Workshop on Exception Handling for a 21st Century Programming Language

May 14, 2001
Leuven, Belgium

To be held at the 6th International Conference on Reliable Software Technologies - Ada-Europe'2001

As the complexity of modern software systems grows, so does the need to deal reliably and efficiently with an increasing number of abnormal situations. The most general mechanism for this is exception handling, which is becoming a standard feature in modern languages.

A general exception handling mechanism should be well integrated with the other features of a language and conform to its programming paradigms. Increasing evidence from researchers and practitioners indicates that the exception handling in Ada 95 does not adequately reflect the whole range of programming paradigms supported by the language. In particular, the exception handling model remains based on Ada 83 while Ada 95 is object oriented. Furthermore, exceptions and concurrency are, arguably, not well integrated. A task with an unhandled exception dies silently, and one has to resort to asynchronous transfer of control for passing exceptions asynchronously between tasks. It is not clear that this solution extends well into a distributed environment. Yet another problem is the existence of anonymous exceptions.

New fault tolerance schemes based on existing exception handling facilities have been developed in research environments. This is important as it allows higher level abstractions providing more advanced mechanisms to be introduced without impacting the language definition.

The aims of the workshop are:

- to share experience on how to build modern systems that have to deal with abnormal situations;
- to discuss how solutions to those needs can be developed employing standard Ada features including the current exception handling paradigm; and
- to propose new exception handling mechanisms / paradigms that can be included in future revisions of the Ada language and also fit high integrity language profiles for safety critical systems.

Participation to the workshop is limited to 25-35 individuals and is by invitation upon acceptance of a submission. All types of submissions are welcome: brief position papers, experience reports, full research papers, etc. All papers will be made available to workshop participants before the workshop. The workshop will include talks based on the submitted papers and intensive shepherded discussion sessions. The submissions and a workshop summary will be published in Ada Letters.

Submissions should be sent electronically to Alexander Romanovsky: alexander.romanovsky@ncl.ac.uk

Electronic submission:	January 31, 2001
Notification:	March 15, 2001
Revised versions of papers:	April 15, 2001

Workshop Co-chairs:
Alexander Romanovsky (U. of Newcastle)
Alfred Strohmeier (EPFL)
Andy Wellings (U. of York)

Workshop Programme Committee:
Bill Bail (MITRE)
Jörg Kienzle (EPFL)
Pat Rogers (Software Arts and Sciences)
Bo Sanden (Colorado Technical U.)
Anand Tripathi (U. of Minnesota)
Tullio Vardanega (ESA)
Thomas Wolf (Paranor)

More information: www.cs.ncl.ac.uk/people/alexander.romanovsky/home.formal/eh-ada.html