

# Кроссплатформенная версия T-системы с открытой архитектурой

С.М. Абрамов, А.А. Кузнецов, В.А. Роганов

Работа посвящена разработке кроссплатформенной исследовательской версии системы автоматического динамического распараллеливания OpenTS. Эта версия OpenTS построена на переносимых исходных кодах и может работать как в ОС Linux, так и на платформе Windows CCS.

## 1. Введение

За последние годы в мире значительно вырос интерес к высокопроизводительным вычислениям и они получают широкое распространение. Нужно отметить, что с ростом производительности микропроцессоров падает удельная стоимость кластерных решений (цена/производительность). Это делает доступным применение высокопроизводительных вычислений (HPC) не только в крупных компаниях, но и в небольших рабочих группах. Более того, современные процессоры являются многоядерными и средства распараллеливания программ скоро могут оказаться полезными и даже необходимыми на обычном персональном компьютере.

Таким образом, вопрос широкой применимости высокопроизводительных вычислений сегодня все более становится зависимым от решения проблем в области программного обеспечения, а не аппаратуры. На первый план выходят проблемы удобства разработки и отладки параллельных приложений, сокращения сроков и совершенствование инструментариев их разработки.

В настоящее время большая часть рынка ПО для высокопроизводительных систем занята решениями, основанными на ОС Linux и других UNIX-системах. Однако, в IT-отрасли значительное количество разработчиков используют (или хотят использовать) Windows в качестве начальной среды разработки и отладки своих программ. Корпорация Microsoft, лидер на рынке операционных систем, в 2004 году открыла у себя новое подразделение по направлению высокопроизводительных вычислений. Это была реакция Microsoft на тот факт, что рынок HPC становится все более массовым. Последовавшие за этим реальные шаги и политические жесты корпорации Microsoft не оставляют никаких сомнений в ее решимости воплотить свою идею повсеместного внедрения параллельных и высокопроизводительных вычислений на платформе Microsoft Windows Compute Cluster Server 2003 (WCCS) — новой ветки в линии ОС Windows, специально разрабатываемой для кластерных архитектур (выпуск официальной версии состоялся летом 2006 года). Тем самым, следует ожидать появления большого числа кластерных решений для Windows-платформ. Это должно составить значимую конкуренцию Linux-решениям [4] (как по цене, так и по производительности) и приведет к востребованности систем распараллеливания программ для Windows-кластеров.

Корпорация Microsoft стимулирует создание таких технологий распараллеливания. Она передала в ИПС РАН аппаратные средства (кластер) и программное обеспечение Microsoft WCCS, и оформила контракт с ИПС РАН на разработку версии OpenTS (T-системы с открытой архитектурой [1, 2]) для платформы WCCS.

## 2. Платформа Windows Compute Cluster Server

Платформа WCCS [3] состоит из двух компонентов:

- Операционная система Windows Compute Cluster Server 2003, которая базируется на ядре Windows Server 2003 Standard x64 Edition и поддерживает только 64-разрядные аппаратные платформы (но некоторые компоненты, например, MS-MPI, способны работать и под 32-разрядными ОС);

- набор кластерных решений Compute Cluster Pack (CCP), содержащий все необходимые программные компоненты для создания, использования и управления кластерной инфраструктурой.

Стоимость ОС Windows CCS 2003 ниже, чем у любого другого издания Windows Server 2003.

Кластер под управлением WCCS состоит из одного главного и любого числа вычислительных узлов, связанных высокоскоростной сетью. Главный узел также может выполнять роль вычислительного узла. Вычислительные задачи можно передавать на кластер через приложение Compute Cluster Job Manager, а мониторинг и управление кластером осуществляются с помощью инструмента Compute Cluster Administrator. Оба инструмента поставляются вместе с CCP. Для подачи заданий можно также использовать интерфейс командной строки (Command-Line Interface, CLI). Для развертывания узлов можно использовать службу удаленной установки (Remote Installation Services, RIS), которая поставляется вместе с ОС Windows CCS 2003.

