UNIVERSITY OF KENT
AT CANTERBURY

Agent Dependability in Open Systems

Moderator:

- *Rogério de Lemos (University of Kent, UK)*

Panellists:

- *Gul Agha (University of Illinois - USA)*
- *Gruia Catalin-Roman (Washington University - USA)*
- *Holger Giese (University of Paderborn - Germany)*

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Panel Topic

Agent Dependability in Open Systems

- *Agent* is an *autonomous*, *adaptive* and interactive element that has a *mental state*;
- *Dependability* is the ability of a computing system to deliver its service that can justifiably be trusted.
  - threats, attributes, and means (or technologies);
- *Open systems,* such as the Internet, create conditions where systems can interact and collaborate with one another;

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Panel Topic

Last year's Panel was on "agents and dependability", with emphasis on agents:

- Features that might not so useful for enabling dependability:
  - Autonomy (failure assumptions), introspection (mental state), etc.
- Role to be played by agents;
  - building blocks, additional layer, etc.
- Restrictions to be imposed on agents;
  - failure assumptions, distributed consensus, etc.

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Panel Topic

This year's Panel has emphasis on "agent communities and dependability":

- Collection of agents, communication, and coordination, etc.

- Focus on "open systems" rather than architectures:
  - Open Architecture Agent (OAA), Jade, Zeus, etc.;

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Multi-Agent Communities

Agent communities in open systems:

- *Heterogeneous*:
  - created by different people with different intents at different times, using different languages;
- *Autonomous*:
  - own goals, and own thread of control;
- *Interactions and interoperability*:
  - join/leave at any time, interact with anyone, and perform any action;
- *Dynamic composition and coordination*:
  - contract creation, and business relationships;

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Multi-Agent Communities

Five problematic areas in multi-agent communities:

- *Autonomy*:
  - how to use, control and manage it?
- *Communication*:
  - how to ensure interoperability?
    - standard protocols might restrict agent autonomy;
- *Coordination*:
  - how to ensure coherent actions?
- *Knowledge*:
  - how to enable automatic and interactive discovery of requirements and instructions?
- *Dependability (*usually considered as an afterthought!*)*:
  - how to ensure trust on the services delivered?

Dependability Technologies

- Rigorous designs (fault prevention):
  - semantic Web and Web standards:
    - XML, SOAP, WSDL, UDDI, etc.
  - agent standards:
    - DAML+OIL (DARPA Agent Markup Language Activity), FIPA Agent Communication Language (ACL), Knowledge Interchange Format (KIF), Knowledge Query Manipulation Language (KQML), etc.
  - formal approaches to agent-oriented software;
  - methodologies for design and analysis of agent systems;
    - e.g., Gaia, Tropos;
  - tool support;
    - security in the context of agent-oriented software engineering (Bresciani);

UNIVERSITY OF KENT AT CANTERBURY

Dependability Technologies

- Fault tolerance:
  - most solutions use exception handling techniques:
    - classes of exceptions and handling mechanisms (Dellarocas & Klein):
      - infrastructure failures, protocol violations, and systemic exceptions;
    - the separation and encapsulation of exception handling for an agent environment in a special agent guardian (Tripathi & Miller);
    - application dependent, cannot exploit autonomy, and can restrict coordination;
  - no solutions that have exploited classes of faults;
    - not quite clear how to reach agreement in the presence of malicious faults (e.g., group communication algorithms);

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Dependability Technologies

It seems that so far there has not been major outcomes in the following two areas:

- Verification & validation (fault removal):
  - tests for multi-agents systems (Haendchen Fl., et. al);
  - model checking the behavioural description of the different modules of an agent (e.g., domain tasks);
- System evaluation (fault forecasting):
  - Markov models, and stochastic Petri nets;

Questions to the Panel

Complexity of open systems and the environment heterogeneity requires multiple coordination strategies.

- How to obtain *dependable agent communities* for open systems?
  - What are the challenges in terms of dependability technologies?
    - design time (rigorous design, and verification & validation);
    - run time (fault tolerance);
  - What are the restrictions that have to be imposed on agents and/or architectures for open systems?
  - How features commonly associated with agents can be exploited?
    - adaptability, autonomy, learning, mobility etc.

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Questions to the Panel

Agents can be autonomous and adaptive, and this in the legal and social context has its implications.

- Who should be held responsible if an agent fails, and causing the services delivered by its community to be catastrophic?
  - What are the safeguards that a community should have to deal entities “misdemeanours”?
  - How a society of agents can tolerate an agent failure?
    - What kind of redundancies should be considered in societal terms?

KENT
UNIVERSITY OF KENT
AT CANTERBURY

Panellists

- *Gul Agha (University of Illinois - USA)*
  - dependability is an aggregate property that can only be approximated;
- *Gruia Catalin-Roman (Washington University - USA)*
  - dependability is the desired outcome of a game played on multiple levels ;
- *Holger Giese (University of Paderborn - Germany)*
  - safety is different!;