



Russian Academy of Sciences Program Systems Institute

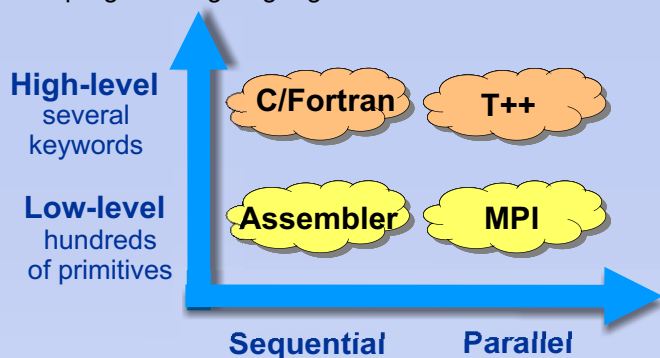
Porting OpenTS Parallel Programming System to Windows CCS Platform

OpenTS Outline

Open T-System was developed at PSI RAS within the framework of the "SKIF" supercomputer joint project of Russia and Belorussia (2000-2004).

OpenTS is a parallelization technology based on the T-System ideology and principles. It provides automatic dynamic parallelization of a program.

OpenTS provides a runtime for the T++ high-level programming language that is a parallel dialect of the C++ programming language.



In this dialect programmer uses a few additional keywords to write parallel programs. Keyword "tfun" designates T-functions. Its invocations produce grains of parallelism. Keywords "tval", "tout" and "tptr" designate input and output data that can be shared by or transferred between T-functions. "tdrop" makes a variable value ready, "twait" waits for a variable value to be ready and "tct" explicitly specifies T-context.

T-System supports millions of lightweight threads for each processor. Runtime environment provides very fast context switching between lightweight threads. It takes only a few nanoseconds. On top of this were realized a lot of important elements and features: mobile threads, mobile objects and mobile references, distributed garbage collection, whiteboard that helps realize data and task exchanges, and so on.

OpenTS supports a wide spectrum of parallel platforms: multi-core processors, SMP-systems, clusters, meta-clusters and grids.

OpenTS Applications

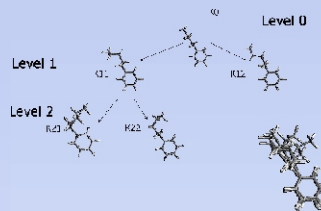
All applications were developed by domain experts who are novices in T++ programming.

The MultiGen system was written in T++ by a group of chemistry engineers from Chelyabinsk State University.

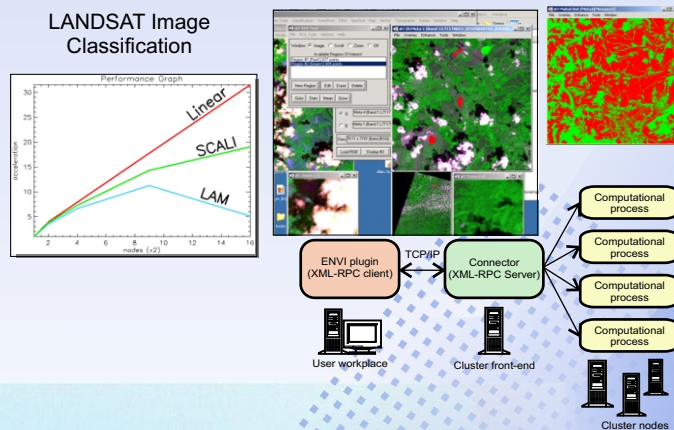
This tool is used to derive a multi-conformational model of a molecule to predict biological activity of drugs.

The MultiGen application demonstrates good speed-up factors for different real drugs.

The next T-application was developed by the scientists from the Institute of Mechanics, which is a branch of the Moscow State University. It is used for modeling of airflow streaming around complex constructions that have no streamlined or smooth shape; specifically, parachute systems, sails, antennas, wind-powered generators and so forth. The approach that is used in the application was developed by a Russian academician Belocerkovski. The overheads of T-System are only about 30-40% percents and the speed-up growth is quite good.



There are a lot of remote sensing applications developed within the framework of the "SKIF" supercomputing project. For instance, one of them is used to categorize pixels of image data observed from a LANDSAT satellite. An operator points to certain small areas of the input image and indicates each area's type: forest, field, lake, etc. These selections define different categories of the surface. The application classifies all image pixels to these predefined categories. This T-application demonstrates a rather good scalability — 18 times on 32 CPUs — and it has some room for improvements.