Для разработки параллельных приложений на платформе WCCS корпорация Microsoft разработала собственную реализацию стандарта MPI — MS-MPI. Она разработана на основе MPICH2, открытой реализации стандарта MPI 2.0. Это дает возможность независимым производителям ПО легко портировать свои продукты на WCCS.

### 3. Проекты, предшествующие переносу

Как уже было сказано выше (см. раздел 1), в 2006 году корпорация Microsoft заключила с ИПС РАН контракт на перенос средств параллельного программирования OpenTS на платформу Windows Compute Cluster Server. Этому предшествовала исследовательская работа по сравнению эффективности разработки параллельных приложений при помощи двух различных средств: MPI и OpenTS.

Сравнение выполнялось в следующем порядке:

1. Были выбраны две прикладные системы, ранее написанные на C++, Fortran-77, MPI большой группой разработчиков в течение длительного периода времени. В качестве таких систем корпорация Microsoft отобрала POVRay и ALCMD:
  - POVRay — приложение для построения реалистичных изображений методом трассировки лучей;
  - ALCMD — приложение для моделирования молекулярной динамики;
2. Сотрудникам ИПС РАН (4–6 человек) предстояло в сжатые сроки (2–3 месяца) переписать MPI-зависимую часть кода этих приложений на OpenTS.
3. Главная цель при этом — получение более компактного, более читабельного кода, сокращение трудоемкости (меньше людей, сжатые сроки). При этом эффективность результирующей программы не должна была сильно пострадать: считалось допустимым пожертвовать 30% производительности за преимущества в процессе программирования.

Исследования показали, что при существенном сокращении объема и сложности кода (в 7–15 раз), OpenTS-версии приложений иногда превосходят, а иногда сравнимы по производительности с их MPI-аналогами (см. Рис. 1). Этот факт, а также существенное уменьшение накладных расходов на разработку параллельных T-приложений позволят ученым разрабатывать эффективные параллельные научные приложения, не отвлекаясь на изучение сложных технологий распараллеливания (таких, как MPI).

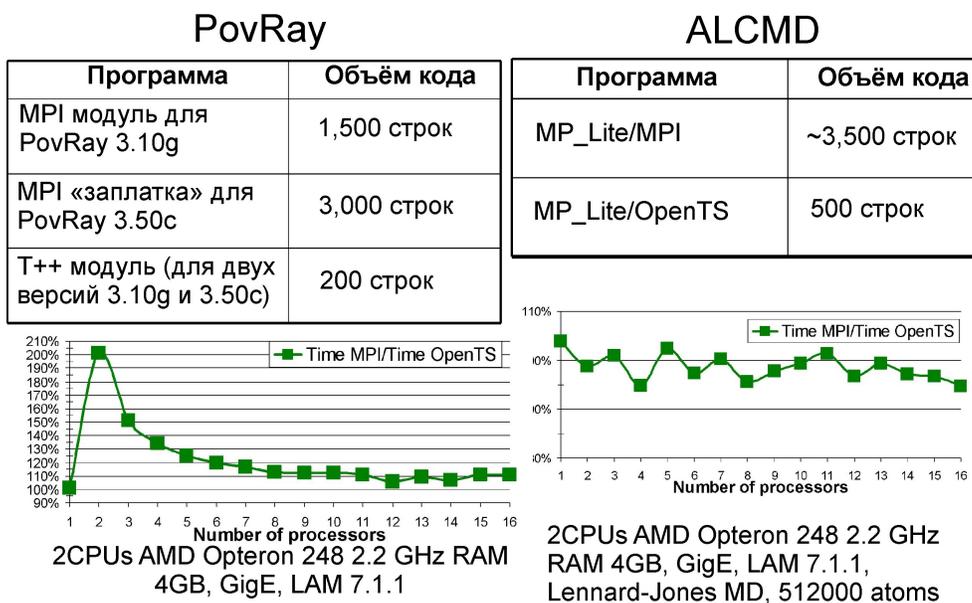


Рис. 1. Результаты проектов переноса MPI-приложений на OpenTS

## 4. Перенос OpenTS под Windows

Портирование основных компонент OpenTS, конвертера и микроядра, осуществлялось в среде разработки Microsoft Visual Studio 2005 с использованием компилятора Visual C++.

