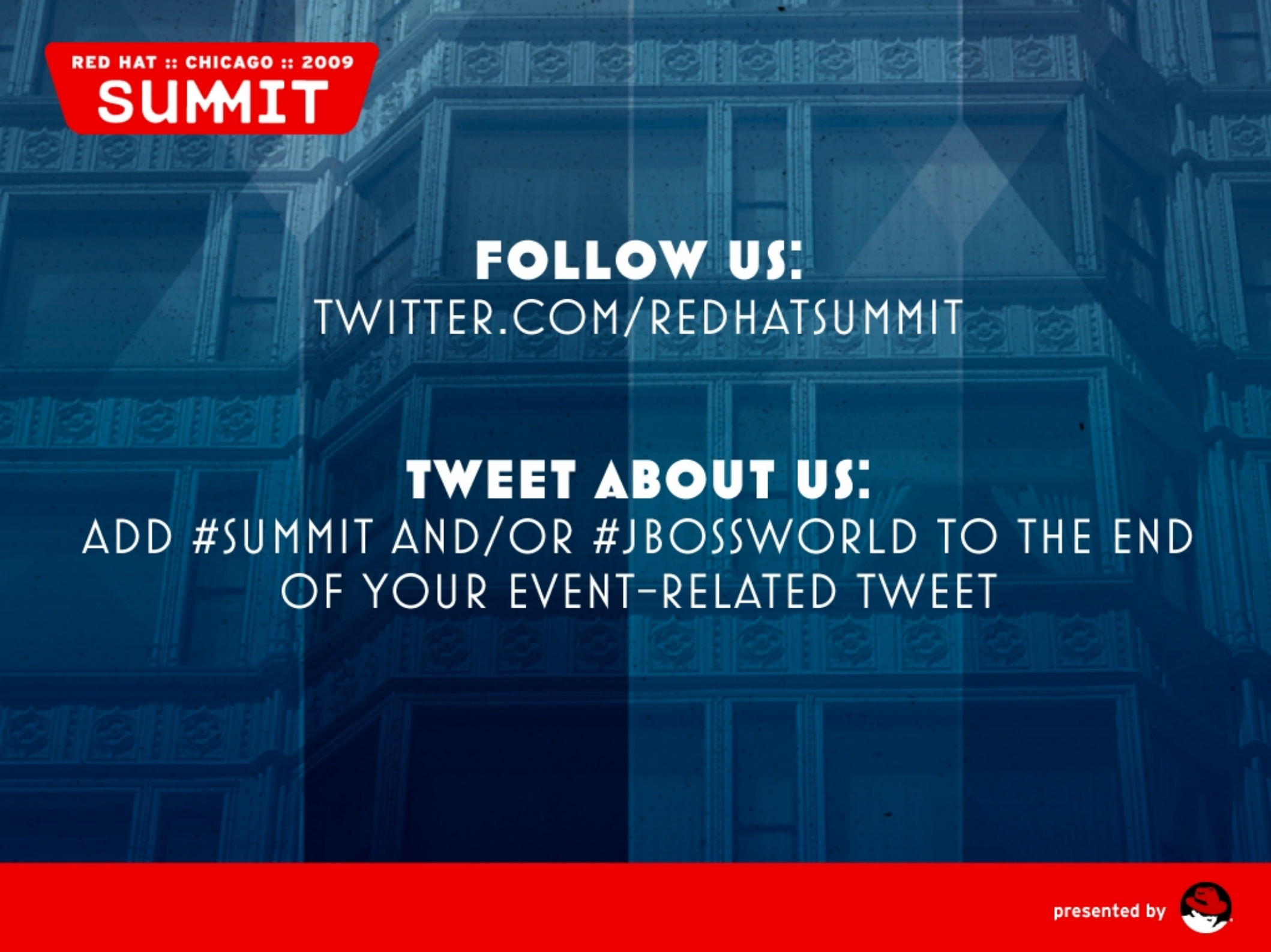
