INRIA internship experience

Michal Malohlava

DISTRIBUTED SYSTEMS RESEARCH GROUP
http://dsrg.mff.cuni.cz/

CHARLES UNIVERSITY IN PRAGUE
Faculty of Mathematics and Physics
Where did I work?

- Lille, France
  - Nord-Pas-de-Calais
  - Strategic location: 1h to Paris, 30min to Brussels, 1.5h to London

- Basic info
  - Agglomeration
    - 1.34mil people
  - Student city

- Villeneuve d'Ascq
  - Campus, LIFL University
  - Research
ADAM team overview

• ADAM project team – part of INRIA
  – Public research, but managed like private company
    • HR, PR departments
  – http://adam.lille.inria.fr/

• Structure of the team
  – Professors, phd students, engineers

• Publication driven
  – Conferences & journals ranking
    • Computing Research and Education Association of Australasia (http://www.core.edu.au/)
ADAM team projects

- Projects involve components (Fractal-based:), deployment, services, models, source code generation, verification:
  - F4E = Fractal for Eclipse (Yann Davin)
    - Fractal application development in Eclipse
      - 1 year full time work
  - Frascati
    - WS
  - Calico (G. Waigner, Anne-Francoise Le Meur)
    - General component based framework
      - Mapping to Fractal, CCM,
    - Verification
  - FDF – deployment (Nicolas Dolet)
  - Soleil (Ales Plsek, Frederic Loiret)
ADAM seminar

- Project presentations
- Presentation for PhD students
  - How to do good research
  - How to write paper
- Vision of whole team work
  - Integration of all the project into a scenario
Living & working in France

- Full time research
- Team social events
- Food
- Kind & open people
  - “Only in the north of France”
- Easy administration
- ... 
- It's France :-)

- French language
- Weather
  - No sun, no summer
- Summer is lazy time in France
Scope of my work

- Part of a sub-team
  - Ales Plsek, Frederic Loiret, Lionel Senturier
    Philippe Merle
  - Interested in component frameworks (based on Fractal)
    - Real-time RTSJ-based component framework (Ales)
      - Soleil
    - Infrastructure of component systems (Frederic)
      - Hulotte
Scope of my work

• Distribution in RTSJ-based systems
  – State-of-the-art
    • Different types of middleware (RT RMI, RTZen)
  – Examples, benchmarks, publication

• Runtime infrastructure generation
  – General infrastructure
    • Component based membranes, bindings,...
  – Scoped to RTSJ-based systems
High-level view
Real-Time Specification for Java

• JSR-1
  – Memory allocation strategies
  – New schedulable entities
    • RT threads, events, timers
  – Determinism in JVM

• Distributed RTSJ
  – JSR-50
    • Status of draft
Soleil

- **Real-time RTSJ-based component framework**
  - How to design component based application in the scope of RTSJ.
  - How to reflect design in a runtime infrastructure
    - Automatic generation of the infrastructure
      - Fractal – Juliac based

- **Main idea**
  - RTSJ based view of component applications
    - Non-functional components specifying memory/threads
Distribution in the scope of Soleil

- Distribution domains
- Adopted a connector concept of TB
  - Adapted to Soleil concept, simplified
    - Component based binding representation
  - Many (technical) restriction in generation of a runtime infrastructure
● Soleil is RT-specific tool
  – But idea of non-func. domains sounds interesting
● Hulotte generalized idea of application non-functional views
  – Distribution, rt-specific domains, ...
● Specify general concept of generation of the runtime infrastructure
  – Everything is a Fractal component
    ● Control membranes
    ● Binding representation
  – Step-by-step refinement of designed application
Hulotte – generation process

- “From the functional view of application to its runtime infrastructure”
- In fact small model-transformation
Work output

- Workshop paper
  - “Introduction of distribution into a RTSJ-based Component Framework”
    - Submitted to JRWRTC 2008

- Paper in progress
  - About Hullote
SOFA related work

- Connectors for
  - Messaging
  - Streaming

- Articles with TB, PH:
  - rewrite+resubmit: “Using a software product-line for creating component systems”
  - Memics paper: “Language for reconfiguring runtime infrastructure of component-based systems”
Q & A