T-конвертер представляет из себя транслятор с языка T++ (параллельный диалект C++) на язык C++. Он базируется на известном программном продукте OpenC++, предназначенном для создания языковых трансляторов. Конвертер производит синтаксический разбор T-программы и преобразует языковые расширения T++ в код на языке C++. Специфика языка Visual C++ состоит в добавлении в него ряда новых синтаксических конструкций (например, «`__stdcall`», «`__declspec`», «`__int64`» и др.). T-конвертер был доработан таким образом, чтобы правильно обрабатывать эти конструкции.

Микроядро OpenTS — это среда поддержки исполнения T-приложений. При портировании микроядра преследовалась цель создать его кроссплатформенную версию. Было принято решение создать отдельный компонент исходного кода — Platform Abstraction Layer (PAL) — в который были вынесены все платформенно-зависимые функции ядра:

- функции обеспечения многопоточности T-приложений, поддержки SMP-режима;
- определение свободных ресурсов на узле;
- прочие функции системных API (например, запрос числа процессоров или размера страницы памяти, подсветка консольного текста и др.).

Быстрое переключение контекста (легковесные потоки) в оригинальной версии OpenTS было реализовано в виде ассемблерной вставки в код микроядра. Встроенный ассемблер в Visual C++ имеет ограничение, не позволяющее использовать ассемблерные вставки для сборки 64-разрядных приложений. В связи с этим, функция реализации переключения контекста была написана в виде статической библиотеки, которая компонуется с T-приложением во время его сборки.

