HITACHI
Inspire the Next

SystemTap for Enterprise

Enterprise Features in SystemTap

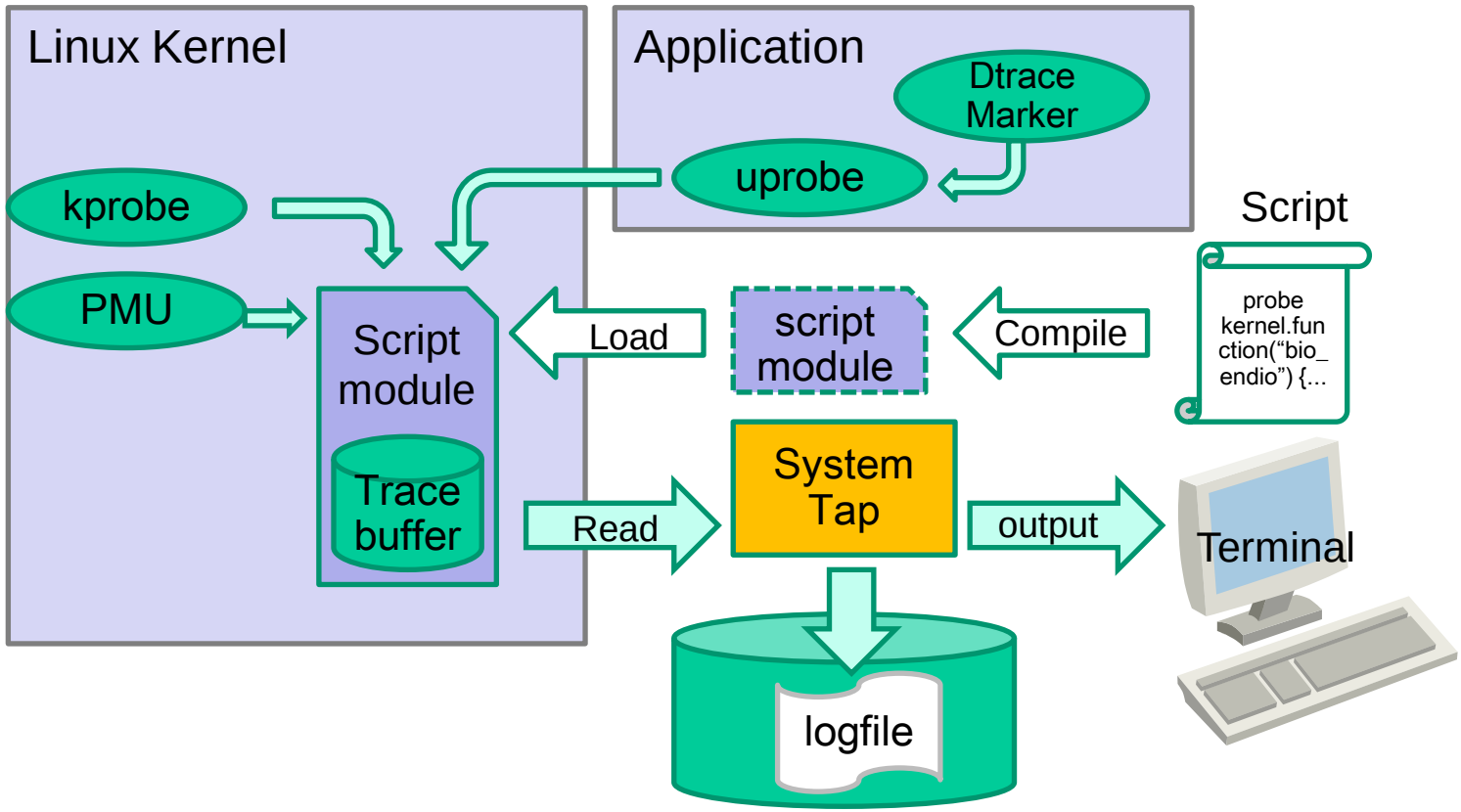
2010/09/28

Hitachi Systems Development Laboratory
Linux Technology Center

Masami Hiramatsu

HITACHI
Inspire the Next

- Tracing Script Framework
  - Run scripts as kernel modules
  - Support User-space and Kernel tracing



