Welcome to CREST

COW/SEBASE workshop

Mark Harman

Centre for Research in
Evolution, Search & Testing

Thursday, 10 December 2009
CREST is home to

4 faculty
1 administrative officer
8 post docs
12 PhD students
1 to 4 long term visitors

http://crest.dcs.kcl.ac.uk/
Testing

All kinds of testing:–
- Mutation testing
- Model based testing
- Automated test data generation
- Web application testing
- Service oriented testing
- Fuzz testing
- Security testing
- Regression testing
Dependence Analysis

dependence clusters
impact analysis
program and model based slicing
assessing third party code

http://crest.dcs.kcl.ac.uk/
Quantitative Information Flow

calculate amount of information flowing
find security leaks
use information theory to compute fitness

http://crest.dcs.kcl.ac.uk/
Service Oriented Computing

Testing services with services
Optimizing performance
Finding dependence

http://crest.dcs.kcl.ac.uk/
Clone detection using static analysis applied to code provenance applied to graphical languages

http://crest.dcs.kcl.ac.uk/
Clone detection

using static analysis
applied to code provenance
applied to graphical languages

http://crest.dcs.kcl.ac.uk/
Digital Humanities

Mapping outwards from software to systems
Software Engineering Analysis translates to
- musicology
- archaeology

http://crest.dcs.kcl.ac.uk/

Thursday, 10 December 2009
SEBASE

Search Based Software Engineering – SBSE
York, Birmingham, King’s
Motorola, IBM

http://crest.dcs.kcl.ac.uk/
What is SBSE?

In SBSE we apply search techniques to search large search spaces, guided by a fitness function that captures properties of the acceptable software artefacts we seek.

Genetic Algorithms, Hill climbing, Simulated Annealing, Random, Tabu Search, Estimation of Distribution Algorithms, Particle Swarm Optimization
Why is SBSE?

http://crest.dcs.kcl.ac.uk/

Thursday, 10 December 2009
Why is SBSE?
Why not SBSE?
Why not SBSE?

EPSRC network
1999 – 2002
Laid foundation for SBSE

http://crest.dcs.kcl.ac.uk/
<table>
<thead>
<tr>
<th>SBSE Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Transformation</strong></td>
</tr>
<tr>
<td><strong>Requirements</strong></td>
</tr>
<tr>
<td><strong>Effort prediction</strong></td>
</tr>
<tr>
<td><strong>Management</strong></td>
</tr>
<tr>
<td><strong>Heap allocation</strong></td>
</tr>
<tr>
<td><strong>Regression test</strong></td>
</tr>
<tr>
<td><strong>SOA</strong></td>
</tr>
<tr>
<td><strong>Refactoring</strong></td>
</tr>
<tr>
<td><strong>Test Generation</strong></td>
</tr>
<tr>
<td><strong>Maintenance</strong></td>
</tr>
<tr>
<td><strong>Model checking</strong></td>
</tr>
<tr>
<td><strong>Probe dist’ion</strong></td>
</tr>
<tr>
<td><strong>UIOs</strong></td>
</tr>
<tr>
<td><strong>Comprehension</strong></td>
</tr>
<tr>
<td><strong>Protocols</strong></td>
</tr>
<tr>
<td><strong>Component sel</strong></td>
</tr>
<tr>
<td><strong>Agent Oriented</strong></td>
</tr>
</tbody>
</table>

http://crest.dcs.kcl.ac.uk/

Thursday, 10 December 2009
<table>
<thead>
<tr>
<th>SBSE Applications in which SEBASE is active</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transformation</td>
</tr>
<tr>
<td>Requirements</td>
</tr>
<tr>
<td>Effort prediction</td>
</tr>
<tr>
<td>Management</td>
</tr>
<tr>
<td>Heap allocation</td>
</tr>
<tr>
<td>Regression test</td>
</tr>
<tr>
<td>SOA</td>
</tr>
<tr>
<td>Refactoring</td>
</tr>
<tr>
<td>Test Generation</td>
</tr>
<tr>
<td>Maintenance</td>
</tr>
<tr>
<td>Model checking</td>
</tr>
<tr>
<td>Probe dist’ion</td>
</tr>
<tr>
<td>UIOs</td>
</tr>
<tr>
<td>Comprehension</td>
</tr>
<tr>
<td>Protocols</td>
</tr>
<tr>
<td>Component sel</td>
</tr>
<tr>
<td>Agent Oriented</td>
</tr>
</tbody>
</table>

http://crest.dcs.kcl.ac.uk/

Thursday, 10 December 2009
York SEBASE team and colleagues

**Task allocation**
Reducing time to move between configuration
During development and maintenance
- to make systems robust to change
- to minimise cost when change can’t be avoided

**Wireless Sensor Networks**
- energy, performance, reliability, robustness

**Automatic WCET test-case generation**

**Statistical Testing**

**Non Functional Properties**
Optimize for functionality and power consumption

http://crest.dcs.kcl.ac.uk/
Birmingham SEBASE team

Multi objective optimization
- Power consumption
- Response time
- Cohesion and coupling for modularization

Algorithms
- Estimation of Distribution Algorithms
- Differential Evolution

Theoretical Analysis of SBST

Automatic bug fixing

Co-evolution of program and test cases

http://crest.dcs.kcl.ac.uk/
King’s SEBASE team

**Multi objective optimization:**
- cost and value
- faults, coverage, time
- metric sets
- risk and reward

**Software Engineering Application areas:**
- Requirements
- All kinds of Testing
- Project Management
- Refactoring
- Source Code Analysis

**Survey work, Repositories, Community support**

http://crest.dcs.kcl.ac.uk/

Thursday, 10 December 2009
King’s SEBASE team

**Multi objective optimization:**
- cost and value
- faults, coverage, time
- metric sets
- risk and reward

**Software Engineering Application areas:**
- Requirements
- All kinds of Testing
- Project Management
- Refactoring
- Source Code Analysis

Survey work, Repositories, Community support

http://crest.dcs.kcl.ac.uk/

Thursday, 10 December 2009
Trends in SBSE publication Growth
Trends in SBSE publication Growth
Trends in SBSE publication Growth

[Graph showing the growth of SBSE publications over time, with distinct bars for 'All' and 'Without SEBASE People' categories.]
Trends in SBSE publication Growth: Testing

Number of Publications on Software Testing

\[ y = 0.081 \times 1.2143^x \]

\[ R^2 = 0.9827 \]

Number of Publications on Structural Testing

\[ y = 0.0099 \times 1.2781^x \]

\[ R = 0.9413 \]

Thursday, 10 December 2009
New emerging areas
1st COW/SEBASE workshop

Talks and Lots of discussion
Lunch and refreshments will arrive here
Dinner tonight upstairs
Wireless
Recording of talks NOT discussion
A brief note about COWs

Thanks
- Yue and Jian for organisation
- EPSRC for funding

Logistics
- fire alarms
- toilets

http://crest.dcs.kcl.ac.uk/

Thursday, 10 December 2009