

Параллелизм – ключ к производительности в мультитядерных платформах

Николай Исаков

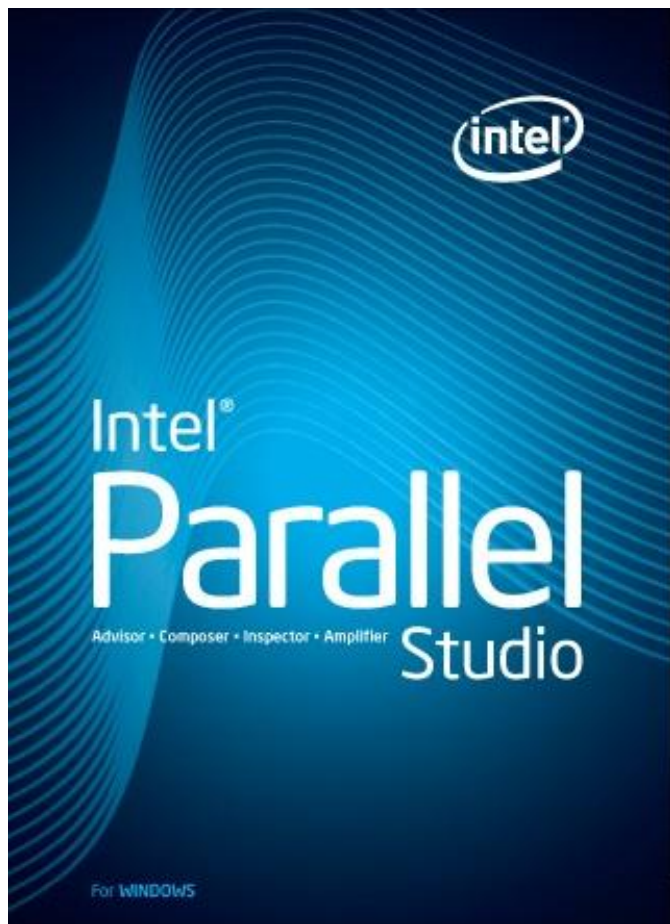
Руководитель программы поддержки независимых разработчиков ПО в России и странах СНГ

Software & Services Group - Developer Relations Division



Intel® Parallel Studio

Полный набор инструментов параллельного программирования разработчикам на C/C++ для Windows



- **Многолетний опыт параллельной разработки в высокопроизводительных вычислениях (HPC)** теперь применим для клиентских приложений работающих на ПК и ноутбуках
- Новые **основанные на стандартах** продукты помогают разработчикам использовать параллелизм в приложениях для Windows
- Все инструменты взаимодействуют с IDE и являются **расширением Microsoft Visual Studio**
- Все продукты поддерживают **разработку для мультитядерных платформ сегодня** и помогают масштабировать приложения для многоядерных систем в будущем

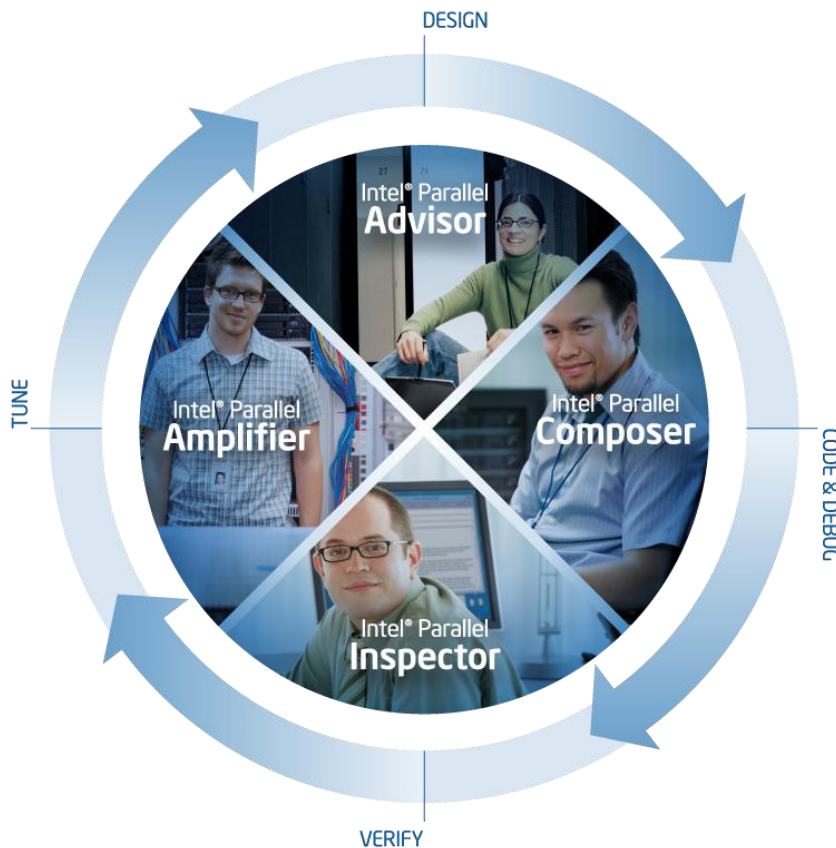
Intel® Parallel Studio

- Параллельное программирование – не простая и не новая задача, имеющая широкое поле для инноваций
- **Intel занимает лидирующие позиции** в индустрии, где используется масштабный параллелизм
 - Сегодня инструменты Intel помогают масштабировать параллельные приложения на тысячи процессоров
 - Большинство “Топ 500” суперкомпьютеров
- **Intel занимает лидирующие позиции** в индустрии параллелизма для мультиядерных платформ
 - Intel® TVB лидирует в разработке на C/C++ поддерживая несколько платформ и вызывая желание сделать подобное
 - Intel вносит вклад в развитие и поддержку OpenMP* уже более 10 лет
 - Библиотеки производительности Intel повсеместно используются в параллельных приложениях
 - Intel® C++ Compiler, Intel® Thread Checker, Intel® Thread Profiler, Intel® VTune Performance Analyzer стали основными инструментами для разработки высокопроизводительных параллельных приложений
- Сегодня: Сделать доступной разработку масштабируемых параллельных приложений в **Microsoft* Visual Studio*** большинству программистов
 - Возможность сегодня разрабатывать параллельные приложения, которые будут масштабироваться на многоядерных системах завтра, - это основная задача сохранения инвестиций.

