Agenda

Software at Intel

Major Technological Change

Software Response

Parallel Programming 2.0
Why Does Intel Care About Software?
All Architectures Supported Across All Environments

Other names and brands may be claimed as the property of others.
Enabling...

- Intel® Software Network
- Intel® Software College
- Intel® Early Access Program
- Industry Programs and Alliances
- Intel® Competency Centers
- Intel® Software Research
- Intel® Press Publications

www.intel.com/software
Welcome to the Intel® Software Network.

Top Stories

Intel-based Apple Computers
See how Intel® Core™ Duo processor and Intel software tools support these new products. [ Intel® Software Development Products ]

Develop for Intel® Viiv™ Technology-based PCs
Maximize your application or service performance by architecting for multithreading or multi-tasking environments enabled by dual-core processors. [ Digital Media ]

Intel Vision for Digital Home
Download the Intel® Software and Services Product Recommendations for 2006. Get the details you need to start developing for the future digital home. [ Digital Home ]

Simplify Development of Threaded Applications

With Worldwide Reach

Over 50 Development Sites
Across More Than 20 Countries

Scaling Innovation on Intel’s Leading Platforms and Tapping the Best Talent Around the World
Hardware
+ Platform Software
+ ISV Value Add

Platform Solutions
Platform Software

- Intel® Active Management and Intel® Virtualization Technology Solutions
  - Rapid adoption throughout ecosystem
  - Many companies bringing products to market

- Carrier Grade Linux for Telecommunications with AdvancedTCA
  - Both hardware and software technology building blocks available
  - Commercial deployments already underway worldwide
Platform Software

- Intel® Platform Administration Technology
  - For internet café’s and SMB installations

- Tiano / EFI
  - Apple using Tiano and EFI to boot Intel-based Macs
  - Embedded Tiano to replace PC BIOS for initialization and management of high performance clusters
XML Traffic Will Exceed:

- All Mail Traffic in 2004
- All Web Traffic in 2007

Source: Zapthink: Solving the very large messaging problem in the enterprise, February 2005
Managed Runtime

- Develop the best available Java and .NET technology for Intel® hardware
  - help JVMs run better on parallel hardware
  - JITs generate great code for Intel® microarchitectures
- Example projects
  - Apache Harmony open-source JVM. Goals:
    - Create an open-source, compatible implementation of J2SE 5 under the Apache License
    - Create a community to carry the open J2SE forward
  - NuMA-cc Awareness for JVMs; object co-location
  - Profile collection and management
  - Dynamic profile-guided optimization
Welcome to Apache Harmony, the J2SE project of the Apache Software Foundation. Please help us make this a world class, certified J2SE implementation!

Note: Apache Harmony is an effort undergoing incubation at the Apache Software Foundation (ASF). Incubation is required of all newly accepted projects until a further review indicates that the infrastructure, communications, and decision making process have stabilized in a manner consistent with other successful ASF projects. While incubation status is not necessarily a reflection of the completeness or stability of the code, it does indicate that the project has yet to be fully endorsed by the ASF.

The aim of the project is to produce a large and healthy community of those interested in runtime platforms tasked with:

- Create a compatible, independent implementation of J2SE 5 under the Apache License v2
- Create a community-developed modular runtime (VM and class library) architecture to allow independent implementations to share runtime components, and allow independent innovation in runtime components
Software Technology Innovation

- concept
- proof of concept
- technology incubation

new products
cur products
open source
enabling
standards
consulting
others

Research Intern Program

Internal Projects as well as Joint Research with Universities
Agenda

Software at Intel

Major Technological Change

Software Response

Parallel Programming 2.0
Quiz

Moore’s law states which of the following roughly doubles every 2 years?

1. Frequency
2. Performance
3. Transistors
4. Transistor Density
Historical Driving Force

Increased Performance via Increased Frequency

- **1971**: I4004 Processor
  - 2300 Transistors
- **1946**: 20 Numbers in Main Memory
- **2005**: 65nm
  - 1B+ Transistors

Frequency (MHz)

- 1970
- 1980
- 1990
- 2000
- 2010
- 2020

2005 65nm 1B+ Transistors
The Challenge

Power Limitations

Power = Capacitance x Voltage\(^2\) x Frequency
also
Power \sim Voltage^3
Energy: The Next Frontier
Energy Efficient Performance – High End

