Maximize Application Performance & Reliability with Oracle Solaris Studio

Vijay Tatkar
Senior Manager, Sun Developer Tools
Oracle Solaris 11 Developer Webinar Series

Join us on April 34th for a complimentary webinar. In just one hour you will learn how to streamline your development process to maximize the performance of your application with Oracle Solaris 11. Access the slides here.

- Learn why there is no more patching!
- Understand Oracle Solaris 11 cloud features
- Review how to maximize application reliability in Oracle Solaris 11
- Live Chat with Oracle Solaris 11 Engineers

Who should attend?

Application developers and administrators wanting a deep-dive on key features of Oracle Solaris 11 which you can exploit to make your applications superior to your competitors and easier to use.

Please note: after registering you will receive details on how to attend the meeting in a separate confirmation e-mail shortly before the webcast.

Agenda

<table>
<thead>
<tr>
<th>Webinar Series Topic</th>
<th>Date</th>
<th>Speaker</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modern Software Packaging for Enterprise Developers</td>
<td>03-27-12 9am PT</td>
<td>Eric Reid (Principal Software Engineer)</td>
</tr>
<tr>
<td>Slides here - Replay here</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simplify Your Development Environment with Zones, ZFS &amp; More</td>
<td>04-10-12 9am PT</td>
<td>Eric Reid (Principal Software Engineer) and Stefan Schneider (Chief Technologist ISV-Engineering)</td>
</tr>
<tr>
<td>Slides here - Replay here</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Managing Application Services – Using SMF Manifests in Oracle Solaris 11</td>
<td>04-24-12 9am PT</td>
<td>Matthew Hosanee (Principal Software Engineer)</td>
</tr>
<tr>
<td>Slides here</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimize Your Applications on Oracle Solaris 11: The DTrace Advantage</td>
<td>05-08-12 9am PT</td>
<td>Angelo Rajadurai (Principal Software Engineer)</td>
</tr>
<tr>
<td>Slides here</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximize Application Performance and Reliability on Oracle Solaris 11</td>
<td>05-22-12 9am PT</td>
<td>Vijay Talikar (Senior Manager Software Development)</td>
</tr>
</tbody>
</table>

Register:

A thousand passions, each a lifetime's pursuit ...

Cricket. Religion. Entertainment. Sport?
Oracle Solaris Studio Introduction

#1 Development Toolsuite for Oracle Systems

- **Maximize ROI:** Highest app performance with **up to 3x faster code** on SPARC M-series, T-series, and x86-based systems

- **Improve time-to-deliver:** **Single suite** of advanced tools geared towards leveraging system features to increase productivity and simplify multicore development

- **Protect your application investment:** **Strongest compatibility guarantee** with Oracle Solaris and Solaris Studio
Agenda

- Oracle Solaris Studio Overview
  - Advantages of Studio on Solaris
- Compiler Performance
- Performance Analyzer
- Code Analyzer
- Debugger, Thread Analyzer, IDE, Remote access
Comprehensive All in One Package

Oracle Solaris Studio

**Compiler Suite**
- C, C++ Compilers
  Utilizes advanced code generation technology to optimize apps for highest performance on Oracle’s Sun servers (SPARC & x86).
- Fortran Compiler
  Optimize compute-intensive application performance.
- Debugger
  Ensures app stability with event handling and multi-thread support.
- Performance Library
  Maximizes compute-intensive app performance across SPARC and x86 using advanced numeric solver libraries.

**Analysis Suite**
- Performance Analyzer
  Provides unparalleled insight into your app, allowing you to identify bottlenecks and improve performance by orders of magnitude.
- Code Analyzer
  Ensures app reliability by detecting app vulnerabilities, including memory leaks and memory access violations.
- Thread Analyzer
  Simplifies complex parallel programming errors by detecting hard to pinpoint race and deadlock conditions.

**Integrated Development Environment**
Solaris Studio 12.3 Highlights

Accelerate Performance

- 3x faster code on SPARC T4 than gcc; 40% faster than Sun Studio 12
- 1.5x faster code on Intel x86 than gcc; 20% faster than Sun Studio 12
- 20% faster compile time

Gain Observability

- New Code Analyzer for more reliable applications; reports common coding & memory access errors faster than competitive alternatives
- Enhanced Performance Analyzer with system-wide performance analysis

Improve Productivity

- Remote access to Solaris tools from local desktop (Windows or Mac, Linux, Solaris)
- Streamlined Oracle DB application development
- IPS distribution on Solaris 11 for simplified management and support

Platforms

| Oracle Linux 5 & 6 | Oracle Solaris 10 & 11 | RHEL 5 & 6 |
Engineered with the Oracle Stack

