DTrace Introduction

Kyle Hailey and Adam Leventhal

DELPHIX™

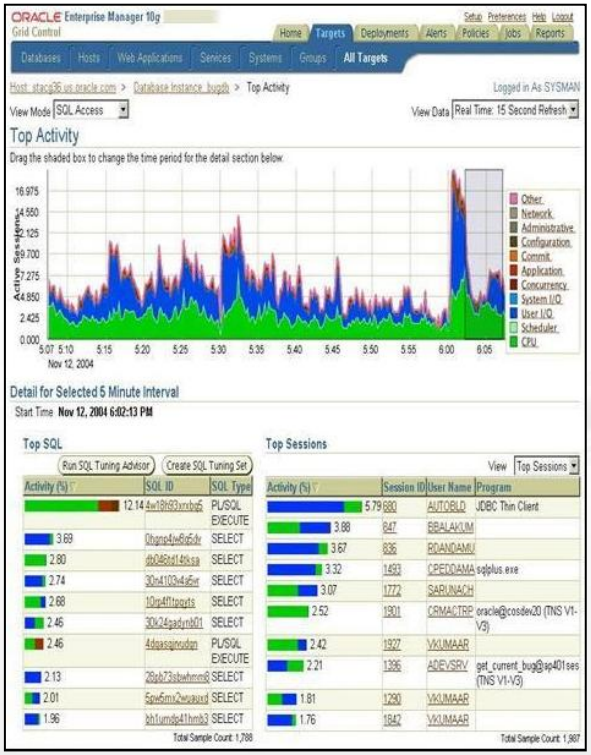
Agenda

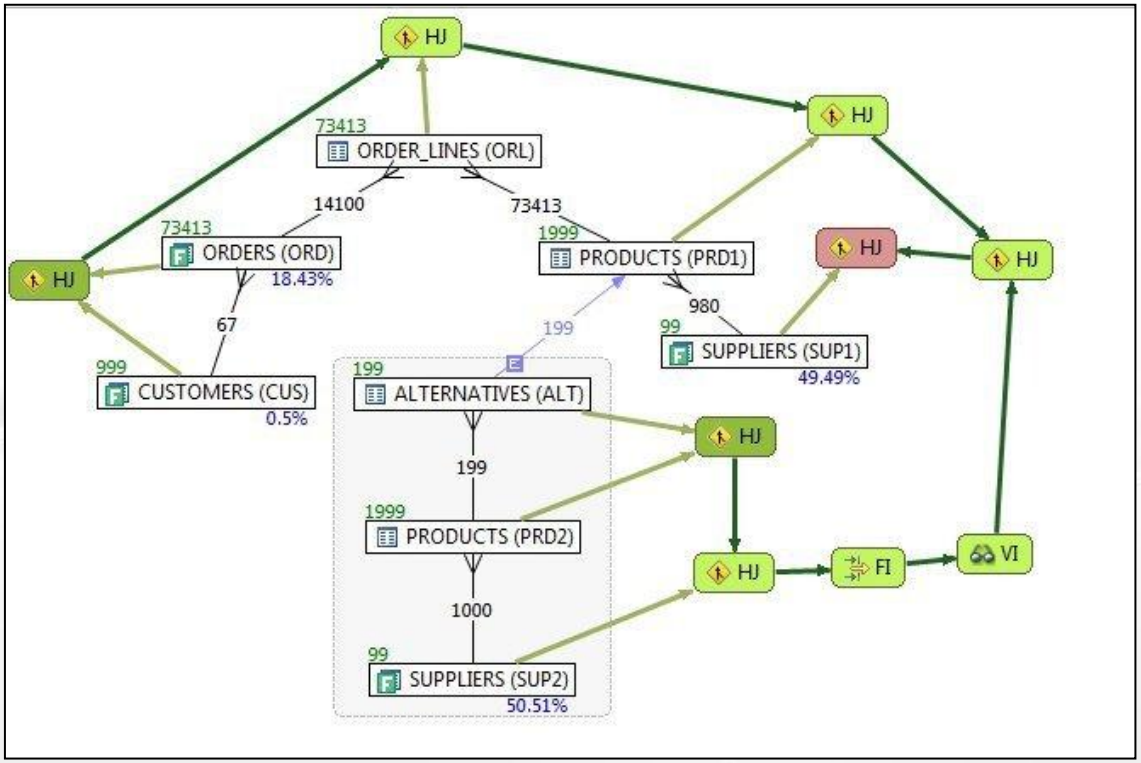
- Intro
- Performance problems
  - Cloned DB slower when everything the same
  - Orion benchmark impossibly fast
  - Oracle process on 100% CPU, no waits
- How DTrace can answer them
- Live Examples
- Getting Started Info
- Resources