**NASA Columbia**
- 2 MWatt
- 60 TFlops goal
- 10,240 cpus – Itanium II
- $50M

**ASC Purple**
- 6 MWatt
- 100 TFlops goal
- 12K+ cpus – Power5
- $230M

**Computational Efficiency**
- 30,720 Flops/Watt
- 1,288 Flops/Dollar
- 7,066 Flops/Watt
- 467 Flops/Dollar

Source: LLNL
Source: NASA
The Classic Tradeoff

Higher
Top Speed and Acceleration

OR

Increased
Range and Economy
A Simple Example

- Performance
- Power

Max Frequency

1.00x
Over-clocking

- Over-clocked (+20%): 1.73x
- Max Frequency: 1.00x

- Performance
- Power
Under-clocking

- **Over-clocking (+20%)**
  - Performance: 1.73x
  - Power: 1.13x

- **Max Frequency**
  - Performance: 1.00x
  - Power: 0.87x

- **Under-clocking (-20%)**
  - Performance: 0.51x
  - Power: 0.87x
Multi-Core
Energy-Efficient Performance

- Over-clocked (+20%)
  - Dual-Core: 1.73x
  - Performance: 1.13x
  - Power: 1.00x

- Dual-core (-20%)
  - Dual-Core: 1.73x
  - Performance: 1.02x
  - Power: 1.00x
Moore’s Law will provide transistors

Intel process technology capabilities

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Feature Size</td>
<td>90nm</td>
<td>65nm</td>
<td>45nm</td>
<td>32nm</td>
<td>22nm</td>
<td>16nm</td>
<td>11nm</td>
<td>8nm</td>
</tr>
<tr>
<td>Integration Capacity</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>16</td>
<td>32</td>
<td>64</td>
<td>128</td>
<td>256</td>
</tr>
</tbody>
</table>

Use transistors for

- Multiple cores
- On-core memory (caches)
- New features (*Ts)

Multiple cores and caches address power and memory latency issues
The Dawn of Energy-Efficient Performance
Agenda

Software at Intel

Major Technological Change

Software Response

Parallel Programming 2.0
Multi-Core Platforms Demand Threaded Software

Biggest Performance Leap Since Out-of-Order Execution

Integer Performance at Introduction (normalized to 25MHz 486DX)

- Single Threaded
- Multi Threaded

Pentium  Pentium II  Pentium III  Pentium 4  Pentium D  Conroe

Copyright © 2006, Intel Corporation
The Importance of Threading

• Do Nothing: Benefits Still Visible
  – Operating systems ready for multi-processing
  – Background tasks benefit from more compute resources

• Parallelize: Unlock the Potential
  – Threaded applications
  – Threaded libraries
  – Compiler generated threads
Unleash Multi-Core Potential

- **Intel® C++ and Fortran Compilers**
  - Built-in threading support with Auto-Parallelization and OpenMP® support
- **Intel® Thread Checker & Thread Profiler**
  - Unique product locates hard to find threading errors before they happen!
  - Helps developers optimize threaded applications
- **Intel® MKL and IPP Performance Libraries**
  - Highly optimized threaded libraries that enable multi-core performance gains even if your application isn’t threaded!
- **Intel® VTune™ Performance Analyzer**
  - Identifies performance bottlenecks in single or multithreaded applications to maximize performance
Example: Threading for Multi-Core

- Architectural Analysis
- Introducing Threads
- Debugging
- Performance Tuning
Introducing Threads

- Functional Structure
- Execution Times
- Counts

Intel® VTune™ Performance Analyzer

Architectural Analysis

Introducing Threads

Debugging

Performance Tuning

Call Graph
Threading for Multi-Core

Architectural Analysis

Introducing Threads

Debugging

Performance Tuning

OpenMP Loop Construct
- Creates one thread per core
- Assigns iterations to threads

Intel® Compilers
U32 uElapsedTime = uCurrentTime - uStartTime;

for (i = 0; i < 11; i++)
    IFXRESULT = pTable[i++] = uPalindrome;
    const U32 tid = IFXGetThreadId();
    i32 indX;

#pragma omp parallel for schedule(runtime)
for (индекс = 0; индекс < 11; индекс++)
{
    U32 uPaletteIndex = pTable[indX];
    i32 indX;
}

#pragma omp parallel for schedule(runtime)
for (индекс = 0; индекс < 11; индекс++)
{
    U32 uPaletteIndex = pTable[indX];
    // For each decoder in the component chain referenced by a palette entry,
    IDXDECLLOCAL(IXFDecoderChainX, pdecoderChainX);
    IFXRESULT = (m_ppDecoderPalettes[i] -> GetResourcePtr(uPaletteIndex, IID_IFXDecoderChainX, (void**) &pDecoderChainX));

    U32 uDecoderCount = 0;
    pDecoderChainX -> GetDecoderCountX(uDecoderCount);

    // For the next decoder palette entry.
    i32 j;
    for (j = 0; j < uDecoderCount; j++)
    {
        IFXRESULT = (pDecoderChainX -> GetDecoderX(j, pDecoderX));
        if (pDecoderX) {
            // Perform idling activities.
        #ifdef TLP_IMPORT_FRONT
        if (tid == IFXGetThreadId())
            #endif
            ThumpX();
            IFXRESULT = iResultTransfer = IFX_OK;
            pDecoderX -> TransAct(iResultTransfer);
        #ifdef TLP_IMPORT_FRONT
        }
        // If a decoder has transferred all of its blocks and the read process has concluded.
Introducing Introducing