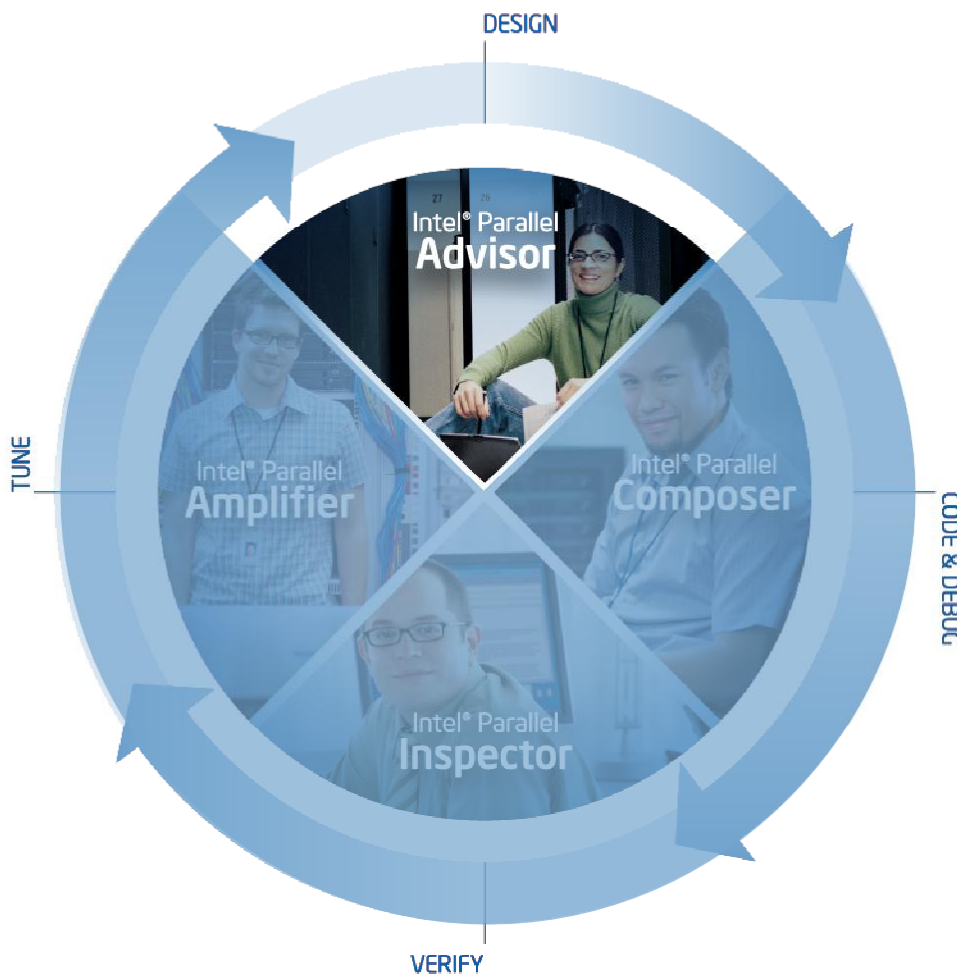
Intel® Parallel Studio

Упрощает 4 этапа параллельной разработки

Parallelism Development Cycle



- **Parallel Advisor**
Находит возможности для распараллеливания
- **Parallel Composer**
Создает параллельные приложения с помощью компилятора и библиотек
- **Parallel Inspector**
Проверяет корректность параллельных программ
- **Parallel Amplifier**
Оптимизирует для параллельной масштабируемости



