Eclipse PTP and TAU

WITH K AND FX10 SUPERCOMPUTERS

Write code for parallel applications

Build applications on supercomputer

Visualise profiles and traces

Run parallel jobs

Analyse performance with TAU

Monitor jobs execution

HPC Usability Research Team of RIKEN AICS in collaboration with TAU developers from the University of Oregon work on adapting Eclipse Parallel Tools Platform (PTP) for developing parallell applications on the K and FX10 supercomputers.

The following features are now available to the K and FX10 users.

• Running batch jobs on K and FX10

To run jobs on supercomputers users have to do extra tedious and sometimes quite complicated work of writing job scripts with instructions to job managing system. Target System Configurations (TSC) we developed for the job scheduling system of the K and FX10 supercomputers let users easily configure their jobs parameters in GUI and submit jobs to the supercomputers. These configurations are saved and managed in Eclipse PTP run configurations.

• Monitoring jobs execution

Eclipse PTP monitoring perspective provides a list of submitted jobs and graphical representation of compute nodes of remote system. Users can check the status of their jobs, see on which nodes their jobs are running and check other useful information about their jobs. They can also cancel their own jobs.

• Profiling and tracing applications with TAU

TAU (Tuning and Analysis Utilities) have been installed on the K and FX10 computers enabling performance analysis of parallel application. "Profile configurations" feature of Eclipse PTP lets users set up parameters for multiple profiling or tracing jobs. We developed two TSCs for profiling and tracing parallel applications with TAU. "PJM-generic-TAU-profile" TSC enables automatic generation of job scripts for profiling jobs on the K and FX10 computers. "PJM-generic-TAU-trace" TSC can be used to generate job scripts necessary for running MPI tracing jobs.

• Local DB for profiles

After profiling job has finished profiles are automatically downloaded to user's computer and can be stored in a DB. Individual profiles from all processes of a parallel application can be automatically merged into one file that will be stored in the DB.

• Profiles and traces visualisation

Profiles are accessible from Eclipse UI and can be opened in ParaProf (included in TAU distribution) for visualisation. For traces visualisation Jumpshot-4 application can be used.

• Eclipse feature-rich Analysis tools for C, C++ and Fortran with MPI and OpenMP source code editing

Analysis tools provide advanced editing, error checking, and programmer assistance for developing parallel applications. These include content assist for MPI, OpenMP, UPC, OpenACC and OpenSHMEM APIs and directives, barrier analysis for MPI programs, and concurrency analysis for OpenMP programs.

• Refactoring for Fortran and C/C++

Refactoring features available in Eclipse PTP for Fortran source code include renaming, procedure extraction, subprogram refactoring, module refactoring, block refactoring, loop refactoring, etc. For complete list of features see https://wiki.eclipse.org/PTP/photran/documentation/photran7advanced). Refactoring for C/C++ code includes similar features.

• Automatic files synchronisation between local and remote computers

Eclipse PTP synchronised project feature ensures that all files get synchronised between user's local computer and remote one. Editing occurs locally and each file is synchronized with the currently active remote system when it is changed, created, or deleted.

• Numerous Eclipse plug-ins

• Building code on remote computer

• Version control with git and CVS

• Managing remote connections

Accomplishments

• Target System Configurations (TSC) for running applications on K and FX10

To run parallel applications on the K or FX10 computer users need to compose a job script with instructions to job scheduling system. These instruction include target resource group (queue), number of nodes, execution time limit, etc. TSC is a feature of Eclipse PTP that lets users define necessary job parameters in a GUI and generates a job script automatically. Thus it can save much time, especially for novice users.

We developed a TSC for “Parallelnavi” job scheduling system installed on the K and FX10 computers. TSC is an XML file parsed by Eclipse using JAXB framework. GUI of TSC lets users select values form lists or input arbitrary values where appropriate and these values are used in the generated job script.

• Monitoring applications on K and FX10

Eclipse PTP Monitoring perspective allows users to see a graphical representation of supercomputer nodes and list of running and queued jobs. Eclipse PTP supports some job management features, such as cancelling queued jobs. Monitoring is based on Large-scale Markup Language (LML) framework.

To support monitoring on the K and FX10 computers we extended LML framework with a set of Perl scripts for collecting and processing information obtained from “Parallelnavi” job management system installed on the supercomputers. Perl scripts produce file in LML format with data, which is used by Eclipse PTP for visualisation.

Graphical representations of supercomputer nodes designed for the K and FX10 computers reflect physical and logical node structure in terms of so-called tofu-units, racks and cabinets.

• TAU

Tuning and Analysis Utilities (TAU) is a set of tools designed for dynamic analysis of parallel applications. S.Shende from the University of Oregon has ported TAU to K/FX10 architecture. TAU and Program Database Toolkit (PDT), which is necessary for source code instrumentation, were installed on K and FX10 computers providing compiler-based and source-based instrumentation and tracing abilities.

TAU can be used from Eclipse PTP for profiling and tracing of parallel applications. To enable profiling and tracing with TAU we developed PJM-Generic-TAU-profile and PJM-Generic-TAU-trace target system configurations for Eclipse PTP.

Known issues

• Monitoring on K

The K computer installed at AICS RIKEN has over 80 000 compute nodes. Monitoring such a large number of nodes poses a performance problem for LML framework. Even processing information with only one user's own job excluding jobs of other users typically takes about 1.5 minutes.

• Visualising traces from > 1000 nodes

Trace files collected with TAU can be converted to SLOG-2 format for visualisation with Jumpshot-4 application. Jumpshot has problems with rendering large trace files (approximately larger then 1 000 nodes). Jumpshot is no longer developed or supported.

It is also possible to convert trace files to Paraver format. It works for most traces but sometimes fails with larger merged trace files.

• Merging trace files

Eclipse PTP doesn't have job results post-processing feature, so merging and converting trace files, for example, must be done manually.

Future work

• Extending Target System Configurations (TSC)

The K computer supports different kinds of batch jobs: normal jobs, bulk jobs, step jobs and workflow jobs. Supporting other than normal kinds of jobs with TSC is still to be done.

Another task for extending TSC is to add a flexible support for staging arbitrary files.

• Debugging

Eclipse PTP has support for debugging parallel applications. Enabling debugging on the K and FX10 supercomputers involves installing and configuring SDM on the supercomputer and developing a TSC that supports debugging.

• Monitoring

Besides a major task of making monitoring system work faster there are also the following tasks:

- Highlight users own jobs;
- Show approximate time till job start;
- Support all job managing tasks provided by the job scheduling system.

Credits

Eclipse PTP ..... *Open Source project lead by Greg Watson, a Senior Software Engineer at IBM.*

TAU ..... *Tuning and Analysis Utilities jointly developed by the University of Oregon, Los Alamos National Laboratory, and Research Centre Jülich, ZAM, Germany.*

Parallelnavi ..... *Batch Job Scheduling system developed by Fujitsu Limited.*

Jumpshot ..... *Developed by Anthony Chan, Bill Gropp and Ewing Lusk in Laboratory of Advanced Numerical Software of Argonne National Laboratory.*

LML ..... *Large-scale Markup Language designed at Forschungszentrum Juelich GmbH.*

http://www.hpcu.aics.riken.jp