LSA pumps a source gradient LSA
DIFFUSE	A gradient LSA is diffused with increasing distance value
YOUNGEST	Aggregate two LSAs keeping the most recent one
SHORTEST	Aggregate two LSAs keeping the one with shortest distance
CROWD	The presence of crowd increases distance value
BOND-EXT	A users binds with LSAs of POIs of interest

Related Works

Programmable Tuple Spaces

TuCSoN [10], MARS

⇒ We aim at more carefully balancing expressiveness and tractability

Semantic Tuple Spaces

RDFSwarms, TripleSpaces, Semantic TuCSoN

⇒ We smoothly move from syntactic tuples/matching to RDF/SPARQL

Self-organisation in Tuple Spaces

TOTA [7], SwarmLinda

⇒ We achieve self-organisation by laws of reaction/diffusion

Chemical Coordination

Gamma [4]

⇒ From syntactic/global chemistry to semantic/situated chemistry

Future Works ..track the "SAPERE" STREP project www.sapere-project.eu

On the general framework of pervasive service ecosystems

- Self-organisation patterns
- ALCHEMIST Simulator
- Tuple-based middleware
- Semantic Web reasoning
- Adaptive displays prototype

From the model viewpoint.. tackling predictability

- Sticking to a fixed set of eco-laws
- Translating to known-to-be-tractable formalisms
- Relying on the continuous abstraction (fluid analysis)

Outline

SAPERE and students

Opportunities

- Small Projects (for this exam, for LMC)
- Master Thesis
- Master Thesis to a project partner
- Post-degree trip to a partner
- Post-degree collaboration

Master thesis so far

Completed

- Tosi 12/2010 – Architecture of LSA-space
- Virruso 12/2010 – Eco-laws and fuzzy matching
- Santarelli 03/2011 – LSA-space with eco-laws
- Pianini 03/2011 – Simulator for self-organisation
- Desanti 10/2011 – Complete LSA-space Alpha
- Pronti 10/2011 – Complete Simulator Alpha
- Morgagni 10/2011 – Android Integration

Ongoing

- Cioffi 07/2012 – Self-organisation patterns
- Contessi 07/2012 – Semantic LSA-space

New opportunities

Macro-areas

- Developing applications
  - conceiving cases, experimenting Android integration
- Middleware implementation
  - building bricks of the middleware, tackling efficiency
- Simulating self-organisation cases
  - experimenting with new self-organisation patterns
- Developing Alchemist simulator
  - Reporting, input specification languages
- Prediction/Control of behaviour
  - Model-checking, relying on external analysis tools

Outline

References I

[1] ARQ - a SPARQL processor for Jena.
`http://jena.sourceforge.net/ARQ/`, 2011.

[2] RDF primer.
`http://www.w3.org/TR/rdf-primer/`, 2011.

[3] Chris Anderson.
The Long Tail: Why the Future of Business is Selling Less of More.
Hyperion, 2006.

[4] Jean-Pierre Bonâtre and Daniel Le Métayer.
Gamma and the chemical reaction model: Ten years after.
In *Coordination Programming*, pages 3–41. Imperial College Press London, UK, 1996.

References II

[5] Jose Luis Fernandez-Marquez, Giovanna Di Marzo Serugendo, Sara Montagna, Mirko Viroli, and Josep Lluis Arcos.

Self-organising design patterns.

Natural Computing, 2012.

To appear.

[6] Mehdi Jazayeri.

Species evolve, individuals age.

In *IWPSE '05: Proceedings of the Eighth International Workshop on Principles of Software Evolution*, pages 3–12, Washington, DC, USA, 2005. IEEE Computer Society.

[7] Marco Mamei and Franco Zambonelli.

Programming pervasive and mobile computing applications: The tota approach.

ACM Trans. Softw. Eng. Methodol., 18(4):1–56, 2009.

[8] MIT Proto.

software available at `http://proto.bbn.com/`, Retrieved Nov. 1st, 2010.

References III

[9] Elena Nardini, Andrea Omicini, and Mirko Viroli.

Description spaces with fuzziness.

In Mathew J. Palakal, Chih-Cheng Hung, William Chu, and W. Eric Wong, editors, *26th Annual ACM Symposium on Applied Computing (SAC 2011)*, volume II: Artificial Intelligence & Agents, Information Systems, and Software Development, pages 869–876, Tunghai University, TaiChung, Taiwan, 21–25 March 2011. ACM.

[10] Andrea Omicini and Franco Zambonelli.

Coordination for Internet application development.

Autonomous Agents and Multi-Agent Systems, 2(3):251–269, September 1999.

[11] Raghu Ramakrishnan and Andrew Tomkins.

Toward a peopleweb.

Computer, 40(8):63–72, 2007.

References IV

[12] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz.

Pellet: A practical OWL-DL reasoner.

Web Semant., 5:51–53, June 2007.

[13] Mirko Viroli and Matteo Casadei.

Biochemical tuple spaces for self-organising coordination.

In John Field and Vasco T. Vasconcelos, editors, *Coordination Languages and Models*, volume 5521 of *LNCS*, pages 143–162. Springer-Verlag, June 2009.

11th International Conference (COORDINATION 2009), Lisbon, Portugal, June 2009. Proceedings.

[14] Mirko Viroli, Danilo Pianini, Sara Montagna, and Graeme Stevenson.

Pervasive ecosystems: a coordination model based on semantic chemistry.

In Sascha Ossowski, Paola Lecca, Chih-Cheng Hung, and Jiman Hong, editors, *27th Annual ACM Symposium on Applied Computing (SAC 2012)*, Riva del Garda, TN, Italy, 26-30 March 2012. ACM.

References V

[15] Mirko Viroli and Franco Zambonelli.
A biochemical approach to adaptive service ecosystems.
Information Sciences, 180(10):1876–1892, 2010.

[16] Franco Zambonelli and Mirko Viroli.
A survey on nature-inspired metaphors for pervasive service ecosystems.
International Journal of Pervasive Computing and Communications, 7(3):186–204, 2011.