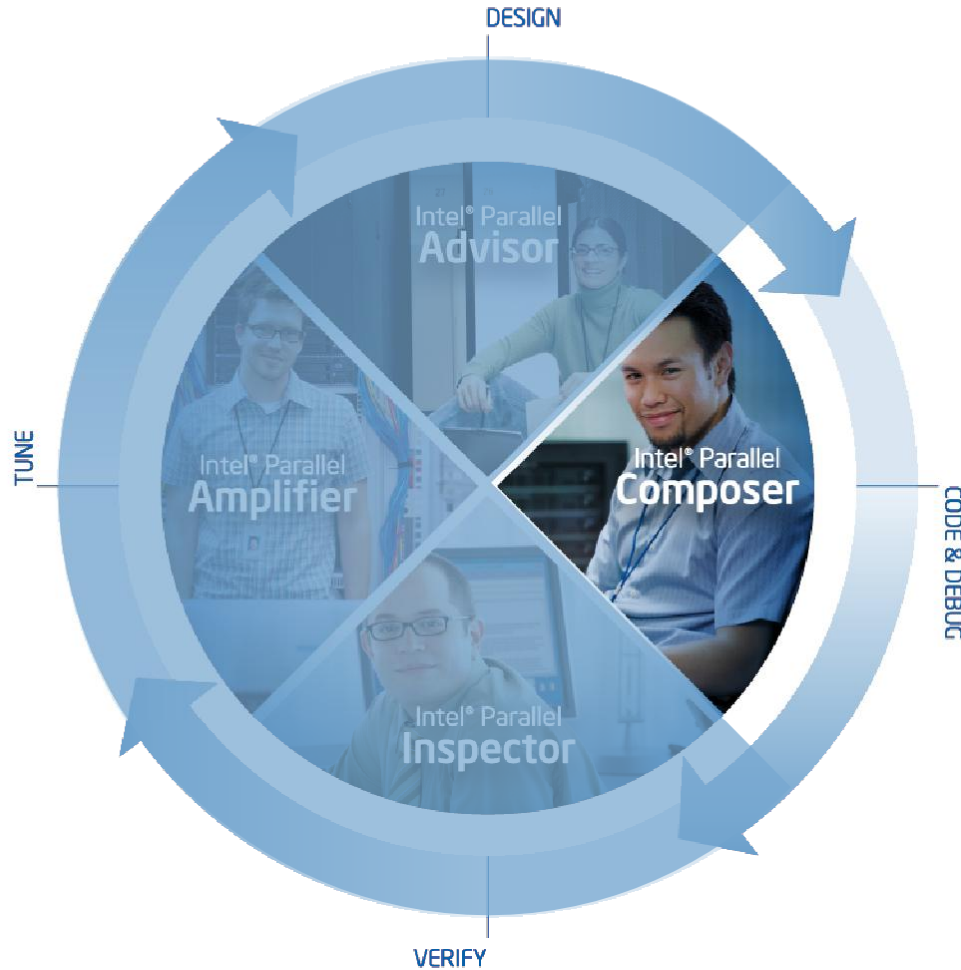
Intel Parallel Advisor

Планируется в
2010

Intel® Parallel Advisor

В каком месте лучше распараллелить приложение?

- Новая категория инструментов разработчика
- Определяет в каком месте лучше распараллелить имеющееся приложение
 - Где распараллеливать
 - Определяет, в каком месте распараллелить код для достижения максимального эффекта при минимальных затратах
 - Как распараллелить
 - Обнаруживает конфликты и рекомендует решения
 - Дает оценки возможного выигрыша по производительности
 - Помощь в написании программы
 - Предоставляет программные сэмплы с использованием высокопроизводительных параллельных библиотек Intel® Threading Building Blocks и OpenMP*