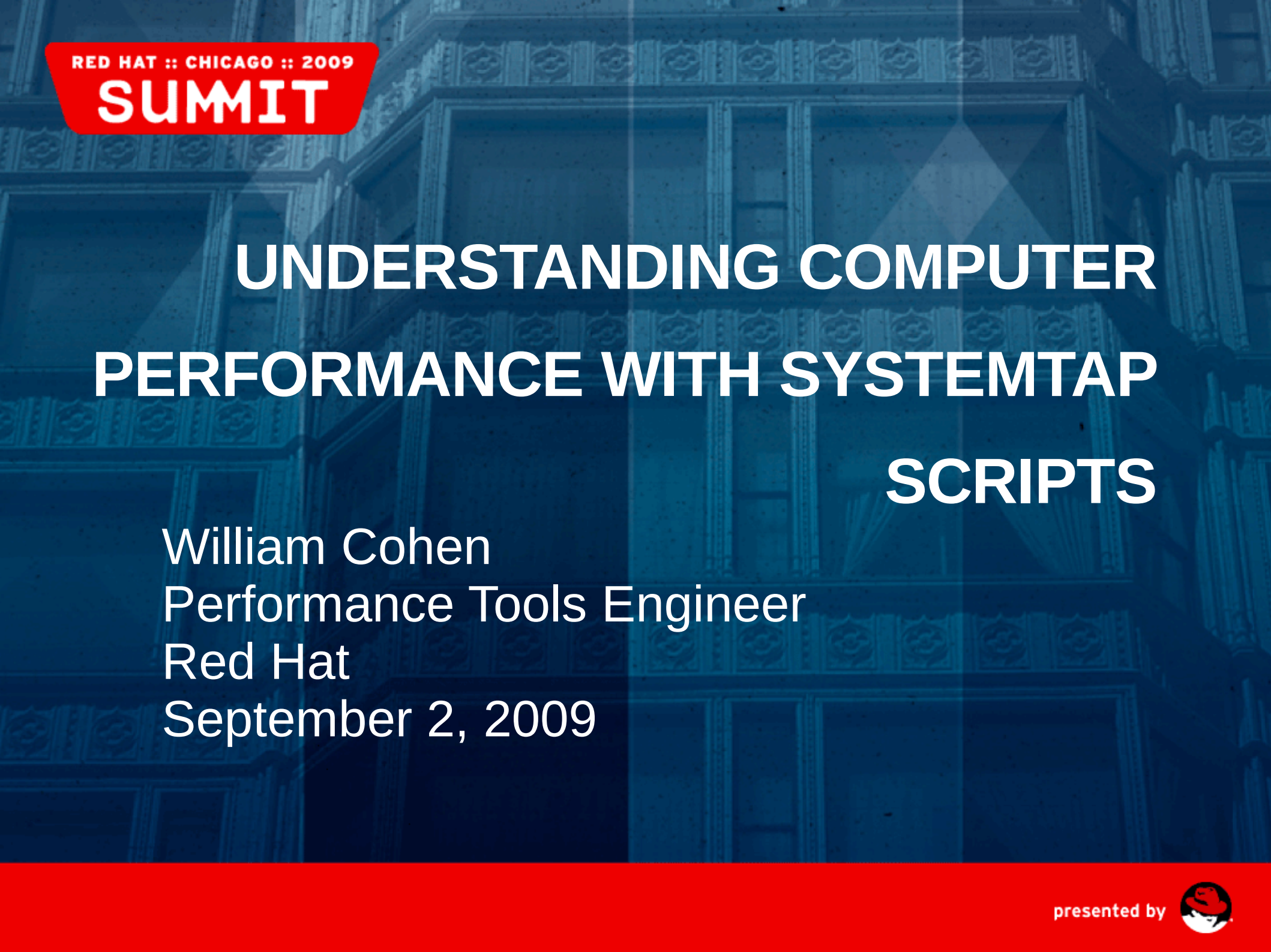

