Mikado
Mobile calculi based on domains

J.B Stefani
INRIA Rhône-Alpes
Plan

- Objectives
- Domains
- Workprogramme
- Milestones
- Consortium and figures
- Challenges and risks
- Dissemination
Objectives

- To develop a formal programming model for large scale, mobile distributed computing
  - In Mikado, ‘programming model’ = ‘(higher-order) process calculus’
- To develop associated type systems and proof techniques
  - Safety, security
- To develop effective prototypes
  - Virtual machines
  - Language experiments
Domains

- Key insight for the Mikado programming model: domains
- Large scale distributed systems take the form of multiple interacting areas or regions, with different properties
  - e.g. spatial location and extent, 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: to develop the Mikado programming model and study its relation with other programming paradigms (functional, object-oriented)
- WP2: to develop type systems and (co-inductive) proof techniques for the Mikado programming model
- WP3: to develop virtual machine and programming language technology based on the Mikado programming model
- WP4: management, dissemination and evaluation
Milestones

- Year 1: key requirements and analysis for the Mikado programming model
- Year 2: core programming model and type systems, early prototypes
- Year 3: combination with other programming paradigms, final prototypes
Consortium and figures

- Mikado Partners:
  - INRIA (prime - F) : G. Boudol, JB Stefani
  - France Telecom R&D (F) : K. Milsted
  - 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 requested : 1.9 M€
Challenges and risks

- To turn the multi-faceted, informal notion of ‘domain’ into 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
Dissemination

- Publications
- Workshops
- Web site
- Open source
  - Inclusion in the ObjectWeb code base for the more mature prototypes (see: http://www.objectweb.org)