

Bigraphical Programming Languages for Pervasive Computing

L. Birkedal¹ M. Bundgaard¹ T. C. Damgaard¹
S. Debois¹ E. Elsborg¹ A. J. Glenstrup¹
T. Hildebrandt¹ R. Milner² H. Niss¹

¹The BPL Project
IT University of Copenhagen, Denmark

²Computer Laboratory
University of Cambridge, UK

Combining Theory and Systems Building in Pervasive Computing
PERVASIVE 2006

Outline

- 1 Background
 - The UKCRC Ubiquitous Computing Grand Challenge
 - Bigraphical Systems
- 2 BPL Project Activities
 - Context Awareness
 - Pervasive Business Processes and Distributed Reactive XML
 - Matching and Axiomatisation(4)
 - Higher-order Mobile Embedded Resources

Outline

- 1 Background
 - The UKCRC Ubiquitous Computing Grand Challenge
 - Bigraphical Systems
- 2 BPL Project Activities
 - Context Awareness
 - Pervasive Business Processes and Distributed Reactive XML
 - Matching and Axiomatisation(4)
 - Higher-order Mobile Embedded Resources

Challenging Engineering and Theoretical Perspectives.

- The UKCRC Ubiquitous Computing Grand Challenge includes an engineering and a theoretical perspective:

Engineering: ... define system design principles that pertain to all aspects of ubiquitous computing. . .

Theoretical: ... develop science whose concepts, calculi, models, theories and tools allow descriptive, explanatory and predictive analysis of ubiquitous computing at many levels of abstraction; to employ these analyses to derive all its systems and software, including languages.

BPL Project Studies Bigraphical Reactive Systems.

The aim of the BPL Project is to contribute to this challenge by researching Bigraphical Reactive Systems (BRS):

- BRS is a general graphical semantical framework for reactive systems
- BRS unifies existing calculi for concurrency and mobility:
 π Calculus, Mobile Ambients, Petri Nets, . . .
- BRS focus on mobile locality and connectivity

We believe that the BRS framework can serve as a foundation for combining the theoretical with the engineering aspect of the Grand Challenge.

Previous Work on Bigraphs.

Previous work on bigraphs include

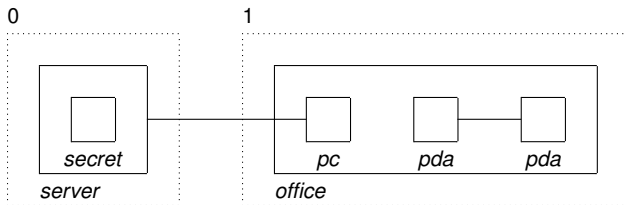
- Ole Høgh Jensen & Robin Milner, *Bigraphs and Mobile Processes (revised)*, Technical Report UCAM-CL-TR-580, University of Cambridge, Computer Laboratory, 2004.
- Robin Milner, *Bigraphs Whose Names have Multiple Locality*, Technical Report UCAM-CL-TR-603, University of Cambridge, Computer Laboratory, 2004.

Outline

- 1 Background
 - The UKCRC Ubiquitous Computing Grand Challenge
 - **Bigraphical Systems**
- 2 BPL Project Activities
 - Context Awareness
 - Pervasive Business Processes and Distributed Reactive XML
 - Matching and Axiomatisation(4)
 - Higher-order Mobile Embedded Resources

A Bigraph Consists of a Place and a Link Graph.

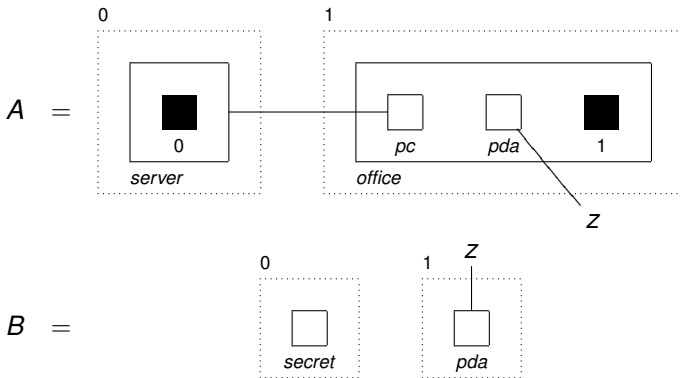
- Bigraphs consist of a place graph and a link graph with common nodes.
- The place graph is a tree, often illustrated by node nesting:



- Roots, drawn with dotted lines, are indexed.

