

## WP1-WP2 Boundary

Spatiality in the SAPERE model and in eco-laws

Mirko Viroli

`mirko.viroli@unibo.it`

ALMA MATER STUDIORUM—Università di Bologna

M4 Plenary Meeting, 20/1/2011, Genève, CH



# 1 Space in eco-laws (WP1-WP2)



# Space and WP1

## Space is key in pervasive computing applications and in SAPERE

- An agent occurring “here” is not the same as occurring “there”
- Things are very context-dependent

## Space is key in SAPERE model (hence in WP1)

- Some (all?) LSAs will describe information about space
- The semantics of eco-laws should take into account space

## Though, what should we mean by “space”?

That is, which topological model?

- In between “a single space” and “a flat net of physical nodes”
- Possibly a logical, multi-layered, mobile, emergent structure of locations

# Accordingly, two views of what is an eco-law

Disclaimer: the right view will probably be in the middle...

## Micro-level

- An eco-law is an instruction for the engine of a single LSA-space
- Reactants should be LSAs in the same space
- Some additional mechanism in eco-laws ( $X \rightarrow X + X^{\rightarrow}$ )
- Topology managed “on top” (topology-LSAs, space agents, self-org)

## Macro-level

- An eco-law is a “global” instruction for the ecosystem
- Reactants are co-fields, namely e.g. diffused LSAs
- Eco-laws abstract from space
- Mechanisms concerning space are “under the hood”



# Three approaches to tackle this duality

## Micro-approach

- Eco-laws are those at the micro-level
- Some of them are space-independent, some are spatial mechanisms
- WP2 algorithms mostly in eco-laws

## Macro-approach

- Only have macro-level eco-laws
- Spatial mechanisms are hardwired in the middleware
- WP2 algorithms mostly in the middleware

## What stays in the middle?

- Design two languages for eco-laws:  
define the semantics of macro-level by translation into micro-level
- Design some intermediate level, with topology making into eco-laws

# Hard to decide now...

## My proposal

- During first year assume the micro-approach (thought, it has some drawback...)
- During second year look for a generalisation
  - (Design application-independent eco-laws)
  - Perhaps raise the level of abstraction of eco-laws...
  - ... without throwing away what has been done

## An additional note

- not sure all high-level self-organisation patterns are to be “programmed” by eco-laws
- as for WP3, some agent may be required that supports complex reasoning



# A review of space-aware chemical reaction mechanisms

## How chemical reactions can be augmented?

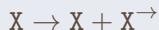
- there are several models/languages/systems we can take inspiration from
- I'd personally try to pick a very small set of features (ideally just one), so as to keep the eco-law language simple/coherent enough

## Approaches

- Firing tuple [Viroli and Casadei, 2009] (membrane computing)
- Firing tuple with gradient [Viroli et al., 2011b]
- Broadcast (Proto language [Viroli et al., 2011a], sensor networks)
- Local-remote coupling [Viroli, 2011]
- ... (something new for SAPERE?)

# Firing tuple

## Example abstract reaction



## Description

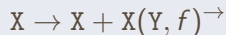
By this reaction, LSA  $X$  generates a so-called “firing” LSA, which is an LSA to be shipped into one neighbouring LSA-space, probabilistically

## Few facts

- This mechanism is inspired by biochemistry
- It is a quite simple mechanism, also to implement
- Not very expressive
- BTW: when an eco-law has to be executed?

# Firing tuple with gradient

## Example abstract reaction



## Description

The space where sending the firing LSA is chosen depending on the slope of gradient created by  $Y$  (difference from “concentration” of  $Y$  here and in the neighbouring space):  $f$  is a numerical value that can be used to control direction and/or speed.

## Few facts

- This mechanism is inspired by biochemistry
- It is a bit more complicated at the implementation level
- Can support ascension of a field, aggregation

# Broadcast

## Example abstract reaction

$$X([v]) + Y \rightarrow Z(g[v])$$

## Description

Variable  $v$  should occur on both sides,  $g$  is a gathering function from sets to values (e.g. *max*, *min*, *sum*). The reaction is applied to an LSA matching  $X(v)$ , let  $val$  be the value instantiating variable  $v$ . Then,  $val$  is spread to all neighbours by a broadcast. Each LSA-space hence receives a set of values for  $v$ : by function  $g$  we obtain a single value used in  $Z$ .