Intel® Parallel Composer

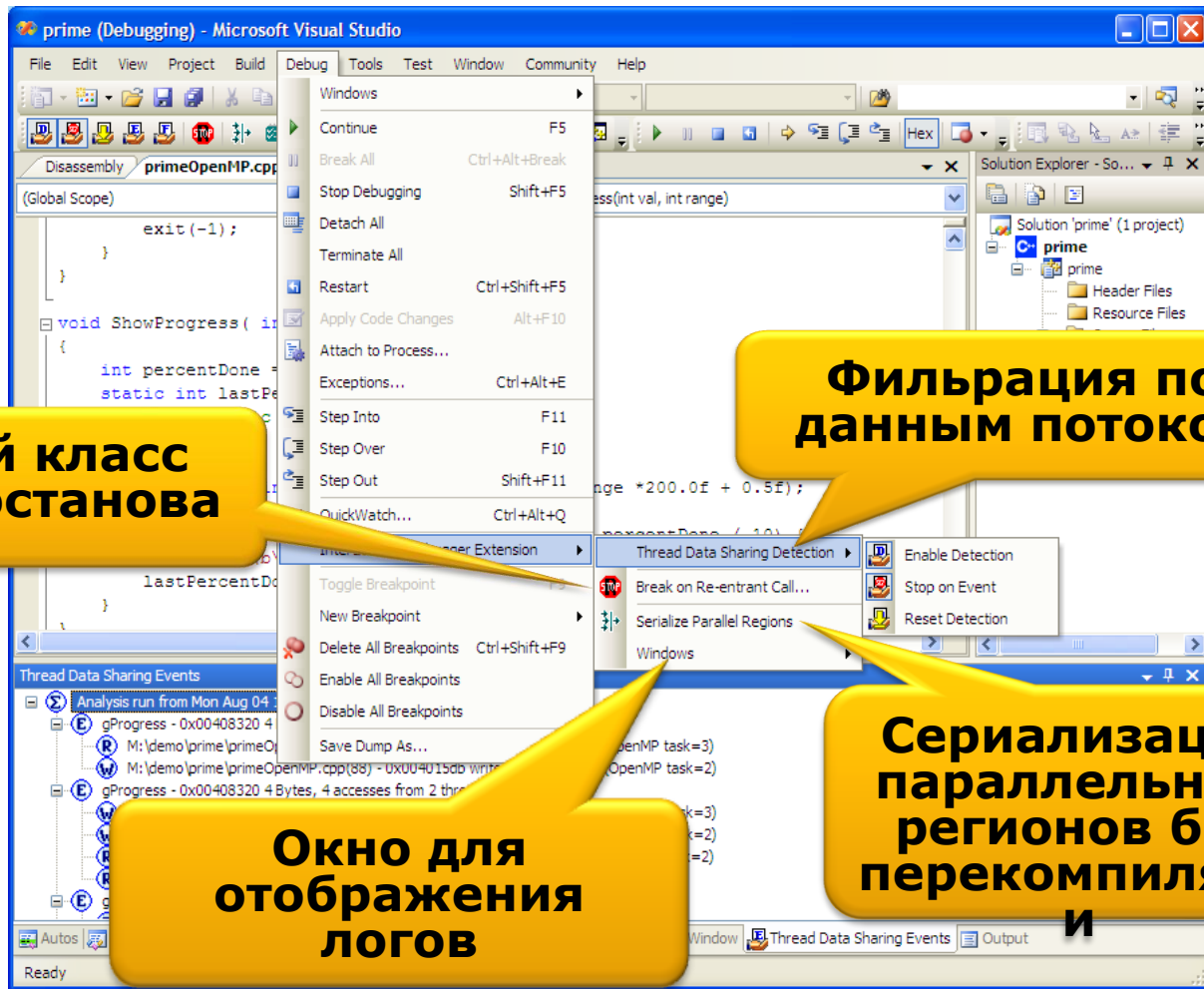
Intel® Parallel Composer

Ускорение разработки параллельных приложений с компилятором C/C++ и полным набором библиотек

- Упрощает распараллеливание увеличивая продуктивность разработчика
 - “Think Parallel” и кодируй без низкоуровневого управления потоками
- Разработчикам под Microsoft* Visual Studio* теперь еще легче создавать параллельные приложения
- Valarray & Intel® Integrated Performance Primitives
- Intel® TBB and Lambda Functions
- Четыре преимущества Parallel Composer
 1. Легкость распараллеливания
 - Параллельные библиотеки и встроенные возможности компилятора уменьшают размер кода и увеличивают масштабируемость
 2. Лучшая производительность на Windows*
 - Оптимизирующий компилятор и библиотеки позволяют получить максимальную производительность от мультиядерных процессоров
 3. Интеграция
 - Поддерживает различные компиляторы и методы распараллеливания
 - Полностью интегрирован в Microsoft Visual Studio
 4. Является частью Intel® Parallel Studio
 - Полный набор для разработки, отладки и оптимизации параллельных приложений

Intel® Parallel Composer

Встроенный параллельный отладчик

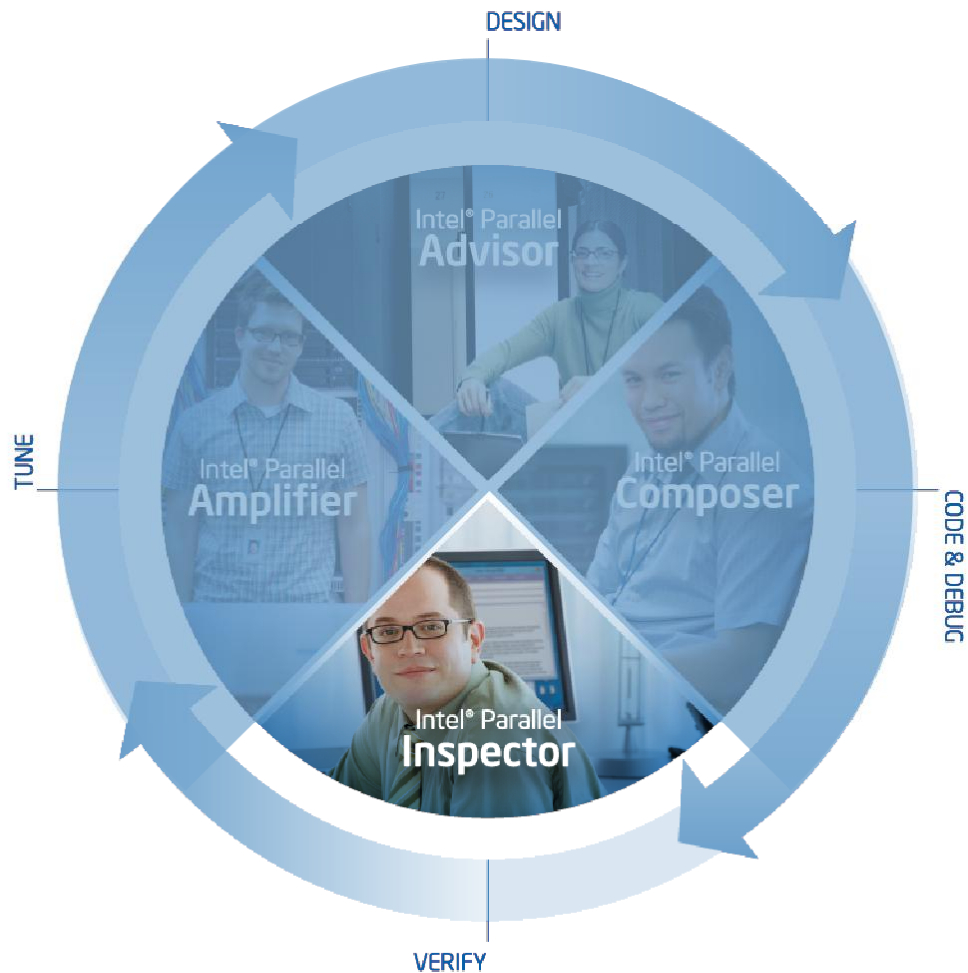


Новый класс точек останова

Фильтрация по данным потоков

Окно для отображения логов

Сериализация параллельных регионов без перекомпиляции



Intel® Parallel Inspector

Intel® Parallel Inspector

Обнаруживает ошибки многопоточности и доступа к памяти

- Обнаружение ошибок доступа к памяти
 - Утечки памяти, например, ошибки аллокации
 - Порча данных, ведущая к падениям программы, и т.д.
- Обнаружение ошибок многопоточности
 - «Гонки» (Data race)
 - Взаимоблокировки (Deadlocks)
- Диагностика многопоточного приложения
 - Находит скрытые ошибки и ставит им с воответствие строки исходного кода, стек вызовов, и адресацию рамяти
 - Отображает сообщения диагностики и предупреждения о возможных ошибках
- Не требуется специальные опции компиляции