Bigraphs are Composable.

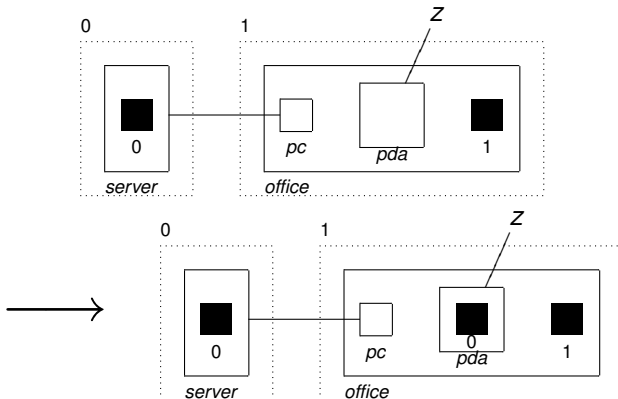
- Composing $A \circ B$, we obtain the preceding bigraph:



- Sites of A —drawn as filled boxes—are indexed, and mark the places where the roots of B are “plugged in.”

Parametric Reaction Rules Specify System Dynamics.

- Example parametric rule: PDA beside server-connected PC may copy server data:



Outline

- 1 Background
 - The UKCRC Ubiquitous Computing Grand Challenge
 - Bigraphical Systems
- 2 **BPL Project Activities**
 - **Context Awareness**
 - Pervasive Business Processes and Distributed Reactive XML
 - Matching and Axiomatisation(4)
 - Higher-order Mobile Embedded Resources

Modelling Context and Agents is Tricky.

- BRS can conveniently model location and movement of software and hardware agents.
- However, the model must be chosen carefully: The naive approach of modelling context and agents together suffers from several problems:
 - Searching the place graph in a recursive manner requires in essence modelling a program counter
 - This in turn may prevent concurrency
 - It is hard to make rules depending on the *absence* of some entity.

Platographical Models Separate Context and Agent.

- To avoid these problems, we base our BRS on **platographical models**.
- A platographical model is a triple **(C, P, A)** of BRS:
 - **C**, a context model,
 - **P**, a proxy modelling a model of the context,
 - **A**, the agent,such that $\mathcal{M} = \mathbf{C} \cup \mathbf{P} \cup \mathbf{A}$ is itself a BRS, and **A** and **C** cannot directly interact.
- Thus the agent can only observe and manipulate the context by using the proxy as sensor or actuator.
- This matches well the real world: Pervasive systems are based on sensors and actuators!

Sortings of Bigraphs Prevent Mixing Up Model Parts.

- As a platographical model is *one* BRS, reaction rules in this BRS manipulate each of the constituent models' elements.
- When operating on the platographical model, we want to ensure that the context, proxy and agent parts are not inadvertently mixed during composition.
- To this end we use **sorted platographical models**, in which bigraphs from each part of the model are assigned different sorts.
- Allowing composition only when the sorts of sites and roots match, we can ensure that model parts are not mixed.

Outline

- 1 Background
 - The UKCRC Ubiquitous Computing Grand Challenge
 - Bigraphical Systems
- 2 **BPL Project Activities**
 - Context Awareness
 - **Pervasive Business Processes and Distributed Reactive XML**
 - Matching and Axiomatisation(4)
 - Higher-order Mobile Embedded Resources

Business Processes can be Modelled in BRS.

- We describe a general approach to define an XML-based exchange and execution format for business process and coordination languages.
- Further, we describe an XML-based format for process language semantics based on BRS.
- Similarities between bigraphs and XML are exploited.
- This allows for implementing these BRS using standard XML tools (XPath, XQuery, etc.) to implement reaction rules.
- Using this technique, a case study led us to suggest extending BRS with **higher-order reaction rules**, as well as a kind of **context dependent reaction rules**.