DELPHIX™

Kyle Hailey

- OEM 10g Performance Monitoring
- Visual SQL Tuning (VST) in DB Optimizer





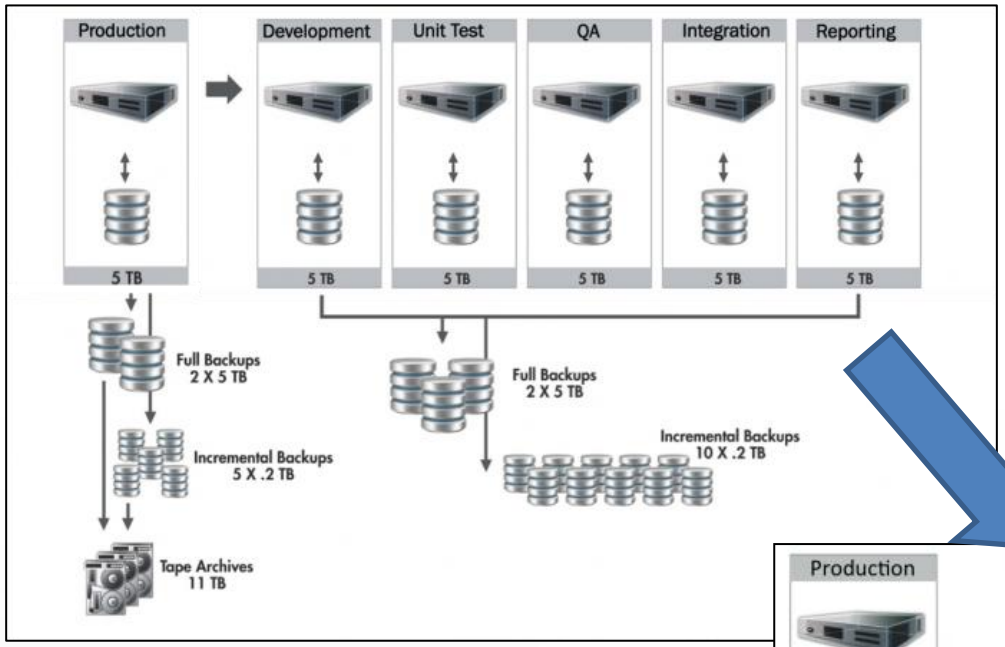
DELPHIX™

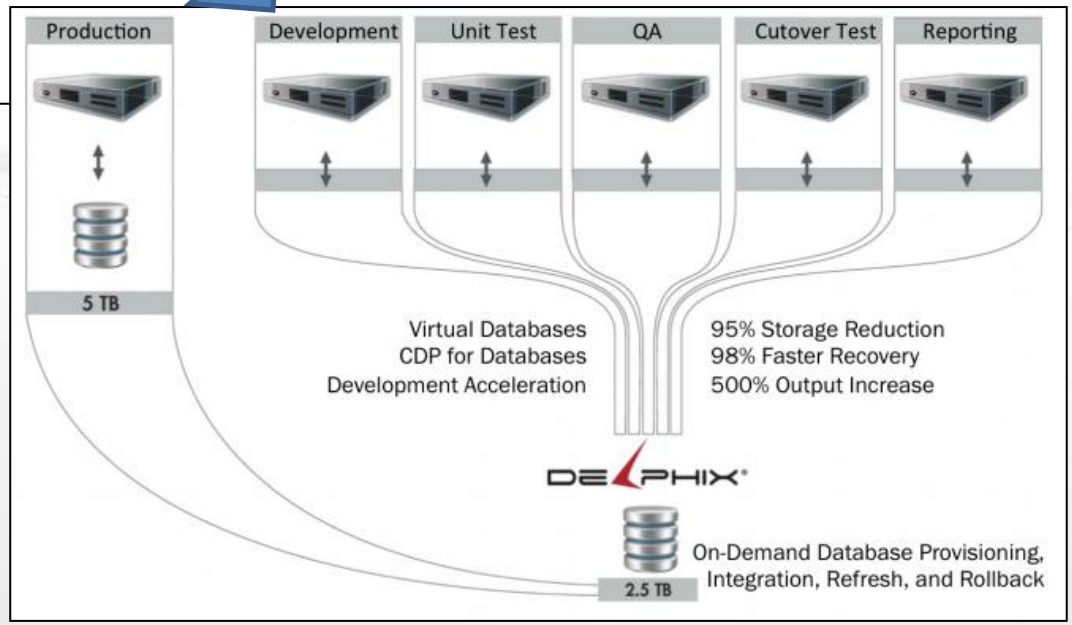
Adam Leventhal

- Co-Creator of Dtrace
- Founder of Fishworks at Sun
  - storage appliance built on ZFS, DTrace
  - invented the Hybrid Storage Pool