Улучшите надежность многопоточного приложения найдя все ошибки многопоточности и доступа к памяти еще до того, как продукт попадет к вашим заказчикам

Intel® Parallel Inspector

Обнаружит ошибочный код

The screenshot displays the Intel Parallel Inspector interface. The main window shows a list of observations under the 'Observations' tab. The table below summarizes the data shown in the screenshot:

Se.	Description	Problem	Source	Function	Module	Object
X1	Allocation site	Memory leak	trace.tbb.cpp:158	operator()	tachyon.tbb.exe	540648
X2	Allocation site	Memory leak	trace.tbb.cpp:158	operator()	tachyon.tbb.exe	7148568
X3	Allocation site	Memory leak	parallel_for.h:71	run	tachyon.tbb.exe	32
X4	Allocation site	Memory leak	parallel_for.h:92	execute	tachyon.tbb.exe	8
X5	Allocation site	Memory leak	parallel_for.h:92	execute	tachyon.tbb.exe	28
X6	Allocation site	Memory leak	parallel_for.h:96	execute	tachyon.tbb.exe	448
X7	Allocation site	Memory leak	parallel_for.h:96	execute	tachyon.tbb.exe	224
X9	Allocation site	Memory leak	task_scheduler_init.h:73	task_scheduler_init	tachyon.tbb.exe	1280
X10	Allocation site	Memory leak	task_scheduler_init.h:73	task_scheduler_init	tachyon.tbb.exe	352
X13	Allocation site	Memory leak	task_scheduler_init.h:73	task_scheduler_init	tachyon.tbb.exe	384
X16	Allocation site	Memory leak			tachyon.tbb.exe	384
X18	Allocation site	Memory leak			tachyon.tbb.exe	16

The right-hand pane shows a summary of the selected 'Memory leak' problem, categorized by Source, Function, and Module. The 'Module' section shows 'tachyon.tbb.exe' with 12 items.

The bottom pane shows the source code for the selected observation (X2) at line 158:

```
156 unsigned int serial = ...
157 unsigned int mboxsize = sizeof(unsigned int)*(max_objectid() + 20);
158 unsigned int * local_mbox = (unsigned int *) malloc(mboxsize);
159 memset(local_mbox, 0, mboxsize);
```

Утечка прамяти:
где и сколько

Observations
позволяет определить
проблему

Сконцентрируйтесь на вашем
собственном коде

Intel® Parallel Inspector

Найдет место, где была выделена память

Быстрое определение адреса аллокации памяти

Как я сюда попал?

```
Focused observation: trace.tbb.cpp:158 - Allocation site
153 void operator() (const tbb::blocked_range2d<int> &r) const
154 {
155     // task-local storage
156     unsigned int serial = 1;
157     unsigned int mboxsize = sizeof(unsigned int)*(max_objectid() + 20);
158     unsigned int * local_mbox = (unsigned int *) malloc(mboxsize);
159     memset(local_mbox,0,mboxsize);
160     if(video->next_frame
161     {
162         drawing_area cols().begin(), totaly-r.rows().end(), r.cols()
163         in(); y != r.rows().end(); ++y, i++) {
164             size_y-i);
165             ); x != r.cols().end(); x++) {
166                 ixel (x, y, local_mbox, serial, startx
167             }
168     }
169 }
```

Obs..	Description	Source	Function	Module	Obj...
X2	Allocation site	trace.tbb.cpp:158	operator()	tachyon.tbb.exe	714...

Observation Relationship

- Allocation site trace.tbb.cpp:158

Intel® Parallel Inspector

Определяет ошибки многопоточности и укажет место в исходном коде

The screenshot shows the Intel Parallel Inspector interface. At the top, there are tabs for 'Problem Sets', 'Sources and Stacks', and 'Observations'. The 'Observations' tab is active, displaying a table of detected issues. A yellow callout bubble points to the 'Observations' table with the text: 'Тип ошибки многопоточности и ее местонахождение' (Type of multithreading error and its location). Below the table, the 'Observations in Problem Set: Data race' section is expanded for observation X10, showing the source code for the 'set_ref_count' function in 'tachyon.tbb.exe'. A second yellow callout bubble points to the source code with the text: 'Быстрая ссылка на исходный код' (Fast link to the source code).

