Bigraphs: a platform for Ubicomp?

Spring School for Ubicomp, 2009

Robin Milner, University of Cambridge

PARTS OF THE TALK

- Ubiquitous Computing, and modelling it
- Space, Motion and Bigraphs
- Examples of Bigraphical Systems
- Conclusion

References and Links

For full understanding these slides need spoken narrative!

But go to **`www.cambridge.org.uk/catalogue/`**
and search on *Robin Milner* to find my book on bigraphs, just published by Cambridge University Press:

The Space and Motion of Communicating Agents

On my website **`www.cl.cam.ac.uk/~rm135/`** find

- **Slides** for a six-lecture introductory course
- **Lecture notes** linking the course to the book
- **Exercises** and **solutions**

PARTS OF THE TALK

- **Ubiquitous Computing, and modelling it**
- **Space, Motion and Bigraphs**
- **Examples of Bigraphical Systems**
- **Conclusion**

Qualities of a ubiquitous computing system (UCS)

What is new about a UCS?

- It will continually make **decisions** hitherto made by us
- It will be **vast**, maybe 100 times today's systems
- It must continually **adapt**, on-line, to new requirements
- Individual UCSs will **interact** with one another

Can traditional software engineering cope?

Concepts for Ubicomp

Each ubicomp **domain**, hence each **model**, will involve several concepts. Here are a few:

provenance obligations
self-management
locality intentions specification data-protection
beliefs continuous space authorisation simulation
encapsulation mobility continuous time role
compilation policy failure
delegation reflectivity verification
stochastics negotiation connectivity
trust security authenticity

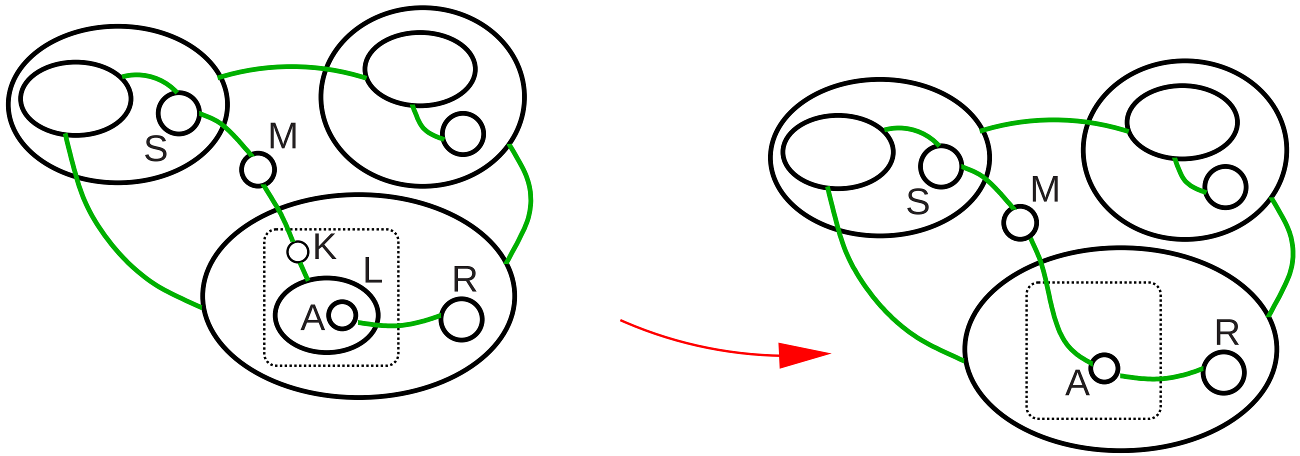
Managing the conceptual overload

- Define **UAM**, the *Ubiquitous Abstract Machine*, in terms of locality, connectivity, mobility, stochastics.
- Build a *model tower* above **UAM**, layering the concepts.

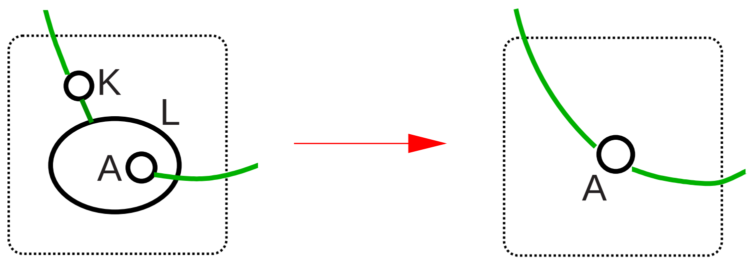
PARTS OF THE TALK

- Ubiquitous Computing, and modelling it
- **Space, Motion and Bigraphs**
- Examples of Bigraphical Systems
- Conclusion

A fanciful system, seen as a bigraph

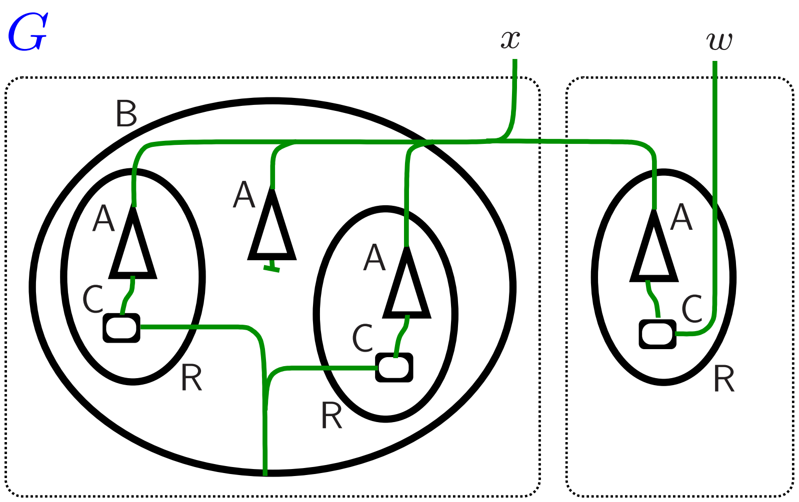


Reaction rule:



A built environment G

A - an agent
B - a building
C - a computer
R - a room



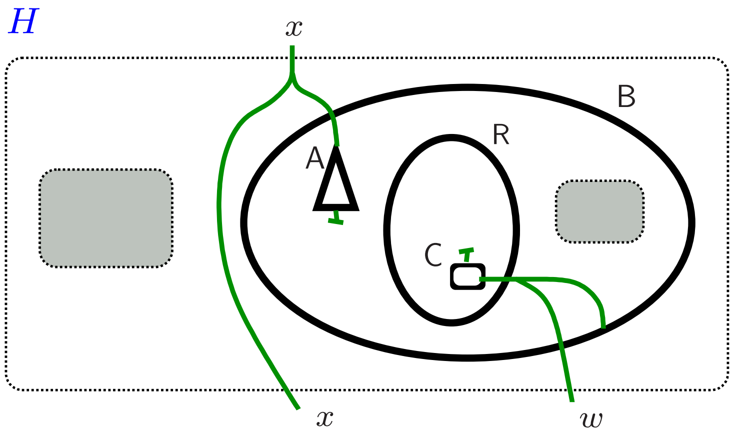
Each node has a *control*, with *arity*, e.g. A has arity 2.

$$G \;=\; /z\, \mathsf{B}_z.(\mathsf{Roomfull}_{xz} \mid /y \mathsf{A}_{xy} \mid \mathsf{Roomfull}_{xz}) \;\|\; \mathsf{Roomfull}_{xw}$$

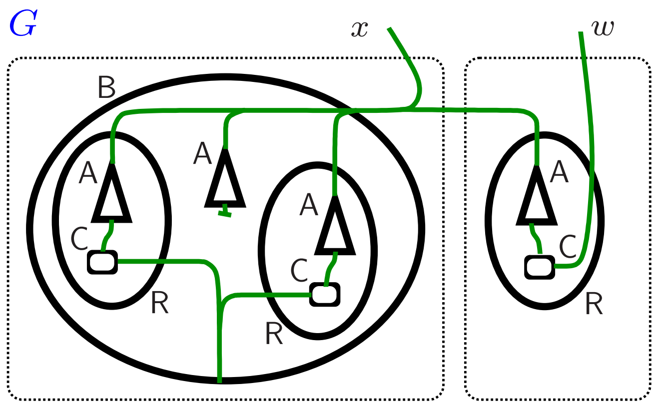
where $\mathsf{Roomfull}_{xz} \stackrel{\text{def}}{=} \mathsf{R}./y\,(\mathsf{A}_{xy} \mid \mathsf{C}_{yz})$.