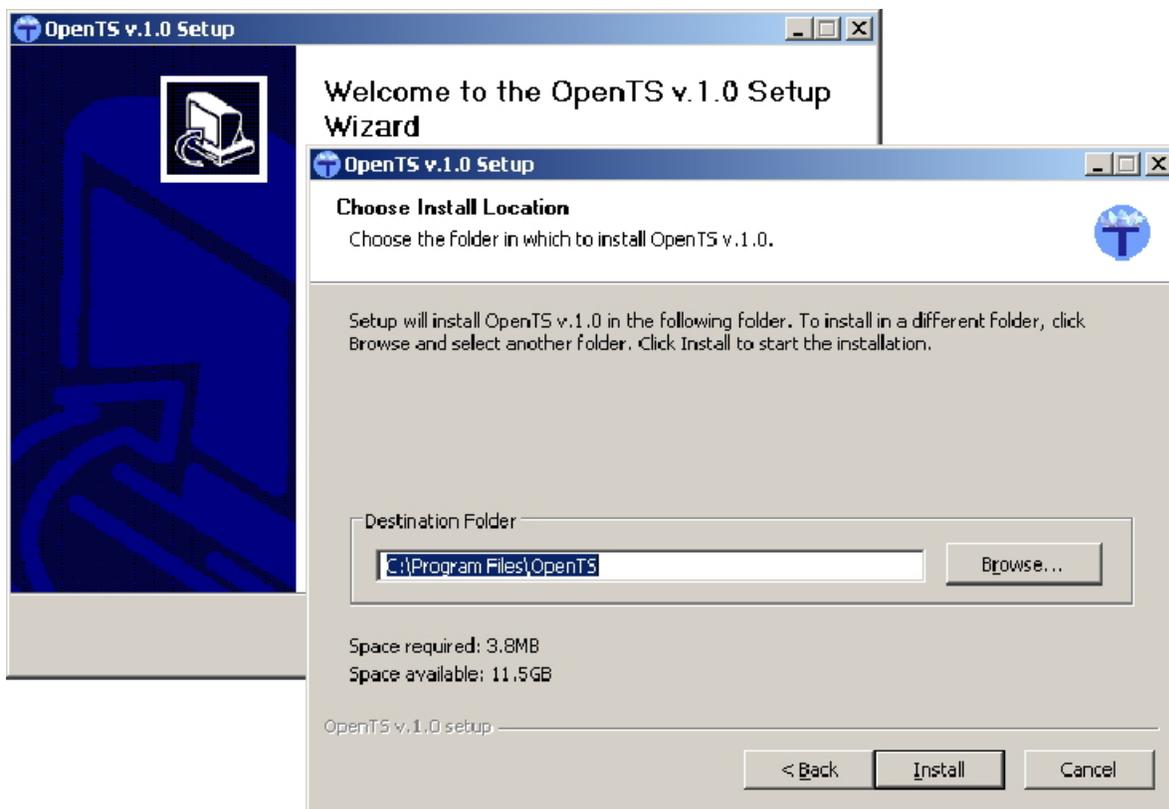


Рис. 2. Установка OpenTS под Windows

## 5. Итоговые решения

### 5.1. Инсталлятор

Для установки OpenTS на ОС Windows разработан инсталлятор (см. рис. 2), который поддерживает платформы x86 и AMD64. Для установки и работы OpenTS требуются:

- Microsoft Compute Cluster Pack Software Development Kit (CCP SDK) , содержащий программное окружение для разработки и запуска параллельных программ;
- компилятор Visual C++ 2005, который поставляется как отдельно (Visual C++ 2005 Express Edition, свободно доступный на сайте Microsoft), так и с любым изданием Microsoft Visual Studio 2005.

Нужная версия CCP SDK может быть установлена непосредственно при инсталляции OpenTS в том случае, если инсталлятор не обнаружит SDK, ранее установленный на компьютере пользователя.

В том случае, если не установлен Visual C++ 2005, либо в случае отказа от инсталляции CCP SDK установка OpenTS отменяется.

Предвидя многообразие системных конфигураций, инсталлятор содержит средства автоматического тестирования работоспособности OpenTS. Процедура тестирования проходит в несколько этапов:

1. сборка микроядра OpenTS;
2. сборка простой Т-программы fib, вычисляющей заданное своим номером число Фибоначчи;

3. в том случае, если при сборке микроядра или `fib` возникли ошибки, разработчикам OpenTS автоматически отсылается письмо с описанием ошибок;
4. производится запуск приложения `fib`; в том случае, если результат его работы не соответствует ожидаемому, установка OpenTS отменяется.

## 5.2. Инструменты сборки T-приложений

Для сборки T-приложений создан командный сценарий `t++.bat`, принимающий следующие аргументы:

- `/auto-c-call` — эта опция сборки позволит T-приложению вызывать Си-версии T-функций, что может в некоторых случаях существенно ускорить работу T/приложения;
- `/c` — компиляция исходных файлов без компоновки, в результате чего создаются объектные модули программ;
- `/dbg` — режим отладочной сборки, который позволит программе-отладчику в случае сбоя в работе T-программы получить символьную информацию о программе;
- `/do dir` — опция, позволяющая указать расположение объектных модулей;
- `/not` — сборка программы для запуска в скалярном режиме, при котором все специальные ключевые слова удаляются из T-программы;
- `/o out.exe` — эта опция позволяет указать имя исполняемого файла, получаемого в результате сборки; если эта опция отсутствует, то в качестве имени исполняемого модуля будет использовано имя первого исходного файла;
- `/p opt` — передача опции «opt» используемому компилятору C/C++;
- `/v` — печать команд и всего программного вывода;
- все остальные аргументы принимаются за исходные файлы на языках C++/T++.