DELPHIX™

Delphix





DELPHIX™

Cloned database Slower

Original Database

call	count	cpu	elapsed	disk	query	current	rows
Parse	25535	3.71	4.80	54	1491	1972	0
Execute	66847	22.46	54.13	1320	23612	8098	1277
Fetch	236644	19.79	282.19	61943	729314	18	215752
total	329026	45.96	**341.13**	63317	754417	10088	217029

Event waited on	Times Waited	Max. Wait	Total Waited	
db file sequential read	62182	0.27	278.55	-> **avg = 4.5 ms**

Clone Database

call	count	cpu	elapsed	disk	query	current	rows
Parse	25412	2.85	3.38	13	1080	650	0
Execute	69435	24.99	63.18	1123	23205	7199	1128
Fetch	245632	14.54	452.71	53127	611208	20	223907
total	340479	42.38	**519.28**	54263	635493	7869	225035

Event waited on	Times Waited	Max. Wait	Total Waited	
db file sequential read	53635	0.45	455.12	-> **avg = 8.5 ms**

DELPHIX™

Cloned database slower

- Database same configuration, hardware, SAN
- Traces show:
  - 4.5 ms on original and 8.5 ms on clone
  - Why?
- Theory: more data cached on host
- Prove?
  - V$event_histogram
    - maximum granularity 1ms
    - have to snap shot and take deltas
    - System wide
  - Tracing 10046
    - session specific
    - custom scripts
    - still guessing
- Solution: DTrace to see how many I/Os are from cache and from disk