The *signature* $\mathcal{K} = \{\mathsf{A}:2, \mathsf{B}:1 \ldots\}$ gives controls with arities.

...... and a host H for G

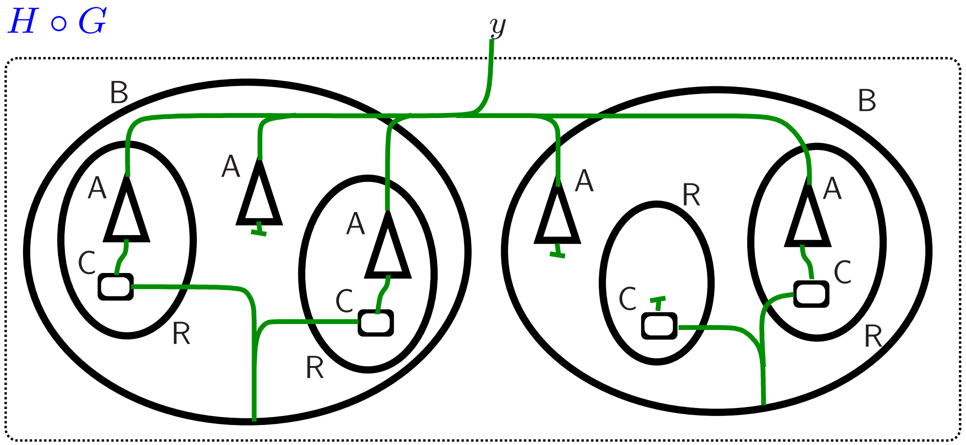


A – an agent
B – a building
C – a computer
R – a room

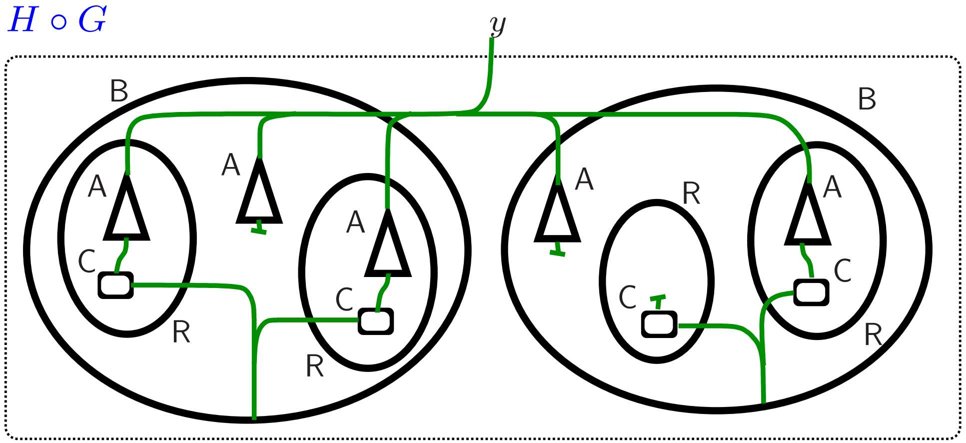


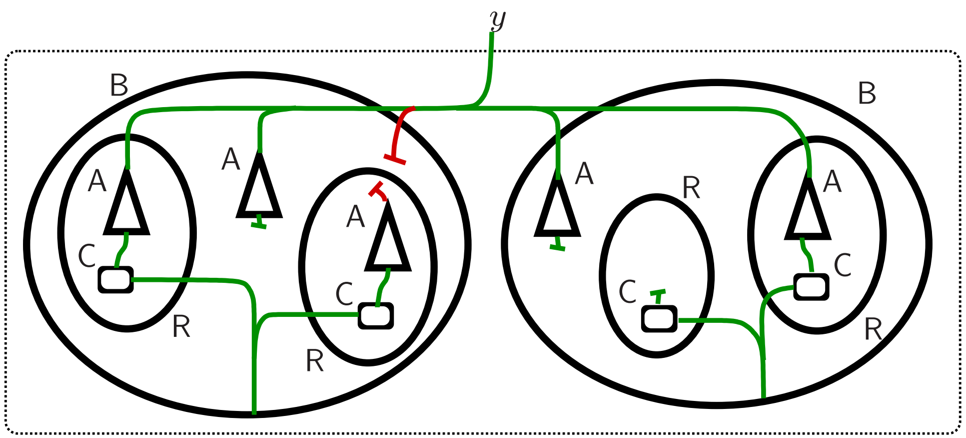
$$H = \mathsf{id}_1 \mid \mathsf{id}_x \mid /w\,\mathsf{B}_w.(/y\mathsf{A}_{xy} \mid \mathsf{R}./y\mathsf{C}_{yw} \mid \mathsf{id}_w \mid \mathsf{id}_1)\,.$$