Hitachi's Background in Enterprise IT Area

HITACHI
Inspire the Next

- Hitachi Ltd., is one of the biggest IT system vendors in Japan
  - Big knowledge and long experience about RAS feature for **Enterprise IT systems**
    - Including Tracer, Dump, etc.
  - We know what the Enterprise system needs
- We are also working on Linux Tracing 9 years
  - LKST, SystemTap, kprobes
  - ftrace and perf
- Enterprise RAS feature on Systemtap
  - What we have developed

Enterprise-level Tracing

HITACHI
Inspire the Next

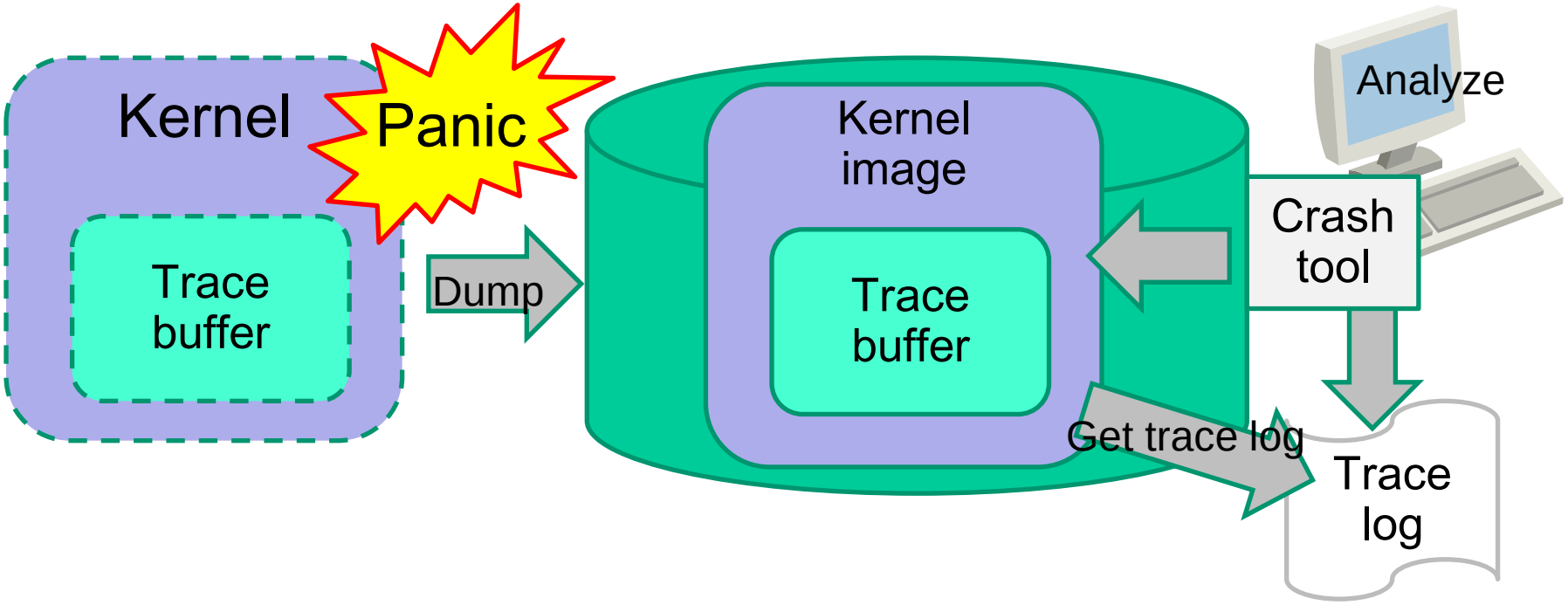
- Typical Troubles on Enterprise IT System
  - A production system has crashed!
    - How can we find a root cause of crash?
  - System slowed down!
    - Which part of the system is the bottle neck?
- We may have a test system, but...
  - Production system can not be used for analysis
    - Because we need to continue use it.
  - It might be hard or take a time to reproduce the Bug on test system.
  - Sometimes it's so hard to build a test system
    - Enterprise system is usually very very Expensive.

->Tracing: avoid/reduce trouble reproducing

Use Case 1: Kernel Panic!

