LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-409241

The Center for Technology for Advanced Scientific Component Software (TASCS) Lawrence Livermore National Laboratory - Site Status Update

T. G. W. Epperly

December 5, 2008

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

The Center for Technology for Advanced Scientific Component Software (TASCS)
Lawrence Livermore National Laboratory - Site Status Update[1]
Period: April - September 2008
Tom Epperly <epperly2@llnl.gov> for Gary Kumfert, LLNL-PI
December 5, 2008

This report summarizes LLNL's progress for the period April through September of 2008 for the Center for Technology for Advanced Scientific Component Software (TASCS) SciDAC. The TASCS project is organized into four major thrust areas: CCA Environment (72%), Component Technology Initiatives (16%), CCA Toolkit (8%), and User and Application Outreach & Support (4%). The percentage of LLNL's effort allocation is shown in parenthesis for each thrust area.

Major thrust areas are further broken down into activity areas, LLNL's effort directed to each activity is shown in Figure 1. Enhancements, Core Tools, and Usability are all part of CCA Environment, and Software Quality is part of Component Technology Initiatives. The balance of this report will cover our accomplishments in each of these activity areas.

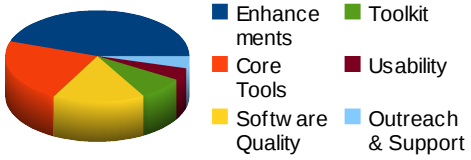

Figure 1: LLNL TASCS effort allocations to various thrust activities.

Enhancements

The goal of the Enhancements section of the CCA Environment thrust area is to extend and to augment the CCA specification and CCA implementations to provide new functionality to support scientific applications. LLNL's primary responsibility in this activity is to enhance Babel, the high-performance language interoperability tool. Thomas Epperly (LLNL) is responsible for coordinating this activity across all organization participating in TASCS.

The primary focus during the performance period was completing and testing the new multi-language struct feature. This feature extends Babel to include a structured data type similar to C/C++ structs and makes that capability available in all supported languages. The LLNL team guided and directed the Fortran 2003 struct binding implemented by Nanbor Wang & Stephen Muszala from Tech-X Corporation. LLNL provided the design, reviewed the implementation, and discussed the testing plan. The Fortran 2003 struct binding was tested with the C, C++, and Python struct bindings previously implemented by the LLNL team.

We also upgraded Babel to the latest release of the GNU Autotools suite: autoconf, automake, and libtool. This upgrade solves problems with building Babel on the latest Mac OSX release. Upgrading to a new revision of autotools is always a major undertaking due to Babel's multi-language build and support for both static and dynamic linking. Babel's sizable library of custom M4 macros had to be edited and debugged to accommodate changes in the new autotools release.

We incorporated parts of LANL's CHASM[2] library into Babel's software distribution. For it's Fortran 90 and Fortran 2003 bindings, Babel uses the array descriptor library from CHASM, a system created to improve Fortran and C/C++ interoperablity. Personnel changes at LANL left

1 LLNL-TR-409241

2 See http://chasm-interop.sourceforge.net/ for more information about CHASM. It was originally developed by Craig Rasmussen, Matthew Sottile, and Christopher Rickett while they were at LANL.

CHASM without a clear development team, so with Craig Rasmussen's approval, we extracted the features we needed from CHASM's array descriptor library, named that part CHASM-lite, and added it to Babel's source code repository. We extended CHASM-lite to support the PathScale compilers used on jaguar.

In addition, Tom Epperly mentored Monica Poche, an undergraduate summer student at LLNL. Monica initiated work on Batooki, a basic Babel toolkit of fundamental data structures. Monica's work demonstrated the feasibility of the toolkit but requires some additional hardening before it can be released as part of Babel.

Core Tools

The core tools activity, part of the CCA environment thrust, focuses on providing essential software maintenance, porting, and support in the face of changing HPC environments. LLNL's responsibility on this activity is to port Babel to DOE leadership class machines and to maintain it on existing systems.

During the report period, we helped port Babel to ORNL's Jaguar with Compute Node Linux, NERSC's franklin, and ANL's surveyor. In the case of Jaguar, we consulted with Jim Kohl (ORNL) to get Babel building and passing Babel's nightly testing suite. For Franklin, we modified Babel's custom M4 autoconf macros to facilitate proper configuration and building. In the case of surveyor, we verified that Babel's build and testing worked.