The complete system $H \circ G$

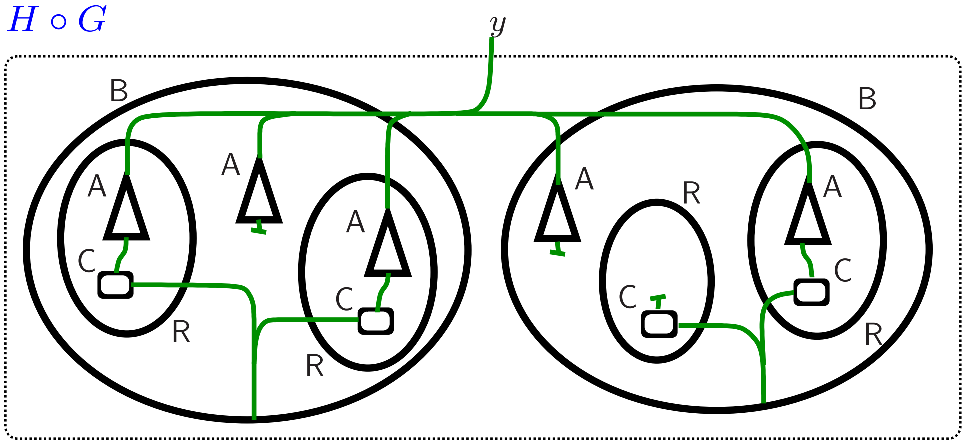


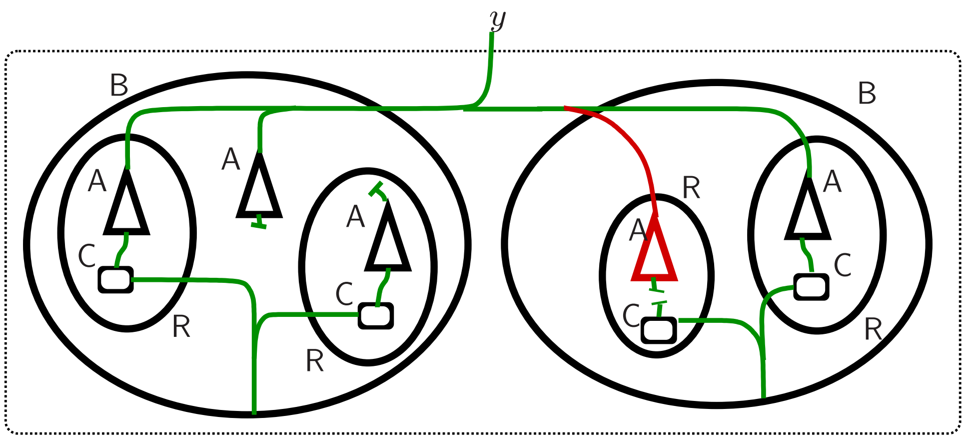
. . . . . . and after one reaction





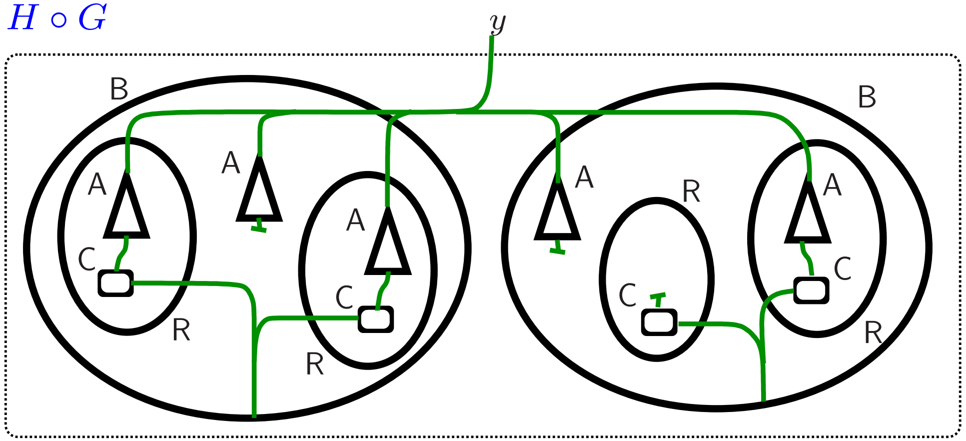
......and after two reactions

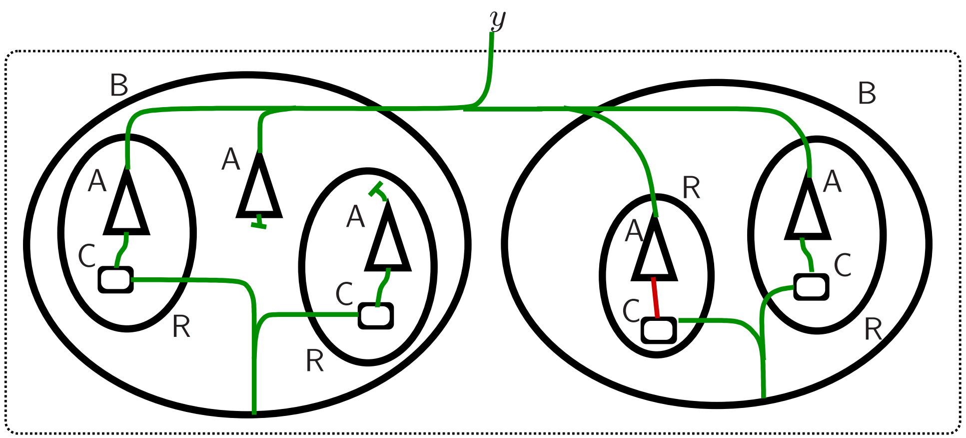




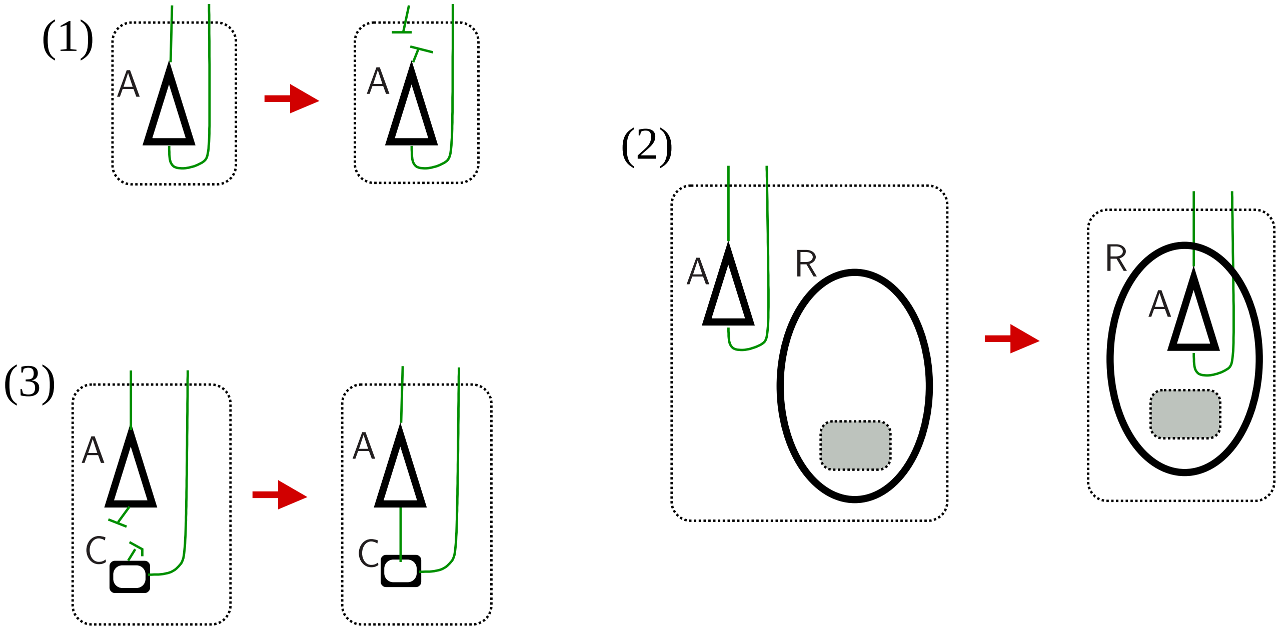
......and after **three** reactions

$H \circ G$

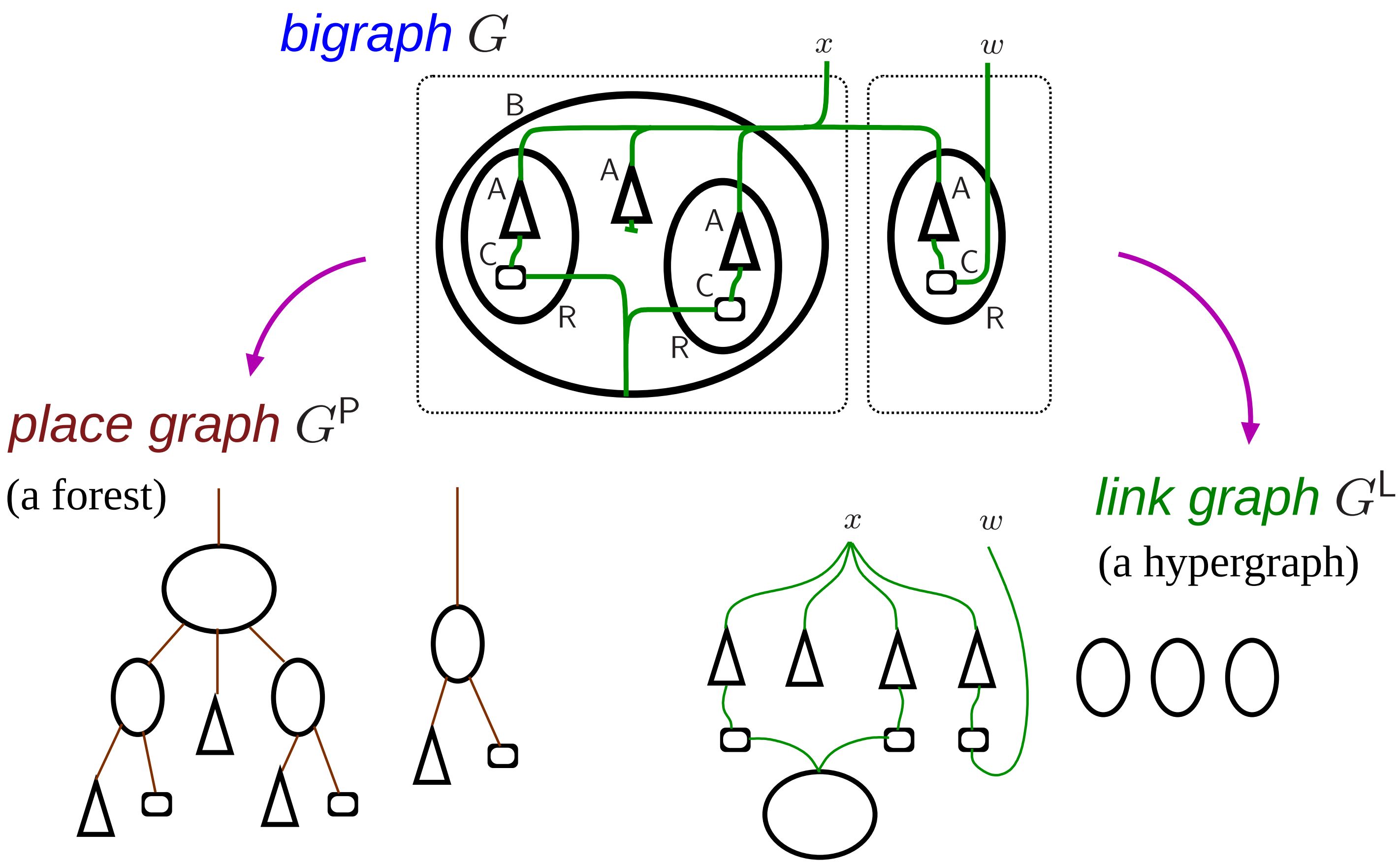




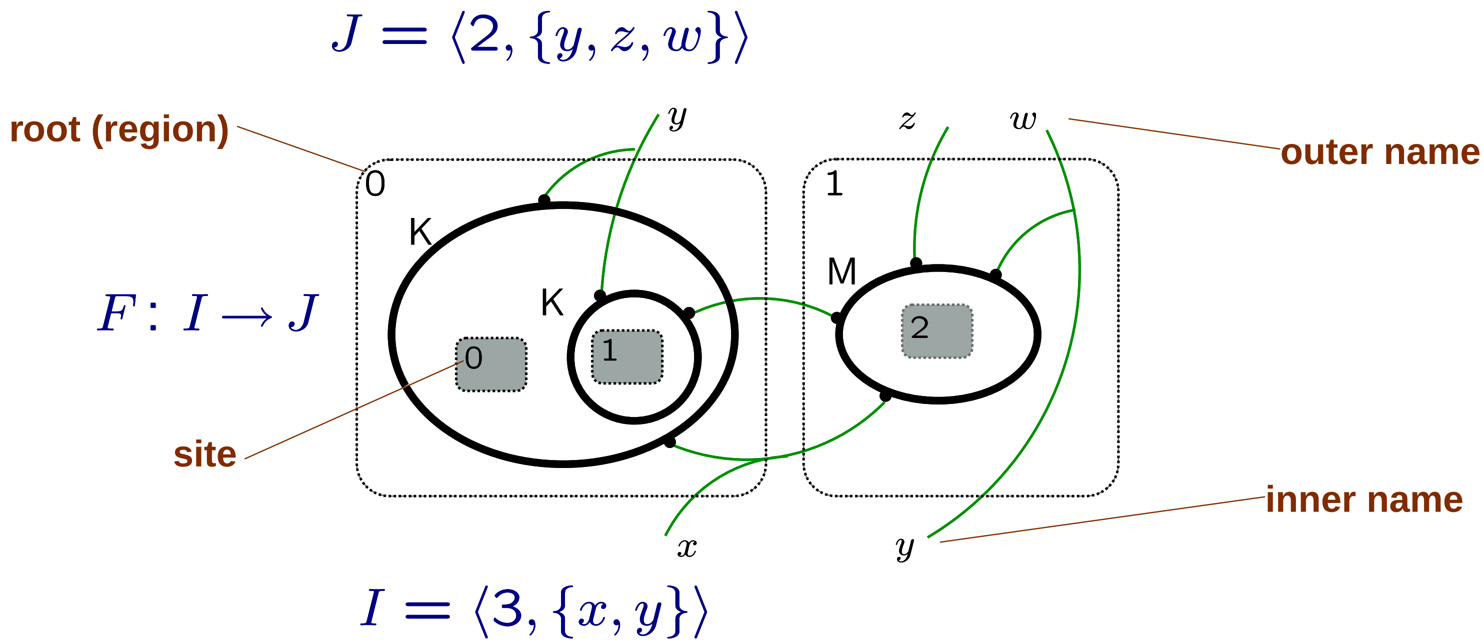
Three possible reaction rules



The ‘bi-’ structure of a bigraph



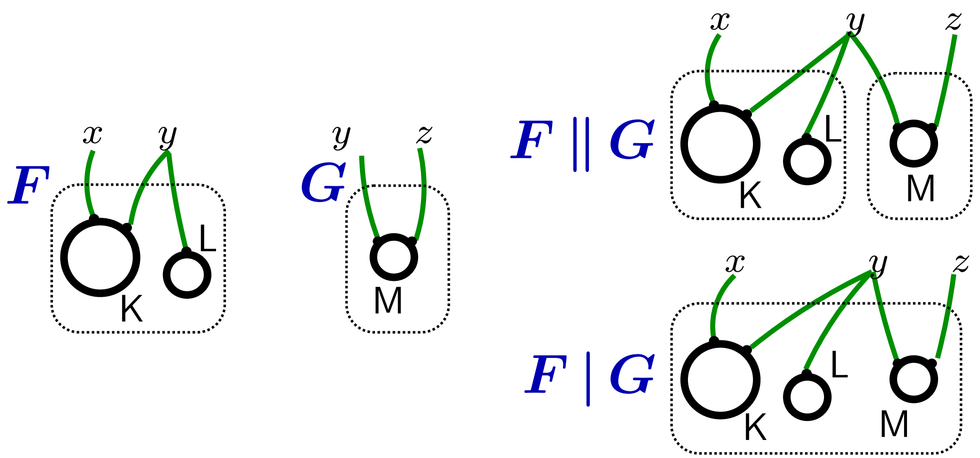