HITACHI
Inspire the Next

- Trouble shooting with trace log
  - **Set up tracers** on the running system
  - **Trace system always on memory**
  - When the system has crashed, dump memory image
  - **Get trace log** from the image



Setup System Tracing

HITACHI
Inspire the Next

- Run tracer as a system service
  - Flight recorder always need to start with the system
- SystemTap: initscript support
  - This initscript allows us to start/stop tracing automatically
    - Start all registered scripts

```
# service systemtap start
```

    - Stop all running scripts

```
# service systemtap stop
```

SystemTap Initscript Support

HITACHI
Inspire the Next

- Registering new script
  - Copy the script and add config options.

```
# cp iomonitor.stp /etc/systemtap/script.d/
# cat > /etc/systemtap/conf.d/iomonitor.conf
iomonitor_OPT="-o /var/log/iomonitor.log"
^D
```

  - Compile the script
    - will be automatically done at the 1st start

```
# service systemtap compile
```

- Start/stop/state individual script

```
# service systemtap start iomonitor
```

- Deploying pre-compiled scripts is also supported

```
# cp iomonitor.ko /var/cache/systemtap/2.6.18-5.el/
```

On-memory Flight Recording

HITACHI
Inspire the Next

- Recording events on kernel memory
  - Always tracing on the system
  - Record the events of last several minutes
- SystemTap: '-F' option
  - Start tracing and detach from the script
    - script continues to run in the kernel

```
# stap -F flightrecord.stp -m frtrace
```

  - Attach to read log
    - Note: read data still remain in the kernel

```
# stap -A frtrace
```

    - To detach again, just push Ctrl+backslash (or send SIGQUIT)

Crash Log Analysis

HITACHI
Inspire the Next

- Even if we have a kernel trouble Kdump can dump kernel image.
  - But how can we get actual tracing log data?
  - “Crash” can retrieve it
- SystemTap: staplog
  - Load a “staplog” extension on crash

```
crash> extend staplog.so
```

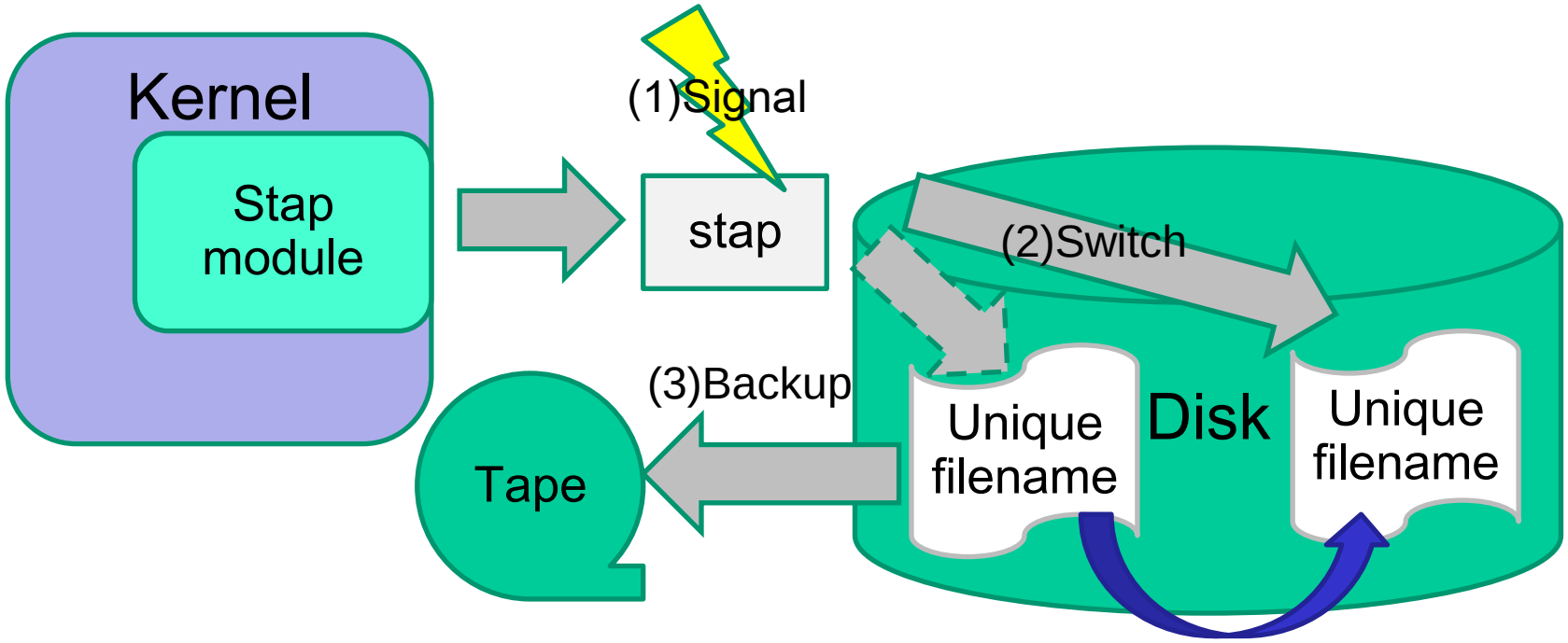
  - And save all logs into logfiles

```
crash> systemtaplog -a frtrace
```