Concurrent BRS is Achieved Using XML-Store.

- The BRS is implemented on XML-store, a peer-to-peer concurrent storage framework with optimistic concurrency control.
- This has directly allowed concurrency in the BRS, effectively making it a pervasive system.
- This **Distributed Reactive XML** system is an ideal scenario for a coordination middleware.

Outline

- 1 Background
 - The UKCRC Ubiquitous Computing Grand Challenge
 - Bigraphical Systems
- 2 **BPL Project Activities**
 - Context Awareness
 - Pervasive Business Processes and Distributed Reactive XML
 - **Matching and Axiomatisation(4)**
 - Higher-order Mobile Embedded Resources

Creating a BPL Platform is Non-Trivial.

- An important goal is to create a BPL platform for experimenting with BRS.
- This platform will allow application of reaction rules on agents (bigraphs).
- Two main issues in designing a BPL platform are:
 - How to represent bigraphs to allow construction, deconstruction, equality test, search, etc.
 - How to implement system reaction.

Axiomatisation Provides Bigraph Normal Forms.

- We have defined a semantic and syntactic **Binding Discrete Normal Form** (BDNF) for bigraphs:

$$\begin{aligned} B & ::= (\omega \otimes \bigotimes_{i < n} (\vec{y}_i) / (\vec{X}_i)) D \\ D & ::= \alpha \otimes (P_0 \otimes \cdots \otimes P_{n-1}) \pi \\ P & ::= (X) (\text{id}_Y \otimes \text{merge}_{n+k}) \\ & \quad (\ulcorner \alpha_0 \urcorner \otimes \cdots \otimes \ulcorner \alpha_{n-1} \urcorner \otimes M_0 \otimes \cdots \otimes M_{k-1}) \pi \\ M & ::= (K_{\vec{y}(\vec{X})} \otimes \text{id}_Z) P. \end{aligned}$$

- Identical bigraphs have identical BDNFs, modulo certain permutations.
- An accompanying set of **axioms** are proven sound and complete, and allow us to manipulate bigraphs as **syntactic expressions**.

Redex Matching is Inductively Defined.

- Matching is defined abstractly by $C \circ R \circ d = A$, with context C , redex R , parameter d , and agent A .
- Problem: given A , how do we find C, R, d ?
- To make it operational, we have constructed an equivalent, **inductive definition**:

$$\begin{array}{c}
 \text{CLOSE} \frac{\{\vec{y}_i\} = Y \quad \{\vec{z}_i\} = Z \quad \sigma_R^Z : U \rightarrow Z}{(\otimes_i^m y_i / X_i) \otimes (\otimes_i z_i / X_{i+m}) \otimes \sigma_a, (\otimes_i^m y_i / Y_i) \otimes \text{id}_U \otimes \sigma_R^C, \quad (\otimes_i z_i / Z_i) \otimes \sigma_R^Z \otimes \sigma_C \otimes \text{id}_Y \vdash a, R \hookrightarrow C, d} \\
 \text{PERM} \frac{\omega_a, \omega_R, \omega_C \vdash a, \otimes_i^m P_{\pi(i)} \hookrightarrow C, (\bar{\pi} \otimes \text{id})d}{\omega_a, \omega_R, \omega_C \vdash a, \otimes_i^m P_i \hookrightarrow C\pi, d} \\
 \text{PAR} \frac{\omega_a, \omega_R, \omega_C \parallel \omega \vdash a, R \hookrightarrow C, d \quad \omega_b, \omega_S, \omega_D \parallel \omega \vdash b, S \hookrightarrow D, e}{\omega_a \parallel \omega_b, \omega_R \parallel \omega_S, \omega_C \parallel \omega_D \parallel \omega \vdash a \otimes b, R \otimes S \hookrightarrow C \otimes D, d \otimes e} \\
 \text{LOCAL-SUB} \frac{\sigma_a \otimes \omega_a, \omega_R, \sigma_C \otimes \omega_C \vdash p, R \hookrightarrow P, d \quad \sigma_a : Z \rightarrow \sigma_C : U \rightarrow}{\omega_a, \omega_R, \omega_C \vdash (\widehat{\sigma}_a \otimes \text{id})(Z)p, R \hookrightarrow (\widehat{\sigma}_C \otimes \text{id})(U)P, d} \\
 \text{MERGE} \frac{\omega_a, \omega_R, \omega_C \vdash a, R \hookrightarrow C, d}{\omega_a, \omega_R, \omega_C \vdash (\text{merge} \otimes \text{id})a, R \hookrightarrow (\text{merge} \otimes \text{id})C, d} \\
 \text{ION} \frac{\omega_a, \omega_R, \omega_C \vdash (\otimes_i^n (v_i) / (X_i) \otimes \text{id})p, R \hookrightarrow (\otimes_i^n (v_i) / (Z_i) \otimes \text{id})P, d \quad \alpha = \vec{y} / \vec{u} \quad \sigma_y : \{\vec{y}\} \rightarrow}{\sigma_y \parallel \omega_a, \omega_R, \sigma_y \alpha \parallel \omega_C \vdash (K_{\vec{y}(X)} \otimes \text{id})p, R \hookrightarrow (K_{\vec{u}(Z)} \otimes \text{id})P, d} \\
 \text{SWITCH} \frac{\omega_a, \text{id}_\epsilon, \omega_C(\sigma_R \otimes \omega_R \otimes \text{id}) \vdash p, \text{id} \hookrightarrow P, d \quad \sigma_R : W \rightarrow}{\omega_a, \omega_R, \omega_C \vdash p, (\widehat{\sigma}_R \otimes \text{id})(W)P \hookrightarrow \ulcorner U \urcorner, d} \\
 \text{PRIME-AXIOM} \frac{}{\omega, \text{id}_\epsilon, \omega(\alpha^{-1} \otimes \text{id}) \vdash p, \text{id} \hookrightarrow \ulcorner \alpha \urcorner, p} \\
 \text{WIRING-AXIOM} \frac{}{y, X_R / \emptyset, y / (X_R \uplus X_D) \vdash \text{id}_\epsilon, \text{id}_\epsilon \hookrightarrow \text{id}_\epsilon, X_D / \emptyset}
 \end{array}$$

Outline

- 1 Background
 - The UKCRC Ubiquitous Computing Grand Challenge
 - Bigraphical Systems
- 2 **BPL Project Activities**
 - Context Awareness
 - Pervasive Business Processes and Distributed Reactive XML
 - Matching and Axiomatisation(4)
 - **Higher-order Mobile Embedded Resources**

HOMER Models Running Process Mobility.

- Higher-order Mobile Embedded Resources (HOMER) allows modelling **mobility of running processes**, **nested locations**, and **local names**:

Processes:

p, q, r	$::=$	$\mathbf{0}$	inactive process
		$\lambda.p$	action prefixing
		$p \parallel q$	parallel composition
		$(n)p$	let n be local in p
		x	process variable
		$p[x := q]$	p in the explicit substitution of q for x

Prefixes:

λ	$::=$	$\delta(x)$	receive a resource at δ and bind it to x
		$\bar{\delta}(x)$	take computing resource from δ and bind it to x
		$\bar{\delta}\langle r \rangle$	send a resource r to δ
		$\delta[r : \tilde{n}]$	computing resource r at δ having type \tilde{n}

HOMER can be Modelled in BRS.

- We have given a translation of HOMER terms and semantics into BRS.
- The bigraph semantics are based on Milner's presentation of the λ -calculus in local bigraphs.
- The translation required a new definition of parametric reaction rules and a representation of the location of names.

Summary

- A **bigraphical reactive system** is a powerful framework for constructing reliable pervasive systems, based on solid theory.
- We have addressed
 - **context awareness**
 - **business processes and XML**
 - **axiomatisation and matching**
 - **higher-order mobile embedded resources.**
- ...more to follow, also a **bigraphical programming language.**

BPL Web page: <http://www.itu.dk/research/theory/bpl/>