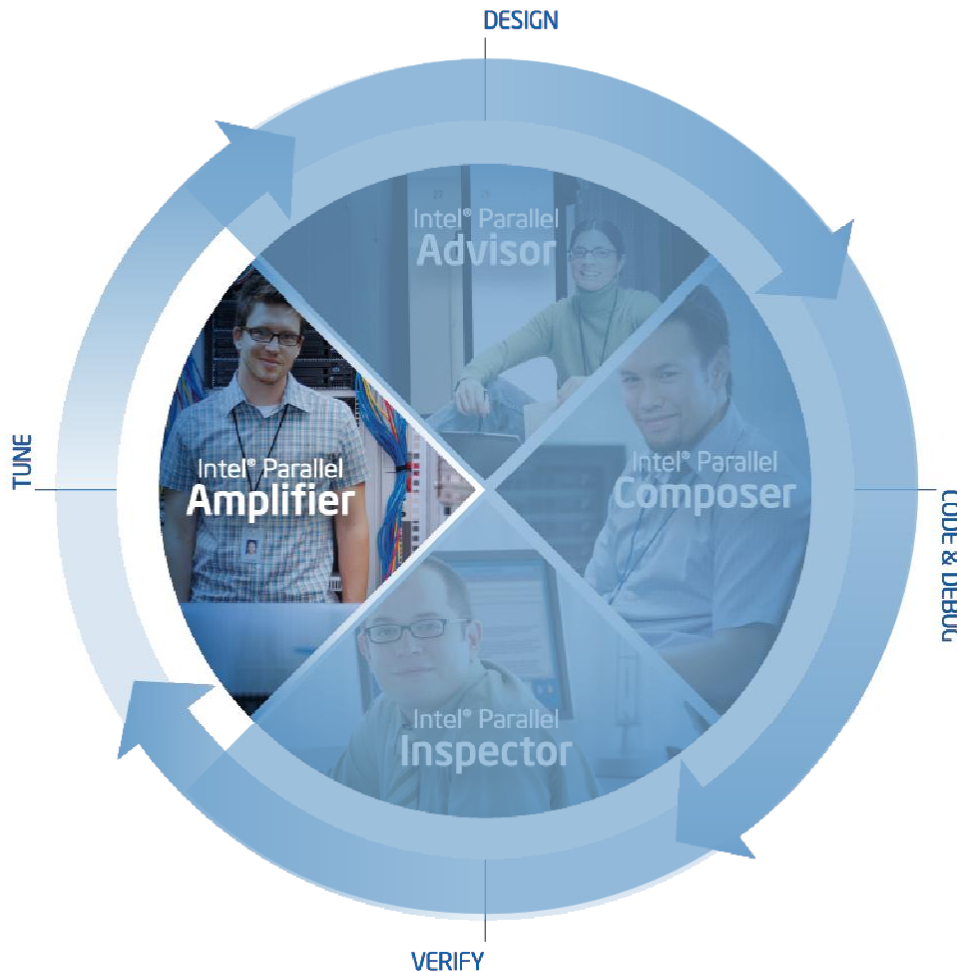
Q	Se...	Description	Problem	Module
X2	i	Creation site	Analyzed thread	tachyon.tbb. ...
X3	i	Creation site	Analyzed thread	tachyon.tbb. ...
X4	i	Creation site	Analyzed thread	tachyon.tbb. ...
X5	⊗	Unprotected read	Data race	tachyon.tbb. ...
X6	⊗	Unprotected write	Data race	tachyon.tbb. ...
X7	⊗	Unprotected read	Data race	tachyon.tbb. ...
X8	⊗	Unprotected read	Data race	tachyon.tbb. ...
X9	⊗	Unprotected write	Data race	tachyon.tbb. ...
X10	⊗	*unused*	Data race	tachyon.tbb. ...

Observat...	Description	Source	Function	Module
X10	*unused*	task.h:475	set_ref_count	tachyon.tbb.exe

```
473 void set_ref_count( int count ) {
474 #if TBB_DO_ASSERT
475     internal_set_ref_count(count);
476 #else
477     prefix().ref_count = count;
```


X5	Unprotected read	winvideo.h:252	next_frame	tachyon.tbb.exe
----	------------------	----------------	------------	-----------------

```
250 {
251     if(!running) return false;
252     g_updates++;
```



Intel® Parallel Amplifier

Intel® Parallel Amplifier

Находит узкие места в производительности многопоточного приложения

- **Hotspot** анализ

- На что моя программа тратит вычислительное время?
 - Определит Hotspot-функции
 - А так же путь к ним

- **Concurrency** анализ

- Почему моя программа плохо параллелится?
 - Определяет эффективен ли параллельный код
 - Демонстрирует использование процессорных ядер

- **Lock/Wait** анализ

- Где моя программа ожидает синхронизацию или I/O?
 - Определяет проблемы блокирования потоков, замедляющие скорость исполнения программы
 - Идентифицирует объекты ограничивающие масштабируемость

Intel® Parallel Amplifier

Инструменты для эффективного анализа

- Source View
 - Отображает данные производительности в окне с исходным кодом
- Statistical Call Tree
 - Позволяет определить путь вызовов функций, влияющий на общую производительность
- Compare Multiple Runs
 - Легко увидеть, на что повлияли изменения в коде

Сделайте масштабируемую производительность преимуществом вашего приложения на мультитядерных платформах

Intel® Parallel Amplifier

Hotspot анализ помогает найти узкие места

Где моя программа тратит время?
Быстро определит критические функции влияющие на производительность

Function	Module	CPU Time
void video::main_loop(void)	tachyon.tbb.exe	407.185 ms
enum Gdiplus::Status Gdiplus::Graphics::DrawImage(class Gdiplus::Image *,int,int,int,int,int,int)	tachyon.tbb.exe	270.597 ms
Stack InternalWndProc(struct HWND__ *,unsigned int,unsigned int,long) <	tachyon.tbb.exe	250.000 ms
Stack InternalWndProc(struct HWND__ *,unsigned int,unsigned int,long) <	tachyon.tbb.exe	20.597 ms
GetVect	tachyon.tbb.exe	107.403 ms
long Int	tachyon.tbb.exe	67.368 ms
stringc	tachyon.tbb.exe	10.750 ms
GetObj	tachyon.tbb.exe	10.747 ms
strlen	tachyon.tbb.exe	10.743 ms
void *	tachyon.tbb.exe	10.731 ms
	tachyon.tbb.exe	5.877 ms
	tachyon.tbb.exe	4.851 ms