T-приложения можно запускать в скалярном и в параллельном режиме, причем как из командной строки (см. Рис. 3), так и на Windows-кластере с использованием Compute Cluster Manager, которая поставляется вместе с Microsoft Compute Cluster Pack.

```

c:\ Open T-System Command Prompt (AMD64)
Setting environment for using Microsoft Visual Studio 2005 x64 tools.
Setting environment for using Open T-System AMD64 tools.

C:\Program Files\OpenTS>mpiexec -n 2 EP
Open T-System Runtime v3.0, 2003-2006, PSI RAS, Russia.
Running under MS-MPI on 2-rank cluster:
<[3.6Gf,1000B,1.50GiB]*2> ~=[7.2Gf,2000B,3.00GiB]
Starting tfun main, good luck!

Tasks activated:      [1013/1024/1035]
Tasks exported:      [1/1/2]
Msgs sent:           [58/58/59]
Async Msgs:          [0/0/0]
Msgs size:           [7256/7368/7480]
Taskboard visits:    [2389/2950/3512]
Scheduler time:      [0.017/0.020/0.024]
MPI time:            [0.001/0.001/0.001]
Idle time:           [0.031/0.043/0.053]
Tasks time:          [15.769/15.804/15.840]
Total time:          [15.974/15.988/16.002]

C:\Program Files\OpenTS>_