Threads

Debugging

Performance Tuning

Thread Safety Issues

• Data Races
• Deadlocks

Intel® Thread Checker

Architectural Analysis

Introducing Threads

Copyright © 2006, Intel Corporation
Threading for Multi-Core

Intel® Thread Profiler

Find Contended Locks
- Most Overhead
- Largest Reduction in Parallelism
Performance Impact

Weather forecast application

Legacy system
8 CPUs

Original source, default options,
Itanium® 2 4 CPU

/O3 – Enable advanced optimizations
Itanium® 2 4 CPU

Compiler directives, minor source changes
Itanium® 2 4 CPU

Adding OpenMP,
Itanium® 2 4 CPUs

Compiler

10x

1x

1.2x

2.5x

4x

Optimizations may improve performance significantly.
Agenda

Software at Intel
Major Technological Change
Software Response
Parallel Programming 2.0
A New Era...

THE OLD

Performance Equals Frequency
Unconstrained Power
Voltage Scaling

THE NEW

Performance Equals IPC
Multi-Core
Power Efficiency
Microarchitecture Advancements

And it is happening fast...
Multi-Core Trajectory

- Dual-Core (2005/2006)
- Quad-Core (2007)
Growing Momentum for Multi-Cores

- Power delivery and management
- High bandwidth memory
- Cache
- Cores
<table>
<thead>
<tr>
<th>Activision (Ravensoft)</th>
<th>Pinnacle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adobe</td>
<td>Pixar (Renderman)</td>
</tr>
<tr>
<td>Algorithmics</td>
<td>Paradigm</td>
</tr>
<tr>
<td>Alias</td>
<td>PTC</td>
</tr>
<tr>
<td>Autodesk</td>
<td>Red Hat</td>
</tr>
<tr>
<td>Business Objects</td>
<td>SAP</td>
</tr>
<tr>
<td>Cakewalk</td>
<td>SAS</td>
</tr>
<tr>
<td>CodecPeople</td>
<td>Siebel CRM</td>
</tr>
<tr>
<td>Computer Associates</td>
<td>Signet</td>
</tr>
<tr>
<td>Corel (WordPerfect)</td>
<td>Skype</td>
</tr>
<tr>
<td>Cyberlink</td>
<td>SLB</td>
</tr>
<tr>
<td>Discreet</td>
<td>SnapStream</td>
</tr>
<tr>
<td>IBM</td>
<td>Sonic (Roxio)</td>
</tr>
<tr>
<td>id Software</td>
<td>Sony</td>
</tr>
<tr>
<td>Landmark</td>
<td>Steinberg</td>
</tr>
<tr>
<td>Macromedia</td>
<td>SunGard</td>
</tr>
<tr>
<td>Mainconcept</td>
<td>Sybase</td>
</tr>
<tr>
<td>Maxon</td>
<td>Symantec</td>
</tr>
<tr>
<td>mental images</td>
<td>Thomson</td>
</tr>
<tr>
<td>Microsoft (Office Suite)</td>
<td>THQ</td>
</tr>
<tr>
<td>Midway</td>
<td>Ubisoft</td>
</tr>
<tr>
<td>MSC</td>
<td>UGS</td>
</tr>
<tr>
<td>Novell SUSE</td>
<td>Valve</td>
</tr>
<tr>
<td>Oracle</td>
<td>Valve</td>
</tr>
<tr>
<td>Pegasus</td>
<td>Yahoo (Musicmatch)</td>
</tr>
</tbody>
</table>
Growing Momentum in Ecosystem Driven by Intel

- Continuous tool improvements
  - Over 15 tools released for multi-core in ‘05
  - Coming in ‘06: more features across all environments

- Training end users and in house developers
  - 2005: > 1500 students  2006 target:  4500 students

- Academia
  - Universities developing parallel programming curriculum
  - Research grants for parallel programming projects

- Contest: Over $100,000 in prizes
  - Topcoder.com: Monthly contests for top performing threaded software
  - Games Developer Conference: Contest for best use of Intel platform features
Opportunity for Software: Extract Full Potential of Multi-core

Parallel Programming 1.0

Parallel Programming 2.0
Parallel Programming 2.0

• Ubiquitous parallel software
  – Large spectrum of domains (consumer/wireless vs HPC/database, home vs nuclear labs)

• Scalable software
  – Explosion of cores (e.g. 2X cores every 18-24 months)

• User experience
  – vs. just raw performance

• Greater ease of programming
  – New programming paradigm, programming language, compiler, tools,

• Greater reliability and security
  – Programming for application and system reliability

• Industry-wide vs government-funded

• Higher demand for parallel programming education
  – Mass vs elite
Imagine what can be
Create what will be

Parallel Programming 2.0

The Beginning of a New Era