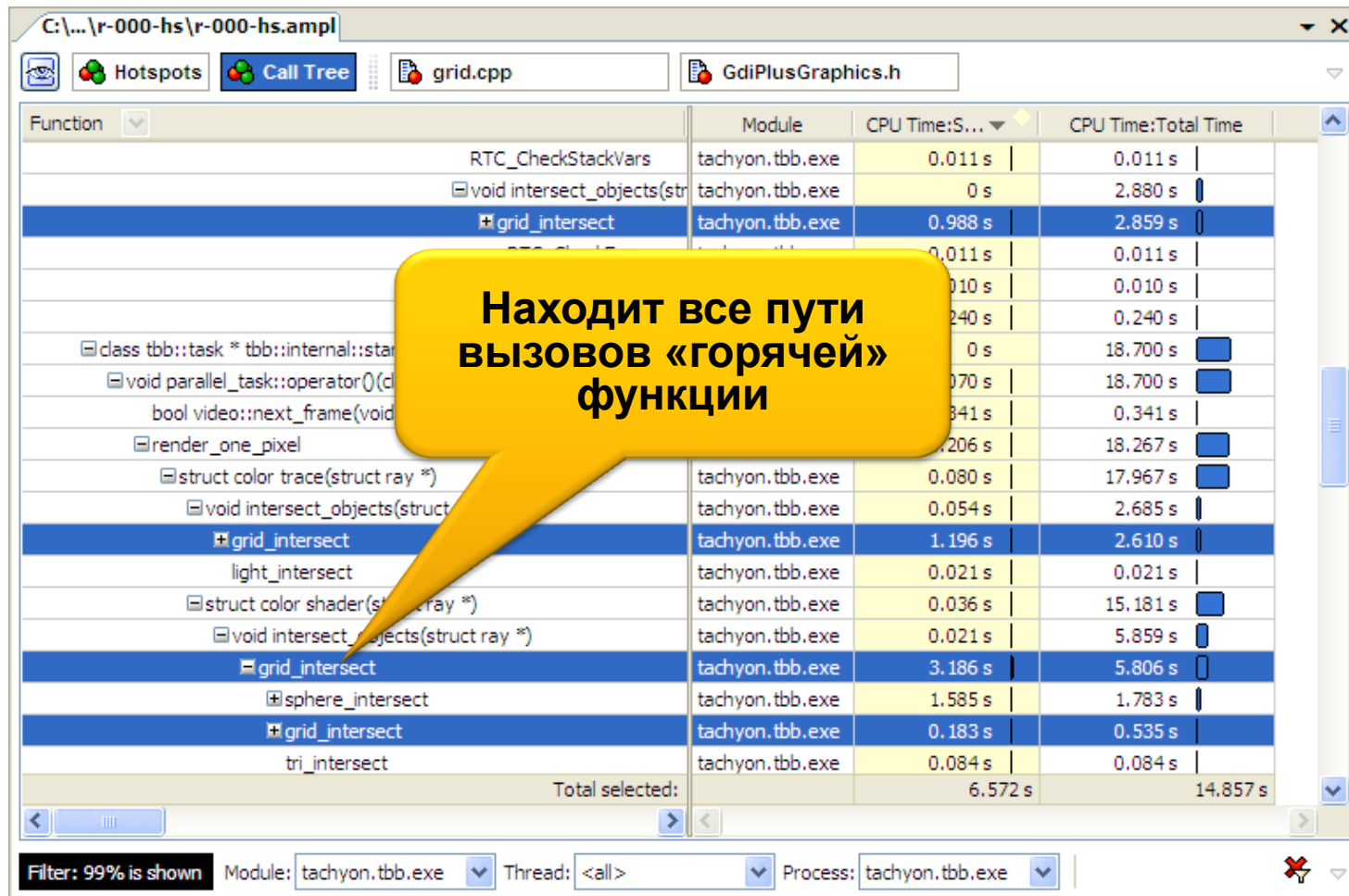
Как я сюда попал?
Стек вызовов поможет понять, где критический путь