Use Case 2: System Monitoring on Disk

HITACHI
Inspire the Next

- Monitoring system, long time
  - Keep the log on disk for auditing system behavior
  - Consider logdata backup
    - Switch logfile anytime before backup
    - Unique logfile name to prevent overwrite



On-File Flight Recorder

HITACHI
Inspire the Next

- Recording events on file
  - Tracing events longer time, permanently
    - Also, we need to take care of a disk-size limitation
- SystemTap: '-o' and '-S' option
  - Flight recorder('-F') with output on the file('-o')
    - stap(stapio) runs in background (as a daemon)

```
# stap -F flightrecord.stp -o logdata -m frtrace
```

  - With file-size limitation (limit to 10MB/file)
    - When the logdata.0 hits 10MB, stap switches to logdata.1

```
# stap -F flightrecord.stp -o logdata -S 10 -m frtrace
```

  - With file-number limitation (leave last 3 log files < 30MB)
    - When stap switches to logdata.4, it removes logdata.0

```
# stap -F flightrecord.stp -o logdata -S 10,3 -m frtrace
```

Signal File Switching

HITACHI
Inspire the Next

- Switch the logfile anytime
  - When the system admin want to backup it
- SystemTap: SIGUSR2
  - SystemTap switches the log file when SIGUSR2 received
  - Useful with on-file flight recorder

```
# ps -o pid= stapio
1234
# kill -USR2 1234
```

