JAVA[†] THREADS WORKSHOP

The first Java Threads Workshop will be held at the University of Kent, England, on 23-24 September, 1996. This workshop is being organised by Java users (under the umbrella of the World occam and Transputer User Group) and will focus on design and performance issues for threaded applications.

Summary

Concurrency has become increasingly important for many areas of system design, whether we are building animated applets for business use over the internet or safety-critical controllers for car braking systems. This workshop will present the threads programming model incorporated into the Java language and consider how best to exploit it for constructing robust and efficient concurrent systems. Reports will be given from experienced developers of real-time and/or parallel applications on their reactions to and experiences with Java. The intention is to exchange information and initiate debate.

Java is an object-oriented language based on C. Unlike C++, it has removed language features (such as structures, typedefs and functions) that are made redundant by the introduction of classes, objects and methods, leaving users with little option but to follow object-oriented principles. Java has also taken issues of security seriously, eliminating many (but, possibly, not all) of those features of C that make its semantics so complex. For example, explicit pointers and the pre-processor are gone, replaced by full language support for packages, arrays and booleans. Concurrency is also supported by language binding, rather than a set of library functions, so as to give greater clarity of expression (and, ultimately, support from the compiler) to this notoriously error-prone area of design and implementation.

In their application of Occam's razor and their concern for security and concurrency, the Java designers were motivated by the same considerations as the designers of occam. It is for this reason that the occam community has looked seriously at Java and its application. There is certainly a clash of cultures between the two communities, but there exists great potential for learning on both sides. The Java threads model is based upon C.A.R.Hoare's concept of monitors, whilst the occam process model is based upon C.A.R.Hoare's (later) concept of Communicating Sequential Processes (CSP). The consequences of this divergence are subtle and worthy of investigation. This workshop will be looking hard into this.

In surveying the rapidly growing literature concerning Java, information about its support for concurrency is currently hard to find in any detail. The aims of the workshop, therefore, are to present the semantics of Java threads in as simple a way as possible, examine how it is being used to implement concurrent applications, compare and contrast it (both qualitatively and quantitatively) against occam processes and provoke debate and feedback from all participants.

Further details, including on-line registration, for the workshop have been placed on the Internet Parallel Computing Archive at:

`<URL:http://www.hensa.ac.uk/parallel/groups/wotug/java/>`
`<URL:ftp://unix.hensa.ac.uk/pub/parallel/groups/wotug/java/>`

Timetable

Please note that this is a first draft and subject to change. Details will be updated on the above URLs.

Monday, 23rd. September, 1996

11:00 – 14:00: *<Registration>*

13:00 – 14:00: *<Lunch>*

14:00 – 14:10: Introduction to the workshop
[Prof. Peter Welch, University of Kent, England]

14:10 – 14:45: Java Basics (classes, objects, methods, exceptions, threads and packages)
[Dr. Paddy Nixon, Univeristy of Dublin, Ireland]

14:45 – 15:30: Java Threads I (thread states – not-started, running, runnable, blocked and stopped; priorities; `yield`, `sleep`, `suspend` and `resume`; `synchronize`, `wait` and `notify`)
[Rick van Rein, University of Twente, Netherlands]

15:30 – 16:00: *<Coffee/Tea and biscuits>*

[†]Java is a registered trademark of Sun Microsystems Inc.

16:00 – 17:00: Java Threads II (design paradigms – activity threads versus object threads; user reports – case studies of some real Java multi-threaded applications)
[*speakers to be announced*]

17:00 – 18:30: Poster Session and Demonstrations (if you would like to contribute, please mail/fax/email a short – e.g. one page – abstract to the contact address below)

19:00 – 20:00: *<Dinner>*

20:00 – 24:00: Informal discussion groups and late bar (if you would like to suggest, or set up, an informal workgroup to focus on some particular area, please mail/fax/email us)

Tuesday. 24th. September, 1996

09:00 – 10:30: Comparison with occam (occam processes – unified objects and threads plus structure; using occam as a design language for Java; modelling occam channel and buffered communication between Java object threads; `ALT`s (fair and `PRI`), low-level `PAR` communications; occam3 shared channels; problems – spin locks, race hazards, infinite overtaking, starvation, deadlock and livelock; performance reports)
[Peter Welch, University of Kent, and Gerald Hilderink, University of Twente]

10:30 – 12:30: WorkGroups (including *<Coffee/Tea and biscuits>*). Participants will divide into groups to discuss various themes (e.g. "Concurrent programming in Java is Easy"). Each group will have a facilitator who leans in a particular (announced) direction. Participants may like to join groups whose leader's inclination opposes their gut feelings. Volunteers will minute the arguments and any conclusions reached. Minority reports will be acceptable.

12:30 – 13:00: Plenary Panel Session. Survivors of the workgroups reconvene for the final panel session. Panel members will be nominated by each workgroup. Brief reports from each panelist on the conclusions of their workgroup will be made. Open discussion from the floor. Have any problems with Java threads been identified? A workgroup will be appointed to write up the conclusions from the workshop.

13:00 – 14:00: *<Lunch>*

Registration Form

The cost of this workshop is £60, covering overnight accommodation on the 23rd, all meals and handouts. Extra nights accommodation, including breakfast, can be arranged at a cost of £16.90 pounds per night. Please tick below as necessary (one of the last two boxes must be checked):

[ ] overnight accommodation for Sunday, 22nd. September, 1996

[ ] overnight accommodation for Tuesday, 24th. September, 1996

[ ] vegetarian meals only

[ ] I enclose a cheque for the workshop registration fee (+ any extra nights)

[ ] I enclose an order form from my company/institution for the workshop registration (+ any extra nights)

NAME: EMAIL:

ADDRESS:

SIGNED: DATE:

Please email this form to <`J.Broom@ukc.ac.uk`> or post to:

Judith Broom (Java Threads Workshop)
Computing Laboratory
University of Kent, Canterbury
Kent – CT2 7NF
ENGLAND
(tel: +44 1227 827695; fax: +44 1227 762811)