```

Рис. 3. Запуск T-приложения в параллельном режиме

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  tfun int fib (int n)
5  {
6      if (n < 2) return 1;
7      return (fib(n-1) + fib(n-2));
8  }
9  tfun int main (int argc, char* argv[])
10 {
11     int n = atoi(argv[1]);
12     printf("Fib(%d)==%d\n",n,(int)fib(n));
13     return 0;
14 }

```

Рис. 4. Пример Т-программы

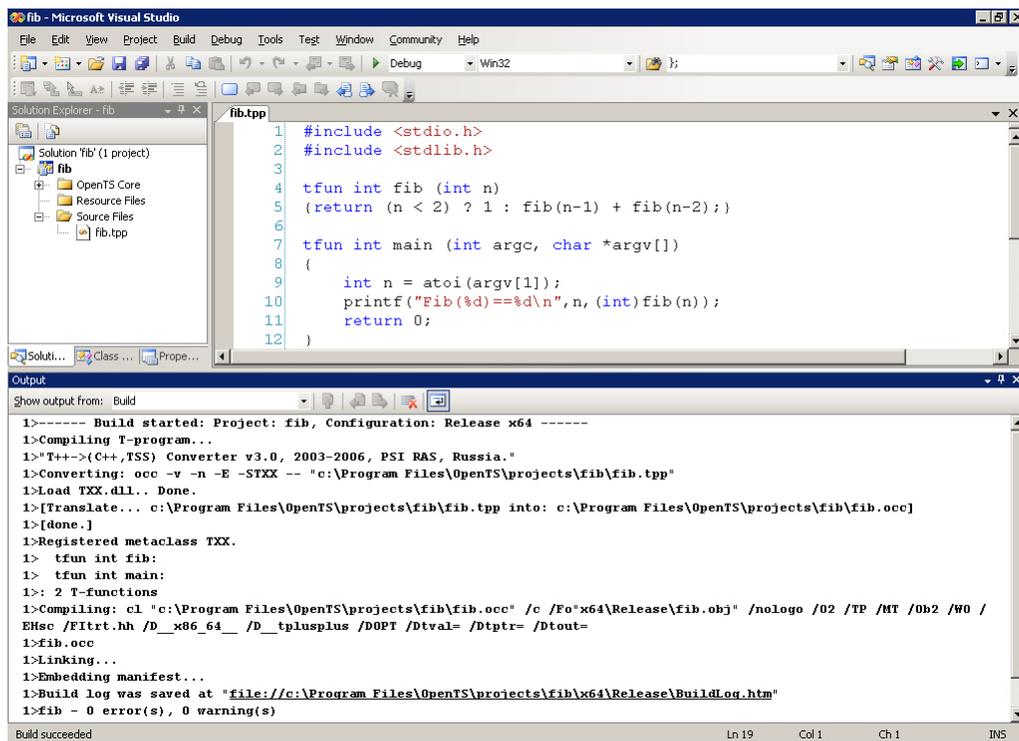


Рис. 5. Сборка Т-приложения в среде разработки Microsoft Visual Studio 2005

### 5.3. Набор для разработчика (SDK)

Набор для разработчика системы OpenTS предоставляет следующие возможности:

- сборка микроядра — возможность делать правки и пересобирать микроядро из его исходных кодов;
- сборка расширений для микроядра — это позволит совершенствовать OpenTS добавлением в нее расширений; примеры уже имеющихся расширений — DMPI (Dynamic MPI), WAD (Wide Area Debugger), VisTrace (журналирование работы Т-программы);
- сборка Т-приложений — инструменты из стандартного дистрибутива OpenTS. Кроме того, в набор включены примеры простых Т-программ.

Для использования каждой возможности в дистрибутив OpenTS SDK включены проекты формата Visual Studio 2005 (файлы .vcproj), с которыми можно работать в среде разработки Visual Studio 2005.

## 5.4. Интеграция с Visual Studio 2005

Сборка T-программ возможна в среде разработки Visual Studio 2005 (см. Рис. 5.2). Реализована возможность добавления проекта создания консольного T-приложения в существующее решение, а также добавить новый исходный файл T-программы в существующий проект.

Сборкой T-приложений в среде Visual Studio 2005 управляет особое правило сборки, которое во время сборки применяется к каждому исходному файлу T-программы. Затем редактор связей компоует вместе объектные модули T-программ, микроядра и некоторых статических библиотек.

## 6. Заключение

В ходе работы над проектом была разработана кроссплатформенная версия системы автоматического динамического распараллеливания программ (OpenTS). Она содержит следующие базовые компоненты:

- T-конвертер, предназначенный для трансляции программ на языке T++ в программы на языке Си++.
- T-микроядро, которое представляет собой среду поддержки исполнения языка T++.

Кроме стандартных средств разработки T-приложений из командной строки, также обеспечена возможность разработки T-приложений в среде Microsoft Visual Studio 2005.

Создан комплект программ для разработчика OpenTS (SDK). Он позволяет вносить изменения в микроядро и разрабатывать расширения для него.

Для тестирования отдельных функциональных характеристик T-системы с открытой архитектурой на платформе Windows был адаптирован комплект простых тестов, использовавшийся ранее для этих целей на платформе Linux.

Авторы надеются, что система OpenTS позволит упростить создание параллельных приложений в среде Windows. Это послужит широкому распространению методов и технологий параллельного программирования.

В дальнейшем планируется реализация поддержки еще некоторых других аппаратных платформ. Также планируется более тесная интеграция OpenTS и Windows для увеличения производительности T-приложений для платформы WCCS.

## Благодарности

Эта работа выполнена в рамках НИР «Сравнительное исследование эффективности технологии параллельного программирования OpenTS с технологией MPI» и НИР «Перенос средств параллельного программирования OpenTS на платформу Windows Compute Cluster Server» (выполнено в ИПС РАН по контракту с корпорацией Microsoft), а также в рамках проекта 1.3 «Разработка и реализация языков T# и T++ и соответствующих им средств для эффективной поддержки высокопроизводительного параллельного счета» при поддержке Программы фундаментальных научных исследований ОИТВС РАН «Оптимизация вычислительных архитектур под конкретные классы задач, информационная безопасность сетевых технологий».

## Литература

1. Абрамов С.М., Адамович А.И., Инюхин А.В., Московский А.А., Роганов В.А., Шевчук Ю.В., Шевчук, Е.В. T-система с открытой архитектурой // Суперкомпьютерные системы и их применение SSA'2004: Труды Международной научной конференции, 26–28 октября 2004 г. Минск, ОИПИ НАН Беларуси. — Минск, 2004, с.18–22.

2. Abramov S., Adamovich A., Inyukhin A., Moskovsky A., Roganov V., Shevchuk E., Shevchuk Yu., Vodomerov A. OpenTS: An Outline of Dynamic Parallelization Approach // PACT'2005, LNCS.
3. Don Jones «Windows Goes High Performance» — <http://www.redmondmag.com>, Redmond Magazine, 2006. (англ.)
4. Roger Howorth «Windows invades Linux territory». — <http://www.itweek.co.uk>, IT Week, 2006. (англ.)