Bigraph algebra: their interfaces and operations



Composition: Place $F\colon I \to J$ inside $G\colon J \to K$
to yield $G \circ F\colon I \to K$.

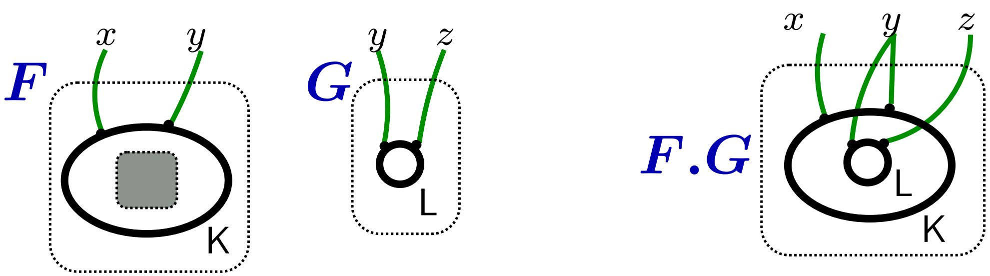
Product: Place $F\colon I \to J$ alongside $G\colon H \to K$
to yield $F \otimes G\colon I \otimes H \to J \otimes K$.

Derived operations: product and nesting



parallel product

merge product

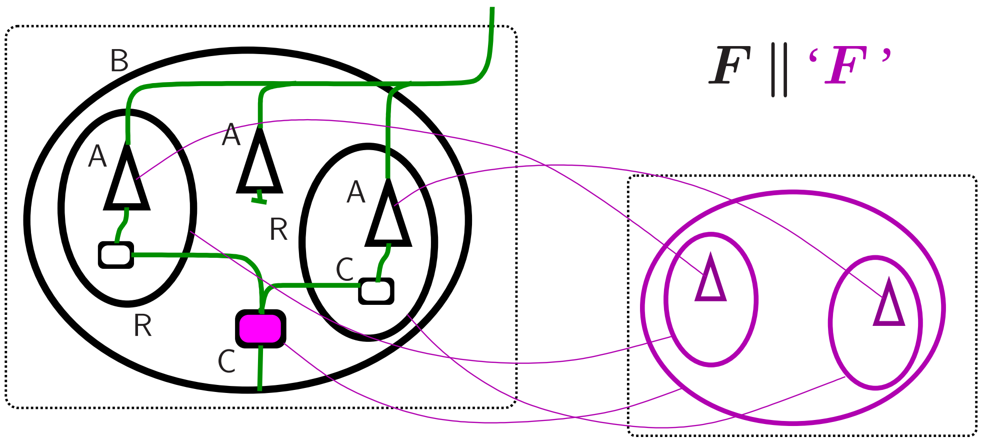


nesting

These operations are *elementary* for process calculi.
Illuminating that they are *derived* in the categorical framework.