Unique Log-file Naming

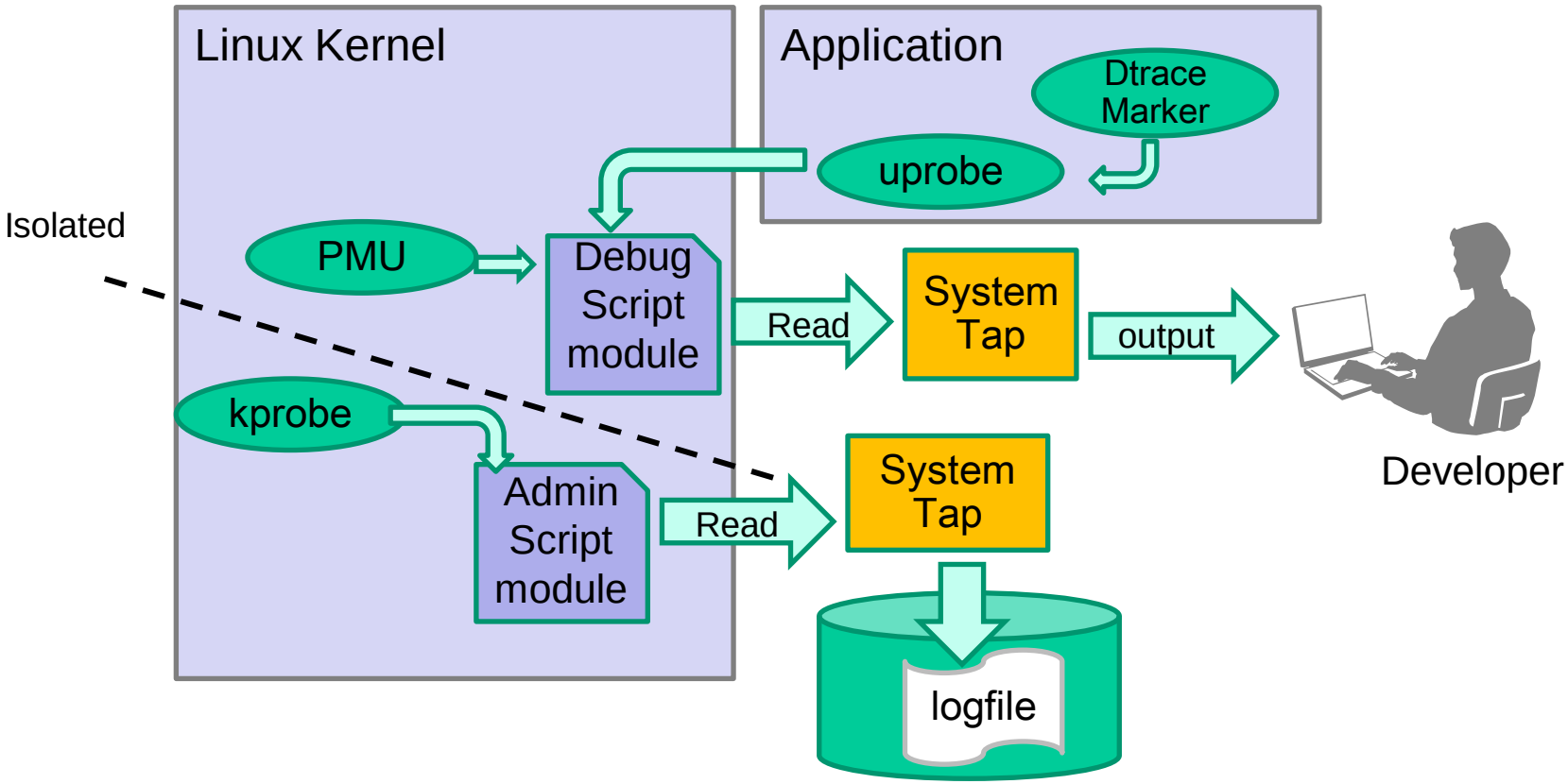
HITACHI
Inspire the Next

- Unique filename helps backup
  - Prevent unwilling overwrite (miss operation)
- SystemTap: Log file format with date/time
  - “-o” option accepts strftime(3) format
    - e.g.
      “%m-%d-%y” converted to “9-28-2010”

Use Case 3: Multi User, Multi Purpose

HITACHI
Inspire the Next

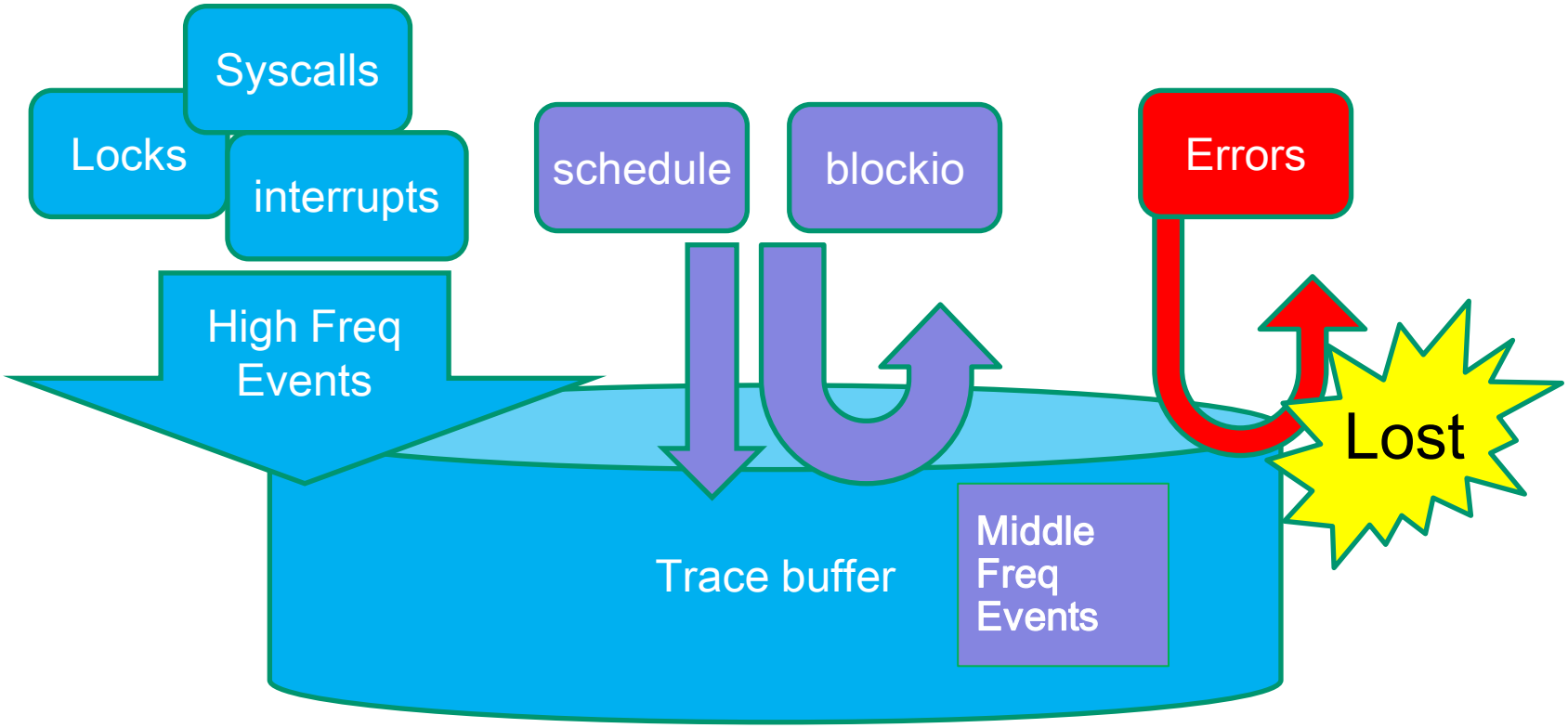
- Tracers for multi purpose concurrently
  - Admin has to run flight recorder
  - Developer may run tracer for profiling/debugging