Total selected: 270.597 ms

Elapsed Time: 23.398 s

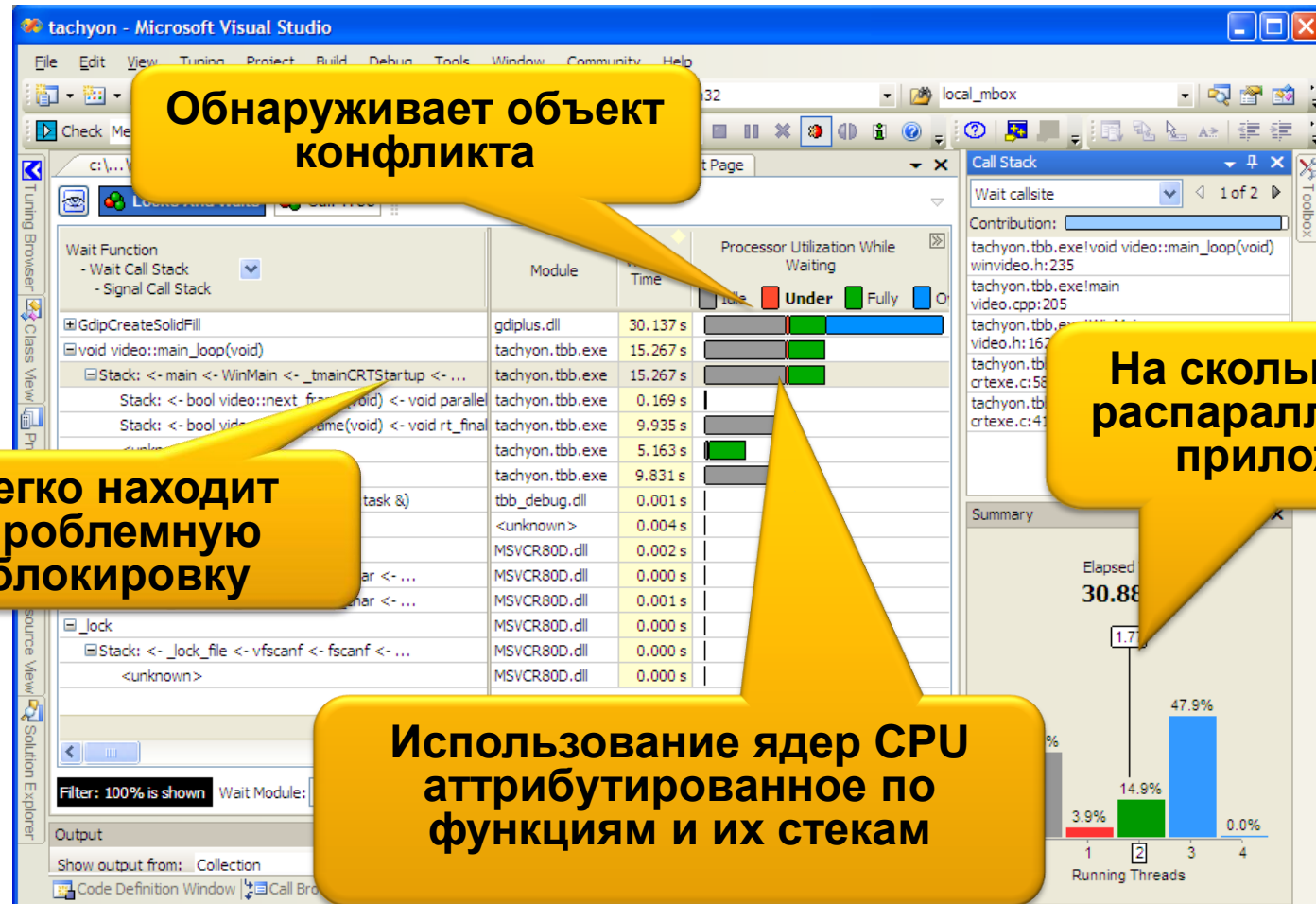
Intel® Parallel Amplifier

Statistical Call Tree – путь вызовов функций, влияющий на производительность



Intel® Parallel Amplifier

Профилировка блокировок – находит конфликты доступа ограничивающие производительность и Concurrency анализ – отображает эффективность использование ядер



Обнаруживает объект конфликта

Легко находит проблемную блокировку

Использование ядер CPU атрибутированное по функциям и их стекам

На сколько хорошо распараллелено мое приложение?

www.intel.com/go/parallel

The screenshot shows a web browser window titled "Intel IDF" with the URL http://barracuda-ext.cmdpdx.com/dev/1996_Intel_IDF/index.html. The page features the Intel Parallel Studio logo and a central image of a man in a blue shirt. The main content includes sections for "Parallelism Breakthrough Welcome Visual Studio developers.", "The Path to Parallel Programming", and "Attention C/C++ and Fortran Developers for Linux* and Mac OS X*". A "SIGN UP NOW" button is visible. Three yellow callout boxes with arrows point to specific elements: "Sign up to be first in line for beta programs" points to the sign-up button; "Current beta program for Linux and Mac" points to the Linux/Mac section; and "Resources for more information" points to the "Explore Parallelism" section.

Intel® Parallel Studio
Intel® Software Solutions

Parallelism Breakthrough
Welcome Visual Studio developers.

Just announced at IDF—Intel provides new tools to scale legacy apps for multicore and manycore.

Intel Parallel Advisor: Spotlights where applications benefit most from parallelism.

Intel Parallel Composer: Speeds build time and accuracy with a compiler, debugger, and comprehensive parallel libraries.

Intel Parallel Inspector: Finds latent bugs within the increasing complexity of parallel threads.

Intel Parallel Amplifier: Fine-tunes application for optimal performance, ensuring cores are fully exploited.

Intel Parallel Studio: Ultimate all-in-one parallelism toolkit.

The Path to Parallel Programming
Sign up now for early notification of BETA.

To qualify for this BETA program you must currently use:

1. Microsoft Visual Studio* 2005 or 2008 (excluding Express)
2. C/C++, and
3. Microsoft Windows XP* or Vista*.

[SIGN UP NOW](#)

Attention C/C++ and Fortran Developers for Linux* and Mac OS X*

The Intel® CompilerPro 11.0 BETA is closing soon.
[Click here to register.](#)

Explore Parallelism

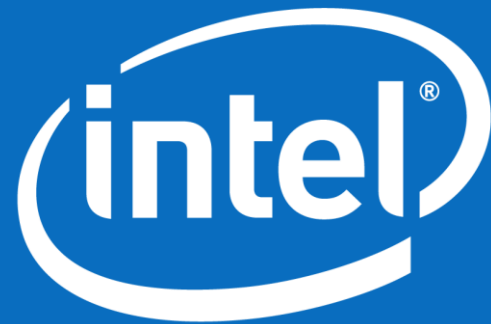
[Intel Parallel Studio FAQ >](#)
[www.go-parallel.com >](#)
[www.intel.com/software/products >](#)

© 2008, Intel Corporation. All rights reserved. Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and other countries.
*Other names and brands may be claimed as the property of others. Privacy

Sign up to be first in line for beta programs

Current beta program for Linux and Mac

Resources for more information



www.intel.com/software/products