- Builds Oracle Stack: Solaris, Java, Database, PeopleSoft, Siebel, Hyperion, ...
- Best Performance and features on Oracle Systems (SPARC and x86)
- Fully integrated and exploits Solaris APIs
  - Debugging: process control
  - Performance Analyzer: Hardware Counters, Dtrace, signals, …
  - Unmatched Java Profiling: User-, JVM- and machine-level models
  - Significantly better performance with Memory analyzer (discover)
- Strongest Binary Compatibility guarantee (Compilers+OS)
- Well-managed interoperability with third party libraries
- Ease of installation and patching
- Common framework for support
- Closely aligned standardization work (C, C++, SVR4 ABI, POSIX...)
World Record-setting Application Performance

**Solaris on SPARC T4**
- Float Pt. Apps: 217% (100%), 186% (100%)
- Integer Apps: 144% (100%), 121% (100%)

**Solaris on Intel Sandy Bridge**
- Float Pt. Apps: 212% (100%), 107% (100%)
- Integer Apps: 123% (101%), 172% (100%)

*Results based on estimated runs of the SPEC CPU2006 benchmark*
Comprehensive Single Suite of Tools

Oracle Solaris Studio

Compiler Suite
- **C, C++ Compilers**
  Utilizes advanced code generation technology to optimize apps for highest performance on Oracle’s Sun servers (SPARC & x86).
- **Fortran Compiler**
  Optimizes compute-intensive application performance.
- **Debugger**
  Ensures app stability with event handling and multi-thread support.
- **Performance Library**
  Maximizes compute-intensive app performance across SPARC and x86 using advanced numeric solver libraries.

Analysis Suite
- **Performance Analyzer**
  Provides unparalleled insight into your app, allowing you to identify bottlenecks and improve performance by orders of magnitude.
- **Code Analyzer**
  Ensures app reliability by detecting app vulnerabilities, including memory leaks and memory access violations.
- **Thread Analyzer**
  Simplifies complex parallel programming errors by detecting hard to pinpoint race and deadlock conditions.

Integrated Development Environment
Performance Analyzer
Maximize Application Performance

- Profile fully optimized and parallelized production code
- Understand where time is being spent
- Trace to source and identify hotspots
- View timeline and call stack for select events
- Compare multiple experiments
- Supports C/C++, Java, OpenMP and distributed MPI-based apps
Code Analyzer (New Tool)

Ensure Application Reliability

- Easily detect and fix common coding errors, including memory leaks and access violations
- Advanced error filtering and sorting capabilities
- Synthesizes three types of errors
  - Source code verification
  - Memory access audits
  - Unexercised code
- Identifies 45+ error types
Error Types Found by Code Analyzer

**Static Analysis**
- ABR - beyond Array Bounds Read
- ABW - beyond Array Bounds Write
- DFM - Double Freeing Memory
- ECV - Explicit type Cast Violation
- FIN - PM_IN nonconformance
- FMR - Freed Memory Read
- FMW - Freed Memory Write
- FOU - PM_OUT use before definition
- ICV - Implicit type Cast Violation
- INF - INFinite empty loop
- IPV - Interior Pointer Violation
- Memory leak
- MFR - Missing Function Return
- MRC - Missing malloc Return value Check
- NFR - uNinitialized Function Return
- NUL - NULL pointer dereference, leaky pointer check
- RFM - Return Freed Memory
- UMR - Uninitialized Memory Read
- URV - Unused Return Value
- VES - out-of-scope local Variable usage

**Code Coverage**
- Uncovered function with potential coverage

**Dynamic Analysis**
- ABR - beyond Array Bounds Read
- ABW - beyond Array Bounds Write
- BFM - Bad Free Memory
- BRP - Bad Realloc address Parameter
- CGB - Corrupted Guard Block
- DFM - Double Freeing Memory
- FMR - Freed Memory Read
- FMW - Freed Memory Write
- IMR - Invalid Memory Read
- IMW - Invalid Memory Write
- OLP - Overlapping source and destination
- PIR - Partially Initialized Read
- SBR - beyond Stack Bounds Read
- SBW - beyond Stack Bounds Write
- UAR - UnAllocated memory Read
- UAW - UnAllocated memory Write
- UMR - Uninitialized Memory Read
- UAZS - Allocating Zero Size
- NAR - Non-Annotated Read
- NAW - Non-Annotated Write
- SMR - Speculative Memory Read
- UFR - Unknown stack Frame Read
- UFWR - Unknown stack Frame Write
- USR - Unknown Status while Reading
- USWR - Unknown Status while Writing
Thread Analyzer
Simplify Multicore Programming

- Easily identify complex parallel programming errors
- Detect hard to pinpoint race and deadlock conditions
- Supports Solaris threads, PThreads, OpenMP 3.1
Oracle Solaris Studio IDE
Enhance Productivity