Reflective building (2)

A building may keep a partial record of its occupancy.



So it has a central computer that ‘holds’ the record.

The record could be any data structure, accessible to the real occupants via the building’s network.

PARTS OF THE TALK

- Ubiquitous Computing, and modelling it
- Space, Motion and Bigraphs
- **Examples of Bigraphical Systems**
- Conclusion

Examples

- **Example 1: Finite CCS,** *a process calculus*
- **Example 2: Membrane budding,** *stochastic biology*
- **Example 3: The savannah game,** *a physical-virtual mix*

Finite CCS

$$\text{SYNTAX}\left\{\begin{array}{lcll} \mu & ::= & \overline{x} \;\big|\; x & \textit{actions} \\ P & ::= & A \;\big|\; \nu x P \;\big|\; P \mid P & \textit{processes} \\ A & ::= & \mathbf{0} \;\big|\; \mu.P \;\big|\; A + A & \textit{alternations} \end{array}\right.$$

The BRS for CCS has controls send, get and alt. It has one sort for processes, one for alternations.

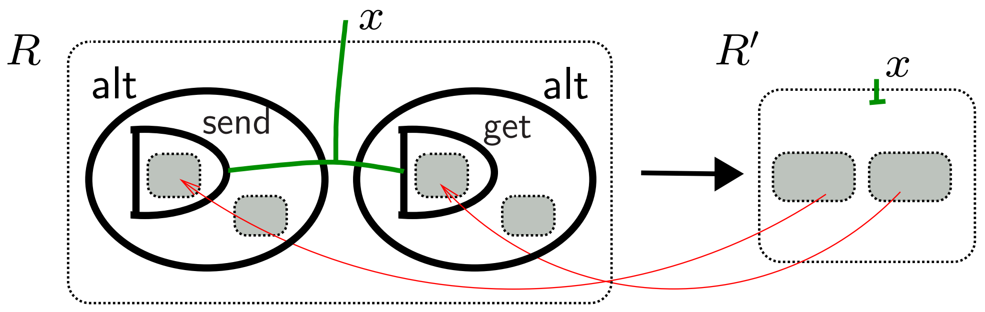
Maps $\mathcal{P}_X[\cdot]$ and $\mathcal{A}_X[\cdot]$ translate CCS entities with names $\subseteq X$ to bigraphs of the right sort:

$$\begin{array}{rcl} \mathcal{P}_X[\nu x P] & = & /x\, \mathcal{P}_{x \uplus X}[P] \\ \mathcal{P}_X[P \mid Q] & = & \mathcal{P}_X[P] \mid \mathcal{P}_X[Q] \\ \mathcal{P}_X[A] & = & \mathsf{alt}.\, \mathcal{A}_X[A]\ . \end{array} \qquad \begin{array}{rcl} \mathcal{A}_X[\mathbf{0}] & = & X \mid 1 \\ \mathcal{A}_X[\overline{x}.P] & = & \mathsf{send}_x.\mathcal{P}_X[P] \\ \mathcal{A}_X[x.P] & = & \mathsf{get}_x.\mathcal{P}_X[P] \\ \mathcal{A}_X[A+B] & = & \mathcal{A}_X[A] \mid \mathcal{A}_X[B]\ . \end{array}$$

Reaction in CCS bigraphs

Reaction in CCS: $(\overline{x}.P_1 + A_1) \mid (x.P_2 + A_2) \longrightarrow P \mid Q$

This is encoded in bigraphs by the rule:



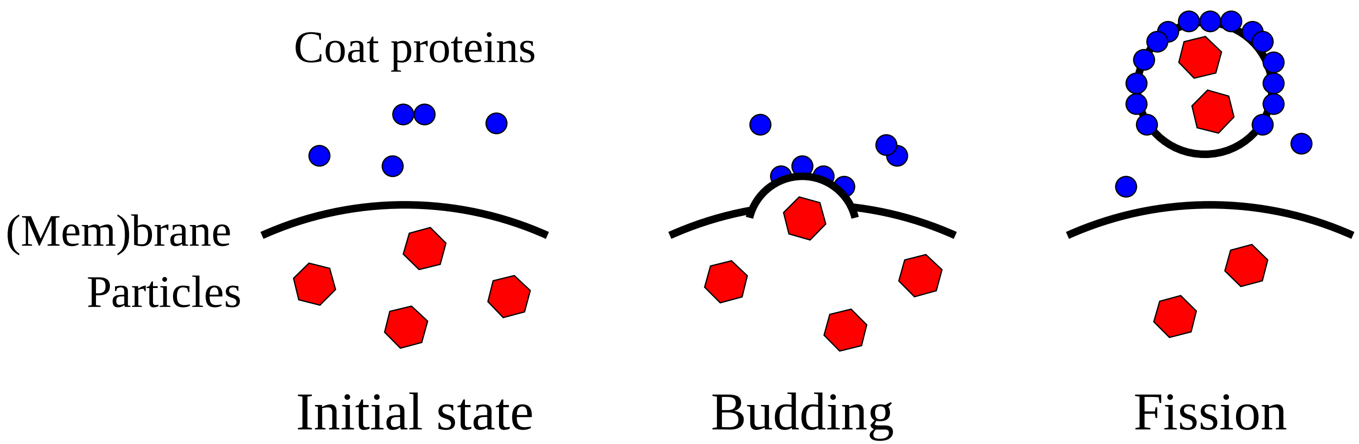
The red arrows show which parameters are retained. The rule generates a reaction relation $\longrightarrow$ between CCS bigraphs.

THEOREM The bigraph model *explains* CCS:
$P \longrightarrow P'$ in CCS iff $\mathcal{P}_X[P] \longrightarrow \mathcal{P}_X[P']$ in bigraphs.

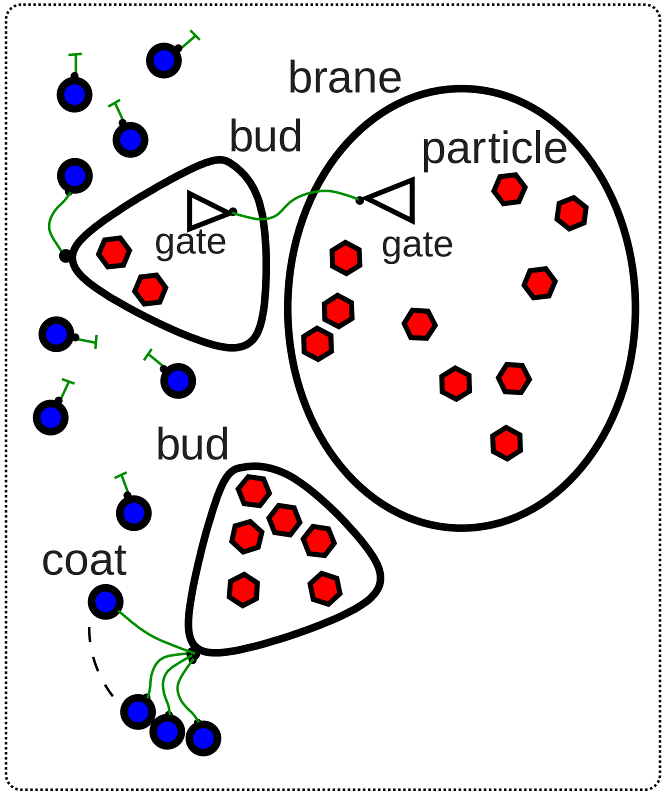
Stochastic dynamics

joint work with Jean Krivine and Angelo Troina

For example, **membrane budding**:



A membrane-bud system



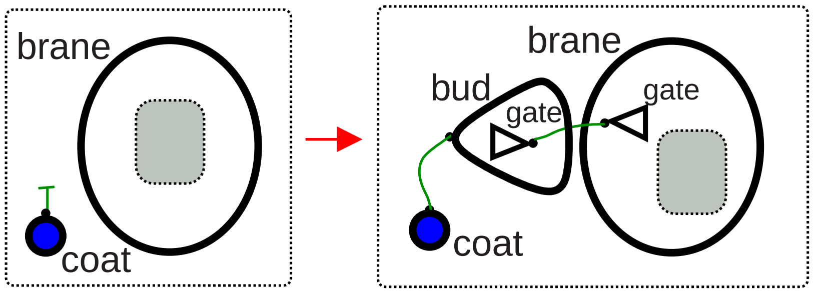
The controls are:

brane, bud, coat, particle, gate

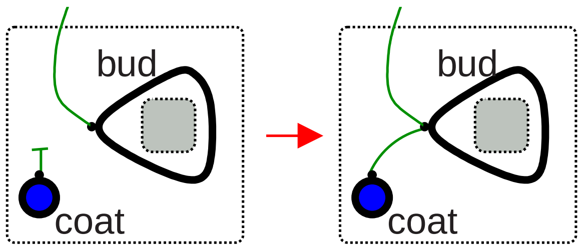
The **sorting** *dictates:*

- a particle, coat protein or gate has no children
- children of a bud or brane are particles or gates

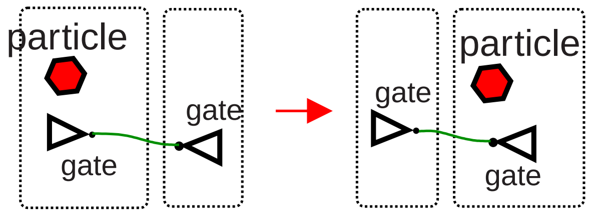
Reaction rules for budding, with stochastic rates



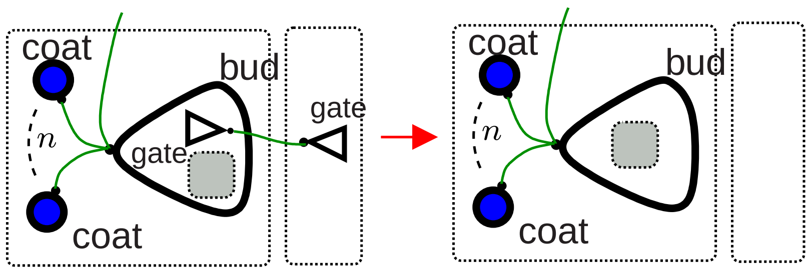
bud formation



coating



particle migration



bud fission

Stochastics: the rates of reactions

Assign a **rate** ρ_i to each reaction rule $R_i \longrightarrow R'_i$

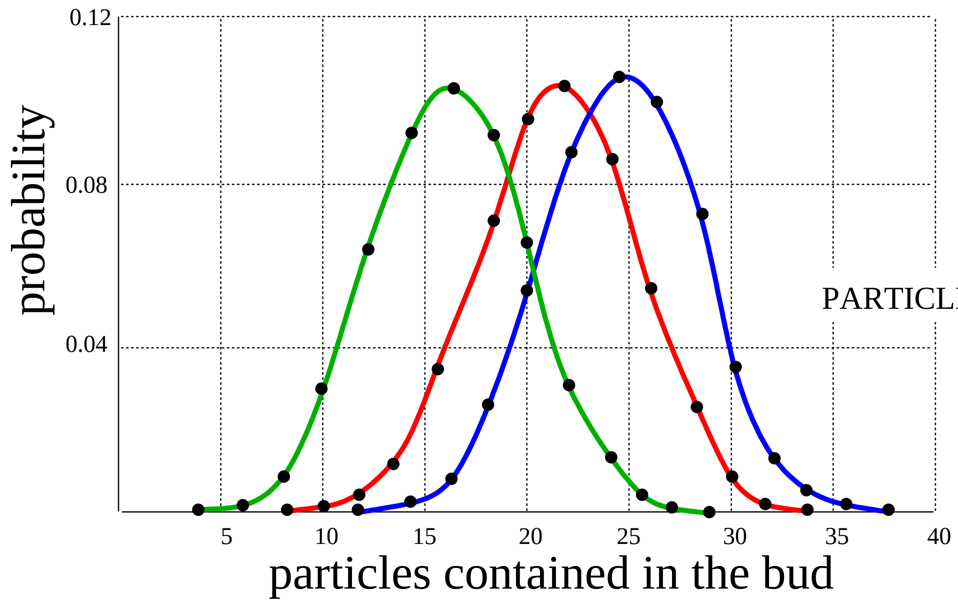
The rate of a particular **reaction** $g \longrightarrow g'$ is given by

$$\sum_i \rho_i \cdot n_i$$

where n_i is the number of different ways that the i$^{\mathrm{th}}$ rule can give rise to the reaction $g \longrightarrow g'$.

The rate of a **labelled transition** $a \xrightarrow{L} a'$ in a process calculus can be *derived* from rate of its underlying reaction.

A simulation of budding, using PRISM



RATES:

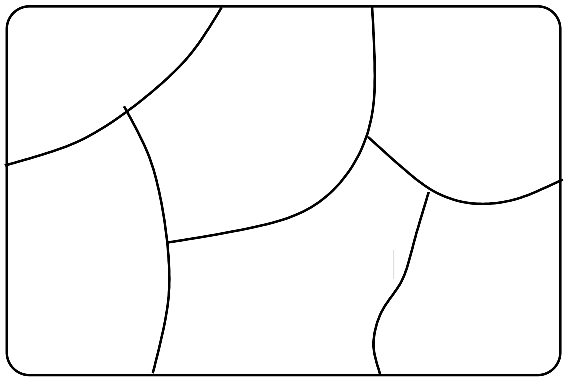
PARTICLE MIGRATION	1	1	2
COATING	2	1	1

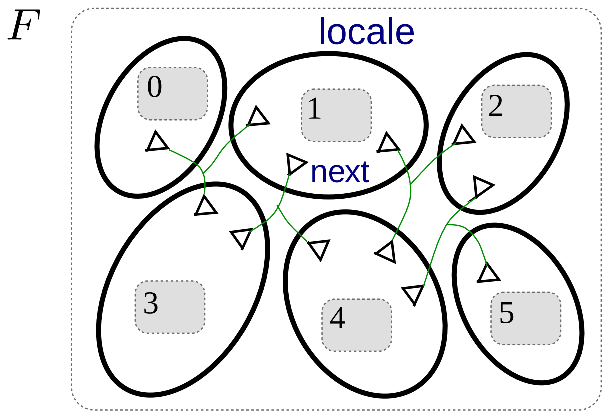
As the rate of particle migration increases, relative to the coating rate, the expected number of particles in a bud increases.

This number has a normal distribution of constant width.