Use Case 4: Event Flooding

HITACHI
Inspire the Next

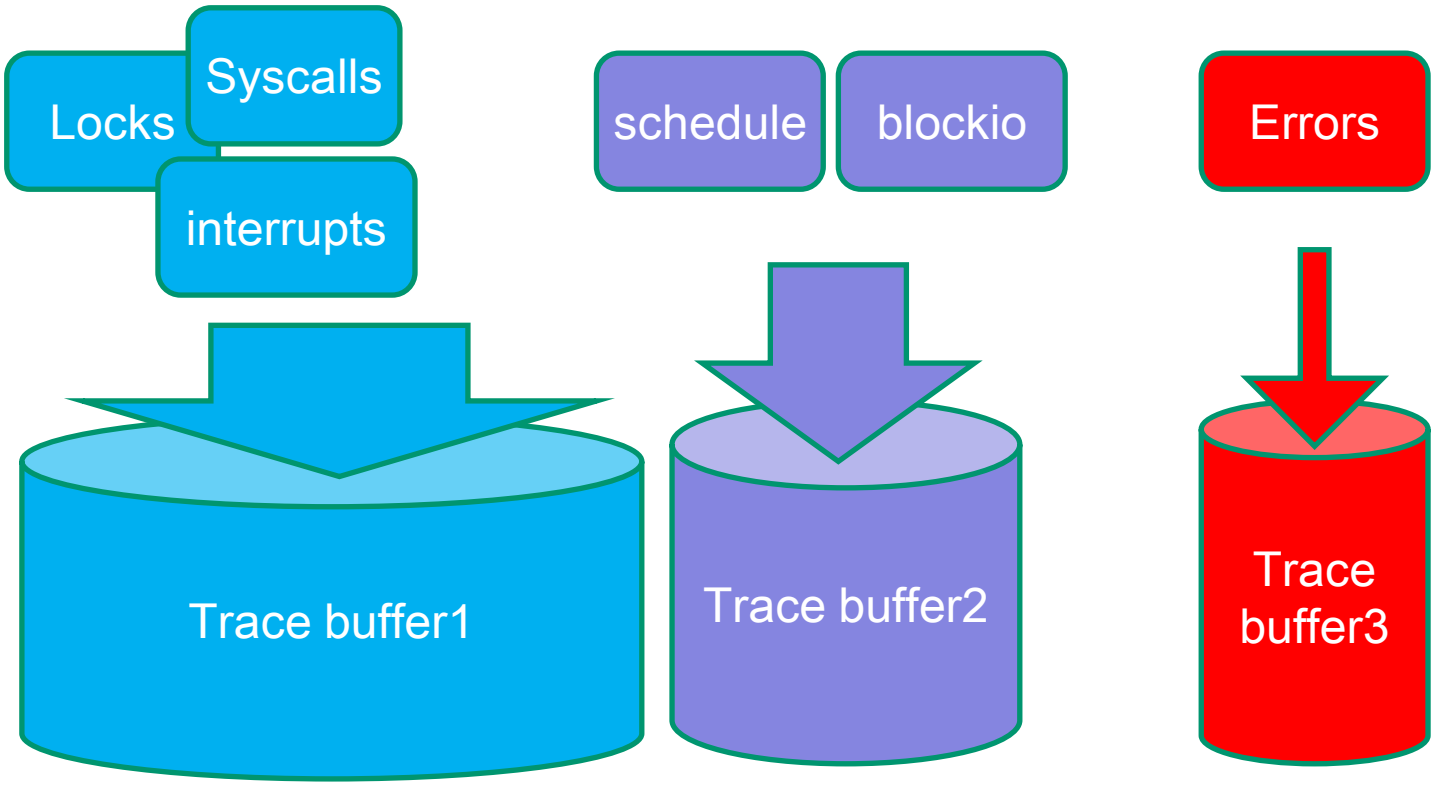
- There is a variety of events
  - High-freq events can push out all other events
    - Very rare events (e.g. Error) can be most important.