RED HAT :: CHICAGO :: 2009
SUMMIT
FOLLOW US:
TWITTER.COM/REDHATSUMMIT
TWEET ABOUT US:
ADD #SUMMIT AND/OR #JBOSSWORLD TO THE END
OF YOUR EVENT-RELATED TWEET
presented by



UNDERSTANDING COMPUTER PERFORMANCE WITH SYSTEMTAP SCRIPTS

William Cohen
Performance Tools Engineer
Red Hat
September 2, 2009

presented by

Agenda

What is SystemTap?

How does SystemTap work?

Setting up SystemTap

Very simple example

“Ready-to-run” scripts

Common techniques in SystemTap

Writing your own SystemTap scripts

Where to get more information

What is SystemTap?

- Dynamic scriptable tracing tool
- Probes running system:
  - No recompile of code
  - No re-install of executable
  - No re-starting of system
- Powerful scripting features:
  - Conditional constructs
  - Associative arrays
  - Statistics and histograms

How does SystemTap work?

SystemTap scripts composed of:

- Probed events
- Probe handlers

SystemTap produces kernel module from script:

- Maps probed events to kernel mechanism such as utrace, kprobes, and timers
- Translates probe handlers

RPMs needed for SystemTap

SystemTap RPMs:

- systemtap
- systemtap-runtime

Supporting RPMs:

- kernel-debuginfo
- kernel-debuginfo-common
- Kernel-devel

Getting SystemTap Setup

- Installing systemtap RPM
  - `yum install systemtap`
- Finding needed supporting RPMs:
  - RHEL5.4 systemtap-0.9.7 has `stap-prep` (`stapprep.sh` in upstream systemtap sources)
  - The `stap-prep` script lists needed RPM
- Proper permission to run systemtap:
  - Member group stapusr (run precompiled scripts)
  - Member group stapdev (build systemtap scripts)
  - root

Very Simple Example Script

File `hello.stp`:

```
probe begin
{
  printf ("hello world\n")
  exit ()
}
```

Run with:

```
stap hello.stp
```

Output:

```
hello world
```

SystemTap Script Termination

Control-c

`exit()` function

Exit of command started with SystemTap `-c` option

“Ready-to-Run” Scripts

Building catalog of SystemTap scripts

Catalog included in systemtap RPM,
e.g. `/usr/share/doc/systemtap-0.9.7/examples`

Have indices with short descriptions:

/usr/share/doc/systemtap-0.9.7/examples/index.html

/usr/share/doc/systemtap-0.9.7/examples/index.txt

Executable IO Statistics

examples/io/iostats.stp

Monitors vfs reads and writes

Generates statistics for each executable:

- Number reads and write operations
- Total amount of data read and written
- Average size of reads and writes

Process Polling Example

examples/profiling/timeout.stp

Monitors syscalls that timeout

Prints a top-like output of processes that timeout

Useful to find processes that keep waking

Page Faults Example

examples/memory/pfaults.stp

Log each page fault with:

- Timestamp
- PID
- Virtual address
- Read/write
- Major/minor
- Elapsed time for handling page fault

Common SystemTap Script Uses

"Super strace"

Determine whether particular function is called

Get traceback to determine what is calling a function

Examine arguments passed into or returned by a function

Determine which process or thread is triggering an event

Determine time between events

“Super Strace”

Strace is a very useful tool

strace limitations:

- Only able to watch a single process
- Limits on filtering (cannot filter on return values)
- Can generate very verbose log

Systemtap able to monitor syscalls system-wide

Systemtap can have more flexible filtering, for example syscall return value < 0

Writing Your Own Systemtap Scripts

- Use existing examples as starting points
- Find possible probe points with "`-L`" option:

```
stap -L 'kernel.trace("*")'
```

- Systemtap man pages
- Look through tapsets (/usr/share/systemtap/tapset) for probe points
- Look through the kernel sources

Navigating the Linux Kernel

Linux kernel cross references (lxr):

- RHEL kernels http://rhkernel.org/
- Upstream kernels http://lxr.linux.no/linux/

Where to get more information

Red Hat Enterprise Linux SystemTap Beginner's Guide:

http://www.redhat.com/docs/manuals/enterprise/

SystemTap project page:

http://sourceware.org/systemtap/

Forums for questions and help:

Email systemtap@sources.redhat.com

IRC #systemtap on irc.freenode.net

Get Your Script into the SystemTap Examples

Submit scripts for the examples

Improve quality of script with feedback on the script from SystemTap developer

Make sure that script works on wide variety of environments, example scripts are run a part of testsuite

More details about submitting examples in:

/usr/share/doc/systemtap-0.9.7/examples/README

QUESTIONS?

TELL US WHAT YOU THINK:
REDHAT.COM/SUMMIT-SURVEY