• Tailored for C, C++ & Fortran developers
• Built on award-winning NetBeans platform
• Feature-rich language aware code editor
• Premier support for C++ templates
• Leverages powerful Solaris Studio memory, thread and performance analysis features
Remote Development
Simplify server-application development

- Remotely build, debug, run, analyze server-applications from any Oracle Solaris, Linux, Microsoft Windows or Mac OS desktop
- Leverage award-winning IDE features
- Remote commands are invoked over SSH
- Remote files are accessed through SFTP or any file sharing mechanism available in OS
- Built-in terminal sharing connection to remote host
Database Development
Streamline C, C++ DB Development

- Support for Oracle DB & MySQL in IDE
- Support for Pro*C and OCI-based applications
- Editor support includes syntax highlighting, code completion, & hyperlinking stmts
- Simplified setup with DB Project and Connection Wizards
- Manage DB schema, table and data within IDE
- Built-in SQL Editor
Availability and Support

- **Availability**
  - OTN download or Media Kit on Oracle Store

- **License:** Perpetual, no-fee

- **Support:** Oracle Solaris Development Tools Support
  - Oracle Premier Support Offering
    - Unlimited 24x7 service requests
    - Access to My Oracle Support for patches, fixes
  - $1200 per year per Named User

- **Lifetime Support Policy:** Oracle and Sun System Software Brochure
  - Premier: 5 yrs, Extended: 3 yrs, Sustaining: Indefinite
Webinar Registration Page

Oracle Solaris 11 Developer Webinar Series

Join us on April 34th for a complimentary webinar. In just one hour you will learn how to streamline your development process to maximize the performance of your application with Oracle Solaris 11. Access the Slides here.

✔ Learn why there is no more patching!
✔ Understand Oracle Solaris 11 cloud features
✔ Review how to maximize application reliability in Oracle Solaris 11
✔ Live Chat with Oracle Solaris 11 Engineers

Who should attend?

Application developers and administrators wanting a deep-dive on key features of Oracle Solaris 11 which you can exploit to make your applications superior to your competitors and easier to use.

Please note: (after registering) You will receive details on how to attend the meeting in a separate confirmation e-mail shortly before the webcast.

Next Webinar June 5th

Agenda

- Click on event to register
- All webinars on Tuesday's 9-10am PT (Event will support VOIP)
- Click here to see a preview teaser

<table>
<thead>
<tr>
<th>Webinar Series Topic</th>
<th>Date</th>
<th>Speaker</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modern Software Packaging for Enterprise Developers</td>
<td>03-27-12 @ 9am PT</td>
<td>Eric Reid (Principal Software Engineer)</td>
</tr>
<tr>
<td>Slides here - Replay here</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simplify Your Development Environment with Zones, ZFS &amp; More</td>
<td>04-10-12 @ 9am PT</td>
<td>Eric Reid (Principal Software Engineer) and Stefan Schneider (Chief Technologist ISV-Engineering)</td>
</tr>
<tr>
<td>Slides here - Replay here</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Managing Application Services – Using SMF Manifests in Oracle Solaris 11</td>
<td>04-24-12 @ 9am PT</td>
<td>Matthew Hosanee (Principal Software Engineer)</td>
</tr>
<tr>
<td>Slides here</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimize Your Applications on Oracle Solaris 11: The DTrace Advantage</td>
<td>05-08-12 @ 9am PT</td>
<td>Angelo Rajadurai (Principal Software Engineer)</td>
</tr>
<tr>
<td>Maximize Application Performance and Reliability on Oracle Solaris 11</td>
<td>05-22-12 @ 9am PT</td>
<td>Vijay Talkar (Senior Manager Software Development)</td>
</tr>
<tr>
<td>Writing Oracle Solaris 11 Device Drivers</td>
<td>06-05-12 @ 9am PT</td>
<td>Bill Knoche (Principal Software Engineer)</td>
</tr>
</tbody>
</table>


Registration: http://www.oracle.com/technetwork/server-storage/solaris11/overview/webinar-series-1563626.html

Oracle Corporation
Questions
Performance Analyzer: Functions View

Notice the metrics for overall resource consumptions here.

Clearly, one function dominates the time here.
Performance Analyzer: Timeline View

Notice the imbalance of loads between Threads (1,4) and (2,3)
Code Analyzer: Static Error View

Let's look at one error: Uninitialized Memory Read.
In this case, either variable may be uninitialized ...
Eg if path goes through 78 and then 80, then i is uninitialized
This is a tricky case, as well for uninitialized memory reads.
Performance Analyzer: Timeline View

The tool can detect various different types of errors.

Look at call stack
Look at allocation stack

Clearly, Memory was allocated in sample1(), but never initialized before calling this routine.
Code Analyzer: Coverage

Potential coverage opportunities, sorted by maximum to minimum coverage effectiveness.
Hardware and Software
Engineered to Work Together