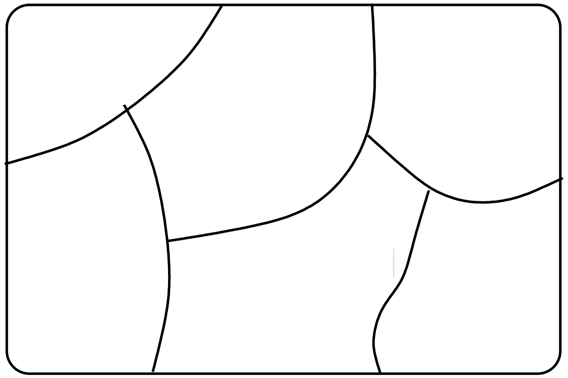
The savannah as a bigraph

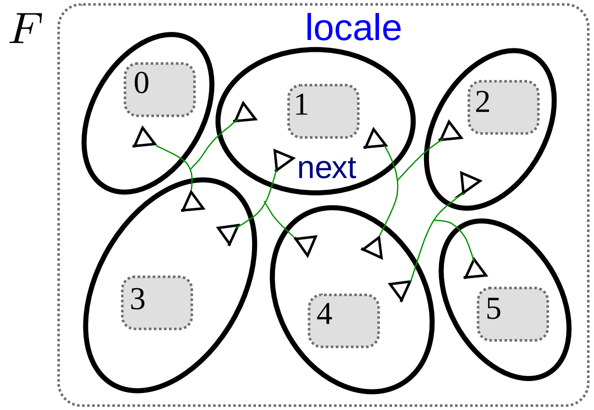
(*thanks to Benson et al.*)

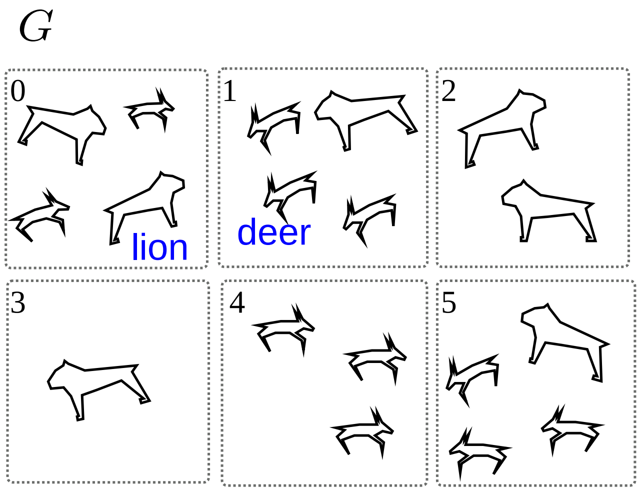


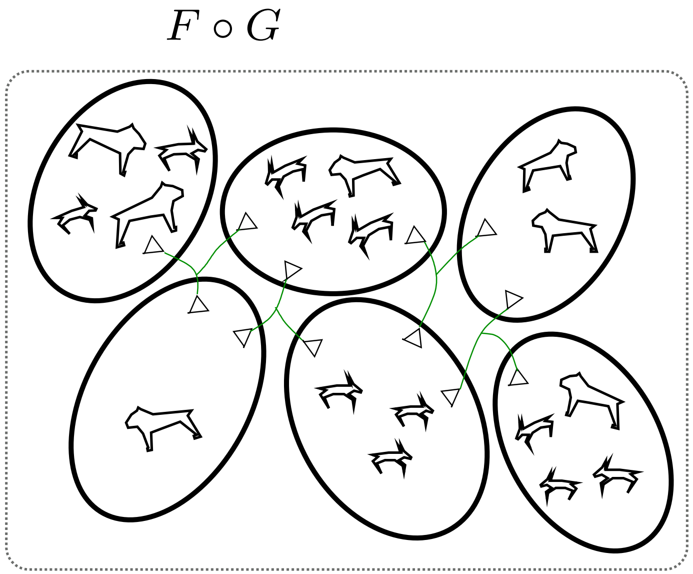


Nodes with control locale are *locales*; their contiguity is represented by linkage of next nodes.

