Debugging and Optimizing Scalable Applications on the Cray

Chris Gottbrath, Principal Product Manager
May 2\textsuperscript{nd}, 2012
Agenda

• Who are Rogue Wave?
• High Performance Computing Challenges
• Debugging: TotalView Updated for the Cray Platform
• Optimization: Introduction to ThreadSpotter
• Future Development Tool Needs
Rogue Wave Today

- **History**
  - Founded: 1989
  - Acquired by Battery Ventures: 2007
  - Acquired:
    - Visual Numerics: 2009
    - TotalView Technologies: 2010
    - Acumem: 2010

- **Customers**
  - 3,000+ in 36 countries
  - Financial services, telecoms, oil and gas, government and aerospace, research and academic

- **Pioneers in C++/object-oriented development**
- **Leading the way in cross-platform, parallel development**
Global Reach

40 years -- 3000+ customers -- 36 countries

| OR  | MA | CO (HQ) | TX | UK | Sweden | Germany | Japan |
Representative Customers
TotalView and Cray give developers the tools they need for application development.
Agenda

• Who are Rogue Wave?
• High Performance Computing Challenges
  • Debugging: TotalView Updated for the Cray Platform
  • Optimization: Introduction to ThreadSpotter
• Future Development Tool Needs
Challenges of HPC Software Development

• **Large Scale Clusters with Many-Core**
  – Massive number of total cores
  – Hybrid (MPI + Threading)
  – Balance (memory per core, bandwidth per core)

• **Accelerators**
  – Fundamentally heterogeneous architecture

• **Bandwidth bottlenecks**

• **Rapid development in paradigms, languages and runtimes**

• **Code complexity**
  – Collaboration
  – Software engineering and maintainability

• **Rising cost of system operations (power, cooling)**

• **Increasing diversity of users**
High Performance Computing

Scalability: A Vicious/Virtuous Cycle
Rogue Wave Solution Portfolio

Enterprise C++ Development

Developing for Parallel Architectures

Parallel, Data-intensive Applications

Leveraging High Performance Data Analytics

SourcePro C++

Visualization

MemoryScape

ThreadSpotter

TotalView

IMSL

PV-WAVE
Agenda

- Who are Rogue Wave?
- High Performance Computing Challenges
- Debugging: TotalView Updated for the Cray Platform
- Optimization: Introduction to ThreadSpotter
- Future Development Tool Needs
What is TotalView?

• Application Analysis and Debugging Tool: Code Confidently
  - Debug and Analyze C/C++ and Fortran on Linux, Unix or Mac OS X
  - Laptops to supercomputers (BG, Cray)
  - Makes developing, maintaining and supporting critical apps easier and less risky

• Major Features
  - Easy to learn graphical user interface with data visualization
  - Parallel Debugging
    - MPI, Pthreads, OpenMP, GA, UPC
    - CUDA and OpenACC (early access)
  - Includes a Remote Display Client freeing you to work from anywhere
  - Memory Debugging with MemoryScape
  - Deterministic Replay Capability Included on Linux/x86-64
  - Non-interactive Batch Debugging with TVScript and the CLI
  - TTF & C++View to transform user defined objects
What Is MemoryScape?

- **Runtime Memory Analysis : Eliminate Memory Errors**
  - Detects memory leaks *before* they are a problem
  - Explore heap memory usage with powerful analytical tools
  - Use for validation as part of a quality software development process

- **Major Features**
  - Included in TotalView, or Standalone
  - Detects
    - Malloc API misuse
    - Memory leaks
    - Buffer overflows
  - Supports
    - C, C++, Fortran
    - Linux, Unix, and Mac OS X
    - MPI, pthreads, OMP, and remote apps
  - Low runtime overhead
  - Easy to use
    - Works with vendor libraries
    - No recompilation or instrumentation
Deterministic Replay Debugging

- Reverse Debugging: Radically simplify your debugging
  - Captures and Deterministically Replays Execution
    - Not just “checkpoint and restart”
  - Eliminate the Restart Cycle and Hard-to-Reproduce Bugs
  - Step Back and Forward by Function, Line, or Instruction

- Specifications
  - A feature included in TotalView on Linux x86 and x86-64
    - No recompilation or instrumentation
    - Explore data and state in the past just like in a live process, including C++View transformations
  - Replay on Demand: enable it when you want it
  - Supports MPI on Ethernet, Infiniband, Cray XE Gemini
  - Supports Pthreads, and OpenMP
New Capabilities in TotalView 8.10

• CUDA 4.1

• Visual Dive Indicator

• Cray-specific enhancements
  – Improved Cray Compiler Edition Support
  – Replay on Cray XE
  – CUDA support on Cray XK
  – Early Access Preview for OpenACC on the Cray with CCE 8

• Reverse Debugging
  – Replay on Demand
  – C++View and ReplayEngine interoperability

• TVScript Scalability Improvements
Coming Soon

• Massive Scalability
  — Collaboration with LLNL and Tri-lab partners
  — Targeting the Cielo, Sequioa and Linux Clusters

• Shiny new GUI
  — Sleek, Modern and Fast
  — Configurable
  — Improved Usability
  — Provides aggregation capabilities for big data and scale
  — Leveraging math and stat expertise from IMSL

• Intel MIC
  — Demo at ISC

• Talk to us about participation in early access programs
Agenda

• Who are Rogue Wave?
• High Performance Computing Challenges
• Debugging: TotalView Updated for the Cray Platform
  • Optimization: Introduction to ThreadSpotter
• Future Development Tool Needs
How is the silicon used (i7-Ex)?
A rule of thumb

<table>
<thead>
<tr>
<th>Memory system level</th>
<th>Relative latency</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1 cache</td>
<td>1x</td>
</tr>
<tr>
<td>Higher cache levels</td>
<td>10x</td>
</tr>
<tr>
<td>Main memory</td>
<td>100x</td>
</tr>
</tbody>
</table>

Source: AMD, *Michael Wall*
Inefficient Loop Nesting

Explanation
Partially Used Structure

Explanation
Alignment Problem

Explanation
What is ThreadSpotter?

- **Runtime Cache Performance Optimization Tool:** Tune into the Multi-Core Era
  - Realize More of the Performance Offered by Multi/Many-Core Chips
  - Quickly Detects and Prioritizes Issues -- and then Provides Usable Advice!
    - Brings Cache Performance Into Reach for Every Developer
    - Makes Experienced Cache Optimizers Hyper-Efficient

- **Features**
  - Supports Linux x86/x86-64
  - Any compiled code
  - Runtime Analysis
    - Low overhead
  - Cache Modeling
    - Prioritizes Issues
    - Identifies Problem Lines of Code
  - Provides Advice
    - Explanations
    - Examples
    - Detailed statistics (if desired)
Agenda

• Who are Rogue Wave?
• High Performance Computing Challenges
• Debugging: TotalView Updated for the Cray Platform
• Optimization: Introduction to ThreadSpotter
• Future Development Tool Needs
Feedback on debugging and optimization tool priorities

• How key are the following potential enhancements
  – C++ 0x (2011)
  – Co-Array Fortran
  – OpenCL
  – Improved support for comparative debugging
  – Cray Fast Track
  – ATP
  – Thread race condition detection
Thank you!

Rogue Wave Software

Developing parallel, data-intensive applications is hard. We make it easier.

sales@roguewave.com