During our porting work, we discovered that the Fortran compilers on NERSC's franklin and Jacquard do not implement enough of the Fortran 2003 standard to support Babel's Fortran 2003 struct binding which makes extensive use of the new “bind(c)” feature. Until the compilers on these machines fully support the required Fortran 2003 features, applications will need to use a different approach for structs.

Porting Babel to new machines and maintaining it on existing machines required modifications to support new compilers and new compiler releases. For example, we had to change Babel's approach to C inline declarations to support the PathScale compiler suite. Supporting Portland Group compilers required changes to autoconf M4 macros to provide the correct compilation flags. Intel's version 10.0 compiler required a different flag than previous releases to generate position independent code (PIC) for shared libraries.

Usability

The Usability activity aims at making the tools in the CCA environment more accessible to new users. The activity's scope includes making each of the CCA tools more usable in addition to improving activities like debugging and testing. During the report period, LLNL's part of this activity focused on making Babel easier to use.

We added a new Makefile generation feature to Babel to help new Babel users build the Babel-generated code correctly. Our experience in the CCA shows that building software is a major stumbling block to new users, and providing an initial Makefile helps new users hit the ground running.

Several of the changes listed above also have benefits to Babel's usability. The autotools upgrade makes Babel easier to install on Mac OSX. By bundling CHASM-lite into Babel, we made Babel easier to configure and install. We improved CHASM-lite by adding configuration heuristics to automatically determine the vendor and type of Fortran compiler being used.

Software Quality

The software quality activity is part of Component Technology Initiatives, and work in this activity focuses on mechanisms to specify and verify functional software contracts associated with component interfaces. Software contracts help ensure that clients use components correctly and that the components themselves are correctly implemented. Tamara Dahlgren (LLNL) leads this activity across the organizations comprising TASCS.

During the report period at LLNL, work in this activity focused on completing contracts support in Babel to enable the first official release with contracts support. The work involved making changes to Babel's Intermediate Object Representation (IOR), the fundamental data structure that supports Babel's object model and runtime polymorphism. Tamara completed the C and C++ client-side binding and developed the Python, Java, and Fortran bindings from scratch. In addition, she implemented additional tests in Babel's nightly regression suite to exercise the new features and language bindings. She used valgrind and other debugging techniques to identify and plug memory leaks associated with the contracts extensions.

In addition to the development achievements, Tamara published information about her research. She added some material to the Babel Users' Guide and was lead author for a poster and 5-page abstract for the SciDAC conference.

- Dahlgren, T. L., D. E. Bernholdt, and L. C. McInnes, “Gaining Confidence in Scientific Applications through Executable Interface Contracts,” SciDAC 2008, July 13-17, 2008, (LLNL-POST-405149 and LLNL-CONF-404758).

Toolkit

The goal of the CCA Toolkit thrust is to provide a diverse suite of scientific components along with a basic software skeleton to facilitate the creation of new components. During the report period, LLNL is supporting the TASCS Toolkit thrust by introducing a “--cca-mode” to Babel to support Bocca, a tool for rapid CCA component development. The new Babel CCA mode generates extra markings in the Babel-generated code to make it easier for Bocca to direct developers to parts of the Babel-generated code that need to be edited.

Outreach & Support

The User and Application Outreach and Support thrust focuses on assisting applications groups with CCA adoption and usage through direct interactions and development of documentation, tutorials, and example materials. The LLNL team contributes to this by supporting end users, by co-presenting at the ACTS Toolkit meeting at LBL, and by building relationships with other projects.

Because Babel has a relatively large customer base, there is a fairly steady stream of support requests. These include requests for help, bug reports, and feature requests. Many of these come by email and aren't tracked. Our Roundup-based issue tracker logged 40 new support requests in the report period, and our team retired 31 support requests.

Our outreach to other projects focused on FACETS and Chapel. Tom Epperly is co-funded by TASCS and FACETS, so he operates as liaison between the two SciDAC projects. In addition through discussions with Cray's Chapel project, we developed a proposal to extend Babel to support PGAS languages and Chapel that would provide language interoperability to Chapel. This proposal was submitted to the Office of Science's petascale tools CFP.

Personnel

Gary Kumfert left LLNL on October 31 to pursue a career at conviva.com, a real-time internet video delivery company. Tom Epperly has taken over responsibility as LLNL-PI for the TASCS SciDAC. Tamara Dahlgren earned her PhD from UC Davis in May, and she is still serving as activity lead for the Software Quality initiative. LLNL is actively recruiting a post-doc to provide resources needed for the project.