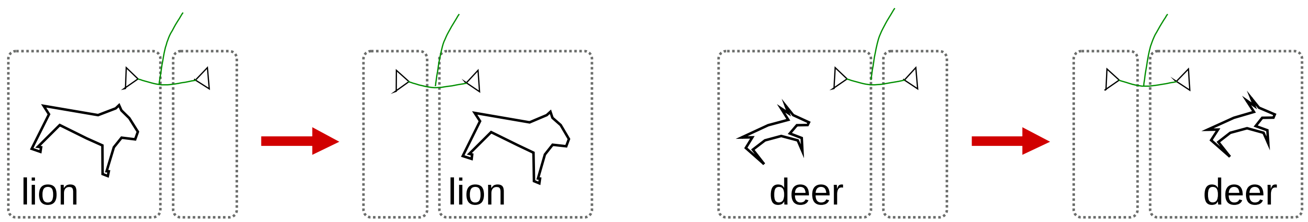



F
locale
0
1
2
3
4
5
next

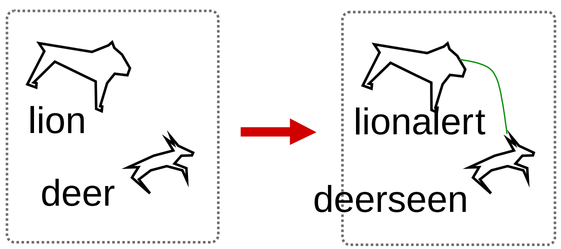

G
0
1
2
3
4
5
lion
deer


$F \circ G$

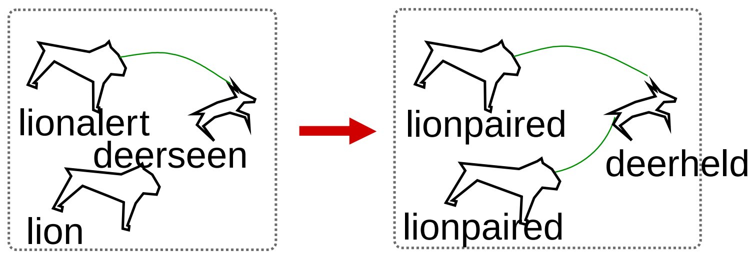
Reaction rules for animals



An animal moves to a neighbouring region

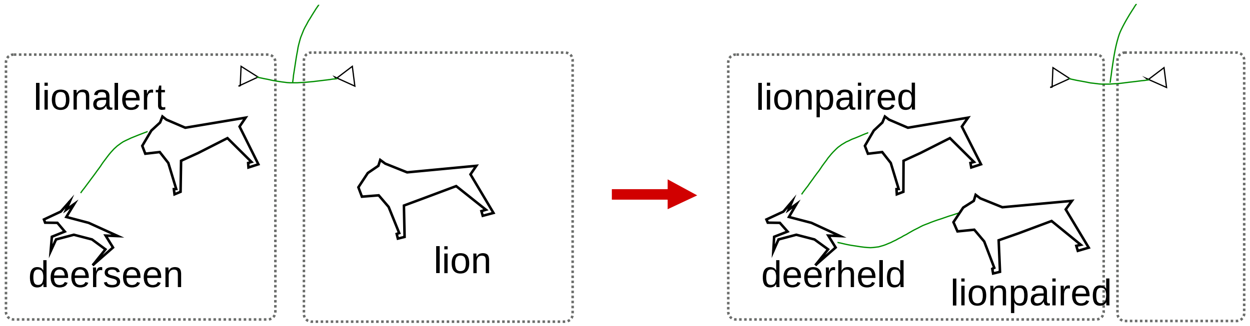


A lion becomes alert to a nearby deer

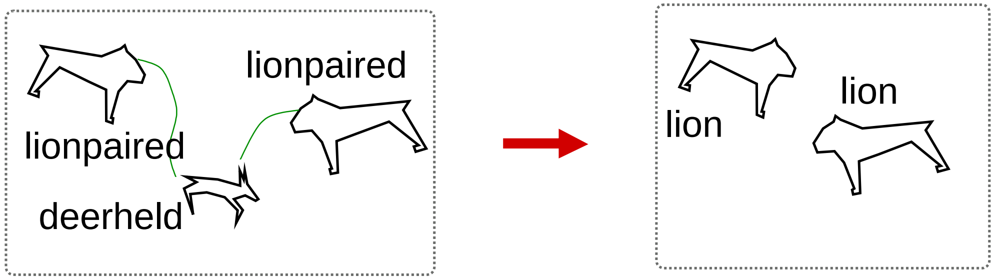


An alerted lion pairs with another nearby

More reaction rules for animals



An alerted lion can attract its pair from a neighbouring locale

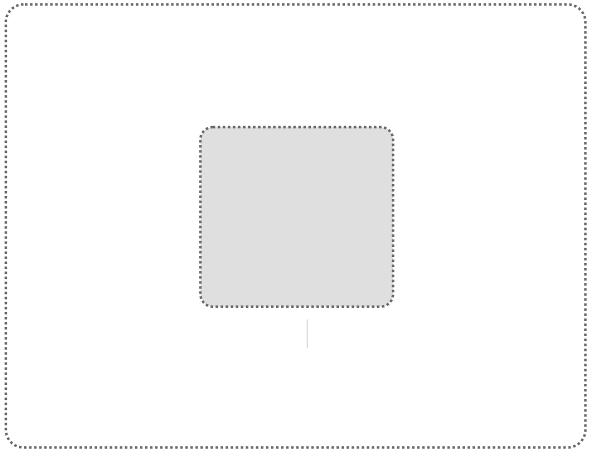


Paired lions can kill their deer

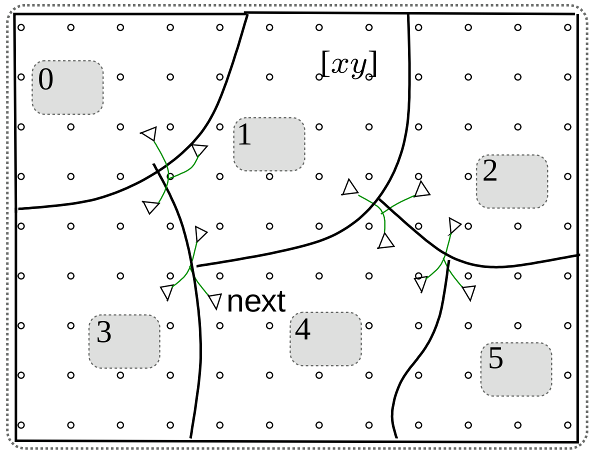
The savannah game

- **Children** roam a featureless field;
- each child has a **GPS** on her back, and carries a **laptop**.
- On the laptop is a **game server** ...
- ... which knows each child's location from her GPS.
- The game server pretends that the field is a **savannah** ...
- ... and allows the children to "be" **virtual lions** ...
- ... and to "kill" **virtual deer** on their laptops.

The field and the virtual savannah

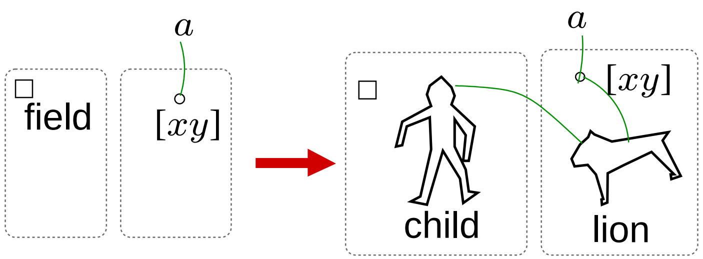


Open field to be occupied by children with GPS+laptop

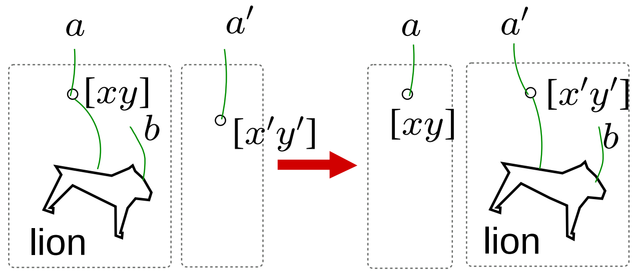


Virtual savannah with six locales and grid spots

Reaction rules for virtual animals

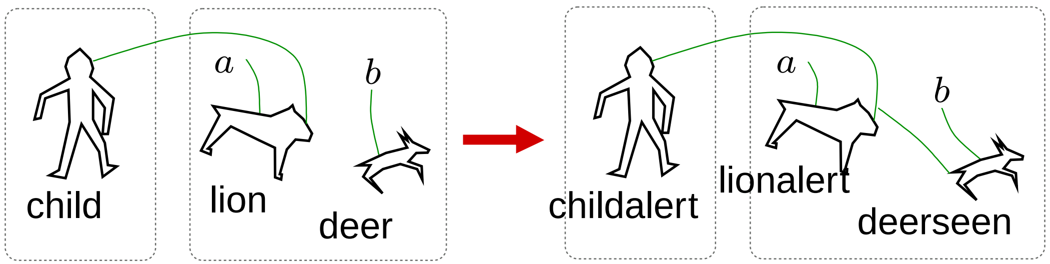


A child/lion enters the game at spot [xy]
(similarly for a deer)



$(|x - x'| + |y - y'| = 1)$

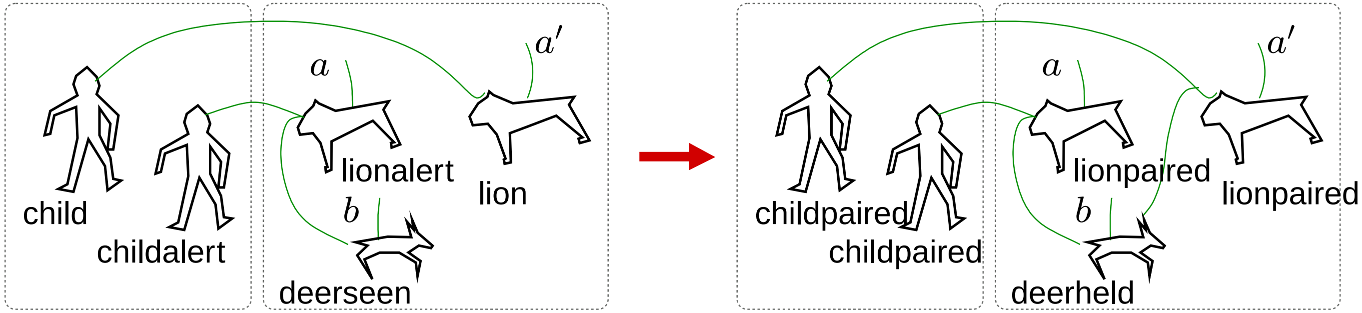
A virtual lion moves to another spot
(similarly for a deer)



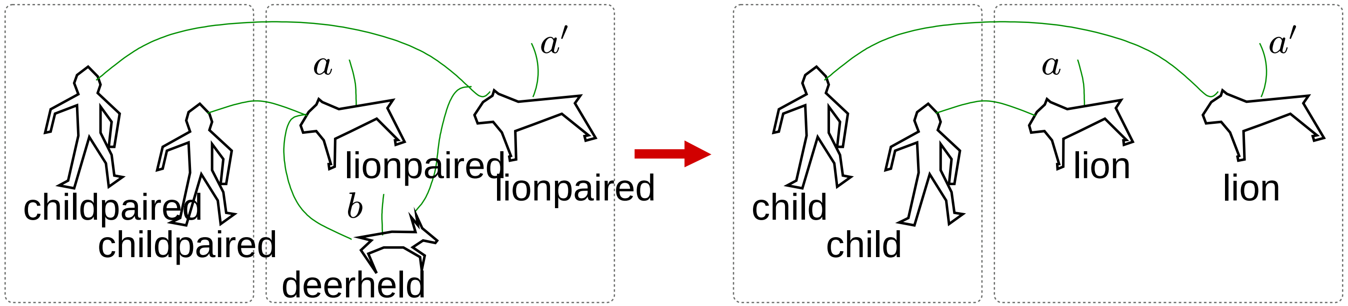
A child/lion becomes alert to a deer in its locale

$$/c\,(\mathsf{child}_c \parallel (\mathsf{lion}_{ac} \mid \mathsf{deer}_b)) \longrightarrow /c\,(\mathsf{childalert}_c \parallel (\mathsf{lionalert}_{ac} \mid \mathsf{deerseen}_{bc}))$$

More reaction rules for virtual animals



An alerted child/lion can pair with another in its locale



The pair of child/lions kill the deer

What does the savannah game illustrate?

- The interplay between **placing** and **linking**
- Modelling **physical** and **virtual** at once
- How to combine **topographical** and **metrical** space
- The interplay between a **human** and a **complex artifact**

PARTS OF THE TALK

- Ubiquitous Computing, and modelling it
- Space, Motion and Bigraphs
- Examples of Bigraphical Systems
- Conclusion

Are bigraphs enough for a Ubiquitous Abstract Machine?

- They represent a space for **reconfigurable** entities
- They admit **self-awareness**, hence self-management
- They embed many **process calculi** and recover their theory
- They can provide models for **natural science**
- They yield a **programming** language