Mikado

Mobile calculi based on domains

J.B Stefani -- INRIA

Outline

- Objectives
- Domains
- Workprogramme
- Consortium and figures
- Challenges

Objectives

- To develop formal programming models for large scale, mobile distributed computing
  - In Mikado, “programming model” = “process calculus”
  - In Mikado, programming models to be based on a (hopefully) common “core domain model”
- To develop associated type systems and proof techniques
  - Safety, security
- To develop effective prototypes
  - Abstract machines
  - Language experiments

Domains

- Key insight for Mikado programming models : domains
- Large scale distributed systems take the form of multiple interacting areas or regions, with different properties
  - e.g. spatial location and extent, communications, fault models, security policies, resource management policies
- A domain is a first-class programming entity introduced to capture the notion of region or sub-system
- Insights for domains :
  - Ambients, locations in distributed process calculi
  - Domains, regions and autonomous systems in network architecture

Work-programme

- WP1 Programming Model :
  - to develop the Mikado “core domain model” and domain-based programming models; to study their relationships with other programming paradigms (functional, object-oriented)
- WP2 Specification and Analysis:
  - to develop type systems and (co-inductive) proof techniques for the Mikado programming models
- WP3 Virtual Machine Technology & Language Support:
  - to develop virtual machine and programming language technology based on the Mikado programming model
- WP4 : management, dissemination and evaluation

Recent results

- Programming models
  - a calculus with programmable domains : the M-calculus
  - process mobility in Ambients : M^3
  - dynamic node connectivity in Klaim
  - lexical scoping in a distributed π : lsdπ
  - calculi with resource access control: Controlled Ambients and Mobile Resources
- Specification and Analysis
  - dynamic acquisition of resource privileges in Klaim
  - co-inductive characterization of barbed congruence for higher-order π
  - linear types for optimizing code generation in TyCO

Recent results

- Virtual Machines and Languages
  - Klava and XKlaim
  - Object-oriented XKlaim and MOMI (objects and mixins in a distributed setting)
  - Multithreaded distributed abstract machine for DiTyCO
  - Distributed abstract machine for the M-calculus
  - Common software infrastructure for distributed abstract machines

Consortium and figures

- Mikado Partners :
  - INRIA (prime - F) : G. Boudol, JB Stefani
  - France Telecom R&D (F) : M. Lacoste
  - University of Florence (I) : R. de Nicola
  - University of Sussex (GB) : M. Hennessy
  - University of Lisbon (P) : V. Vasconcelos
- Duration : 36 months
- Manpower : 434 mm (360 mm funded)
- Funding : 1.9 M€

Challenges

- To turn the multi-faceted, informal notion of ‘domain’ into a “theory of domains” and effective programming model abstractions
- To show that Mikado’s approach can subsume other recent distributed process calculi and meet key technical requirements for programming global computing systems
- To turn Mikado’s theoretical results into effective (i.e. efficient, practical) programming language technology

For more information

http://mikado.di.fc.ul.pt/