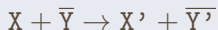
- E.g.:  $\text{grad}([N]) \rightarrow \text{grad}(\max[N] - 1)$

## Few facts

- This is the interaction mechanism of Proto language [Viroli et al., 2011a], nicely applying to e.g. sensor networks

# Local-remote coupling

## Example abstract reaction



## Description

This reaction executes in a (local) space where  $X$  occurs, and finds a neighbouring space where  $Y$  occurs. It accordingly changes  $X$  to  $X'$  locally, and  $Y$  to  $Y'$  remotely.

## Few facts

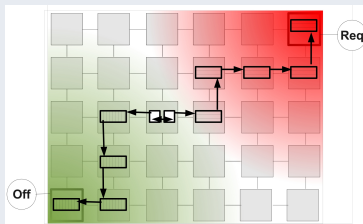
- This mechanism appears rather expressive
- It raises some implementation concerns



# Application of “Local-remote coupling” mechanism [Viroli, 2011]

## Reactions

|                                                                                                                                        |        |
|----------------------------------------------------------------------------------------------------------------------------------------|--------|
| $\text{source}(s, N) \rightarrow \text{source}(s, N) + \text{grad}(s, N)$                                                              | [INJ]  |
| $\text{grad}(s, N) \rightarrow \text{grad}(s, N) + \overline{\text{grad}(s, N-1)}$                                                     | [DIFF] |
| $\text{grad}(s, N) + \text{grad}(s, N+M) \rightarrow \text{grad}(s, N+M)$                                                              | [AGG]  |
| $\text{pstart}(r, s) + \text{grad}(s, N) \rightarrow \text{pstart}(r, s) + \text{grad}(s, N) + \text{comp}(r, s, N)$                   | [SOU]  |
| $\text{comp}(r, s, N) + \text{grad}(s, N) \rightarrow \text{comp}(r, s, N) + \text{grad}(r, s, N) + \overline{\text{path}(r, s, N+1)}$ | [CALL] |
| $\text{path}(r, s, N) + \text{grad}(s, N) \rightarrow \text{comp}(r, s, N) + \text{grad}(s, N)$                                        | [PATH] |



# Which pervasive applications to self-org mechanisms

## List

- Gradient: creating a distance-aware diffusion region
- Directed path: reaching a possibly mobile destination
- Clustering: easing reasoning about historical data
- Partitioning: avoiding clash of service variants



# A proposal for WP1/WP2 split of activities

## WP1

- Topological considerations
- Design of eco-laws  
(mechanisms for space, composition, aggregation of LSAs)

## WP2

- everything that can be done on top of eco-laws
  - self-\* algorithms (of all three levels)
  - controlling/managing also non functional properties  
(composition, aggregation, competition, decay)



# References I



Viroli, M. (2011).

Self-organising composition of competing networked services.

*Science of Computer Programming.*

Submitted.



Viroli, M., Beal, J., and Casadei, M. (2011a).

Core operational semantics of Proto.

In *26th Annual ACM Symposium on Applied Computing (SAC 2011)*,  
Tunghai University, TaiChung, Taiwan. ACM.



# References II



Viroli, M. and Casadei, M. (2009).

Biochemical tuple spaces for self-organising coordination.

In Field, J. and Vasconcelos, V. T., editors, *Coordination Languages and Models*, volume 5521 of *LNCS*, pages 143–162. Springer, Lisbon, Portugal.

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



Viroli, M., Casadei, M., Montagna, S., and Zambonelli, F. (2011b).

Spatial coordination of pervasive services through chemical-inspired tuple spaces.

*ACM Transactions on Autonomous and Adaptive Systems.*



# WP1-WP2 Boundary

Spatiality in the SAPERE model and in eco-laws

Mirko Viroli

`mirko.viroli@unibo.it`

ALMA MATER STUDIORUM—Università di Bologna

M4 Plenary Meeting, 20/1/2011, Genève, CH