Buffers for Each Event Group

HITACHI
Inspire the Next

- Separate trace buffers for each groups of events
  - Protect important events from flooding



Multi-Session

HITACHI
Inspire the Next

- Concurrent/isolated tracing sessions
  - Multi-user environment
    - Admin and developers
  - Multi-purpose tracing
    - Profiling and Flight recording
  - Multi-event tracing
    - High-Freq. events are usually LESS important
    - Low-Freq./Rare events are MOST important
- SystemTap: Each scripts have different buffers ☺

Misc 1: Online Trace Configuration

HITACHI
Inspire the Next

- On-line add/remove tracepoints
  - It's hard to stop flight recorder for adding trace events
    - Unexpected system crash while configuring
  - Trace configuration changes must be on-line
- SystemTap: Shared Buffer
  - Sharing a ring buffer among multiple scripts
  - User can add/remove trace-scripts on-line
    - flightrecord.stp writes log into buffer.stp's buffer

```
# stap -F buffer.stp -s 128M -DRELAYHOST=shbuf
# stap -F flightrecord.stp -m frtrace -DRELAYGUEST=shbuf
```

Misc 2: User-space Tracing

HITACHI
Inspire the Next

- Most part of the system code is in the user space
  - There are 2 major frameworks
  - LTTng: UST
    - Gdb support
  - Sun's DTrace
    - Widely supported on Mac OSX, BSD, Solaris by Java, Javascript, PostgreSQL, etc.
- SystemTap: DTrace compatibility!
  - Apps can use DTrace compatible marker
  - User can use DTrace scripts and dtrace command

Misc 3: Flight Recorder Over Network

HITACHI
Inspire the Next

- Trace from remote machine
  - Not yet supported officially☹
    - "*Actually, we are in the planning stages of two or three new orthogonal efforts in this area*. " Frank Ch. Eigler
  - Maybe, the "netcat" can handle it (not in background).
    - Tracing host(logger) side

```
host# nc -l 12345 > guest.log
```

    - Tracing guest(trace target) side

```
guest# stap -F flightrecord.stp | nc guest 12345
```

Conclusion: Enterprise Tracing Features

HITACHI
Inspire the Next

- SystemTap covers many tracing features for Enterprise-level requirements
  - Flight Recorder Mode
  - Initscript Support
  - Crash Log Analysis
  - Multi-session Tracing
  - User-application Tracing
- Good examples what enterprise people need

Trademarks

HITACHI
Inspire the Next

- Linux is a trademark of Linus Torvalds in the United States, other countries, or both.
- Other company, product, or service names may be trademarks or service marks of others.

HITACHI
Inspire the Next

END

SystemTap for Enterprise

Enterprise Features in SystemTap

2010/9/28

Hitachi Systems Development Laboratory
Linux Technology Center

Masami Hiramatsu


HITACHI
Inspire the Next