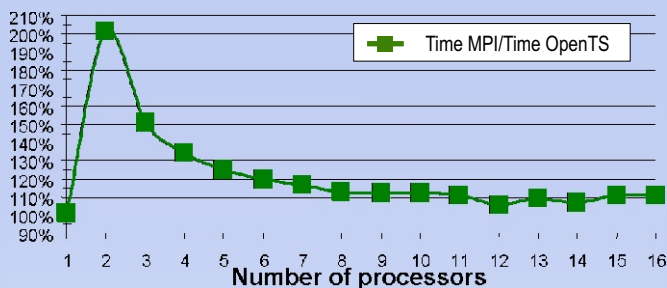




OpenTS vs MPI

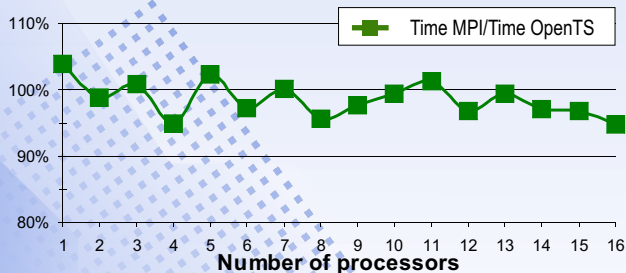
In 2005 PSI RAS and Microsoft negotiated a contract on porting two MPI applications to the T++ basis. The goal was to compare performance of different parallel implementations of PovRay and ALCMD.

PovRay is a well-known parallel application widely used on different supercomputing platforms. It is a ray tracing application used for obtaining realistic images. Implemented originally for unicomputers, it was then extended with MPI (around the year 2000) for parallel platforms. PovRay itself is a sequential program. PovRay 3.50c MPI patch uses about 3000 lines of code. Parallel PovRay implementation under OpenTS is much more compact and simpler than parallel implementation under MPI. OpenTS patch to PovRay is shorter than 200 lines. Efficiency of parallel implementations is comparable for MPI-PovRay and T-PovRay and the fraction (time MPI / time OpenTS) is varied in range from 100% to 200%.



2CPUs AMD Opteron 248 2.2 GHz RAM 4GB, GigE, LAM 7.1.1

ALCMD is a chemical application used for molecular dynamic modeling. It is written in Fortran and uses a lightweight message-passing library MP_Lite which can be used itself or may act as a thin layer on top of MPI. MP_Lite middleware was replaced by T++ module. Original Fortran code of ALCMD was left intact. Original size of MP_Lite library is more than 20 000 lines of C code. Reduced subset of MP_Lite source code which enables to run ALCMD over MPI is about 3500 lines. MP_Lite abstraction layer implementation that supports all functions required by ALCMD contains less than 500 lines of code in T++. As a result, OpenTS realization of MP_Lite is much more compact and simpler than MPI version of MP_Lite. Efficiencies of these implementations are comparable and overhead of T++ is varied in the range of $\pm 7\%$.



2CPUs AMD Opteron 248 2.2 GHz RAM 4GB, GigE,
LAM 7.1.1, Lennard-Jones MD, 512000 atoms

OpenTS for Windows

Many scientists prefer to use Windows for high-performance computing. They are using C/C++ languages for high-performance application development. Scientists are mostly concentrated on domain-specific problems and do not appreciate any kind of MPI-programming.

In 2006 Microsoft will release a new platform, Windows Compute Cluster Server (WCCS), designed specially for high performance computing. A wide variety of cluster solutions follows. There will be a business competition between Windows and Linux cluster solutions. This will lead Windows parallel programming systems to the state of being relevant.

In the early 2006 Microsoft and PSI RAS negotiated a contract on porting OpenTS parallel programming system to Windows.

OpenTS under Windows inherits all the basic features of the original Linux version.

A unified installer of OpenTS for Windows XP/2003/WCCS editions provides the following features:

- support for x86 and AMD64 platforms;
- installation of WCCS SDK (that contains MS-MPI) if necessary;
- integration with Visual Studio 2005;
- OpenTS testing procedure.

OpenTS under Windows does not require any commercial compiler for T-program development.

It is only enough to install Visual C++ 2005 Express Edition available for free on Microsoft website.

Besides working in command-line mode there is a possibility of development and building T-applications in Visual Studio 2005 IDE.