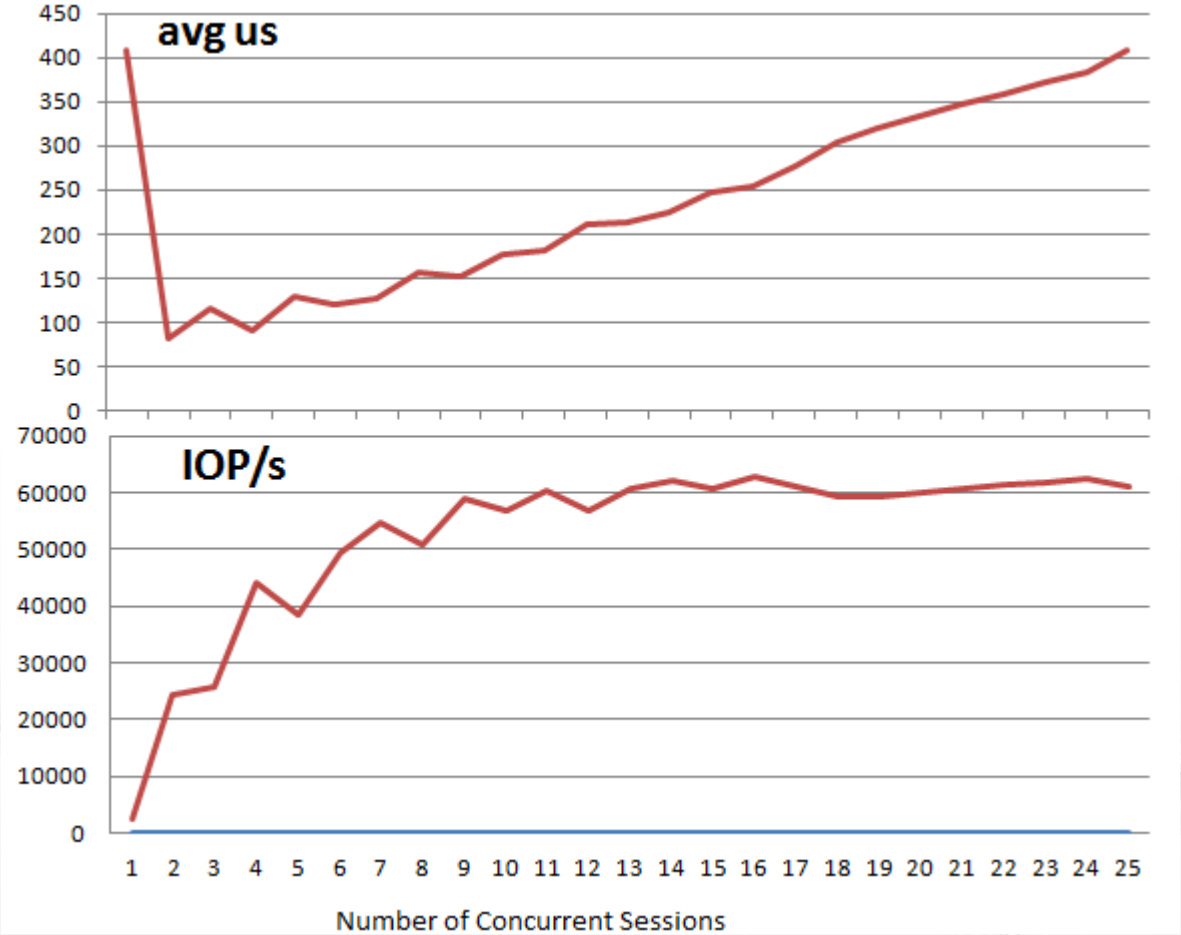
DELPHIX™

Orion Benchmark Anomalies

Setup:
First run of Orion
8K random reads
Host has 48GB
Test file size 96GB
5 Disks
EMC array 2GB cache

Result:
60K IOP/s -> 60 Disks
Latency 0.1-0.4ms !

Theory:
orion is not doing
random reads but
re-reading same blocks



How do we prove it?Dtrace to see if same block is re-read

DELPHIX™

Oracle Process 100% CPU bound

- Process has 100% CPU bound
- Process shows now waits
- Where is it spending it's time?

- DTrace with stack trace to see top function
- DTrace to see how much time is from scheduling and paging

What is DTrace

- Your code unchanged
  - Optional add DTrace probes
  - Optional add Dtrace providers
- No overhead when off
  - Turning on dynamically changes code path
- Low overhead when on

- Event Driven : Like event 10046, 10053
- Not like ASH, though could be using profiling

DELPHIX™

Structure

```
#!/usr/sbin/dtrace -s

something_to_trace
/ filters /
{ actions }

 Something_else_to_trace
/filters_optional /
{ take some actions }
```

DELPHIX™

Event Driven

- Program runs until canceled
- Dtrace Code run when probes fire in OS
- Sections of the same probe fire in sequence

What can we trace?

Almost anything

- All DTrace stable providers
- All System calls (unstable if no provider)
- All function calls in a program

Where can we trace

- Solaris
- OpenSolaris
- FreeBSD …
- MacOS
- Linux – announced from Oracle
- AIX – working “probevue”

DELPHIX™

List of probes that can be traced

- Providers and unstable probes:
  - dtrace –l
- Process functions
  - Dtrace –l pid[pid]

- Probes have 4 part name
  - Provider:module:function:name
- Example
  - Dtrace –l | grep tcp | grep receive
  - tcp:ip:tcp_input_data:receive

DELPHIX™

Providers from: dtrace –l

Example breakdown count of providers

```
Count   provider                    area

  72095 fbt                       - function boundary tracing
   1283 sdt                       - statically defined trace locations
    629 mib                       - system statitics
    473 hotspot_jni, hotspot      - JVM
    466 syscall                   - system calls
    173 nfsv4,nfsv3,tcp,udp,ip    - network
     61 sysinfo                   - kstat statistics
     55 sched                     - scheduler, CPU
     46 fsinfo                    -  file system info
     41 vminfo                    - virtual memory
     40 iscsi,fc                  - iscsi,fibre channel
     22 lockstat                  - locks
     15 proc                      - fork, exit … ?
     14 profile                   - timers tick
     12 io                        - io:::start, done
      3 dtrace                    - BEGIN, END, ERROR
```

DELPHIX™

Dtrace –ln

Limit output to specific probes:

```
sudo dtrace -ln tcp:::
   ID   PROVIDER    MODULE  FUNCTION                    NAME
 7301        tcp       ip   tcp_input_data              receive
 7302        tcp       ip   tcp_input_listener          receive
 7303        tcp       ip   tcp_xmit_listeners_reset    receive
 7304        tcp       ip   tcp_fuse_output            receive
```

DELPHIX™

dtrace –lnv

Find out arguments for specific probe

dtrace -lvn tcp:ip:tcp_input_data:receive

ID	PROVIDER	MODULE	FUNCTION	NAME
7301	tcp	ip	tcp_input_data	receive

Argument Types

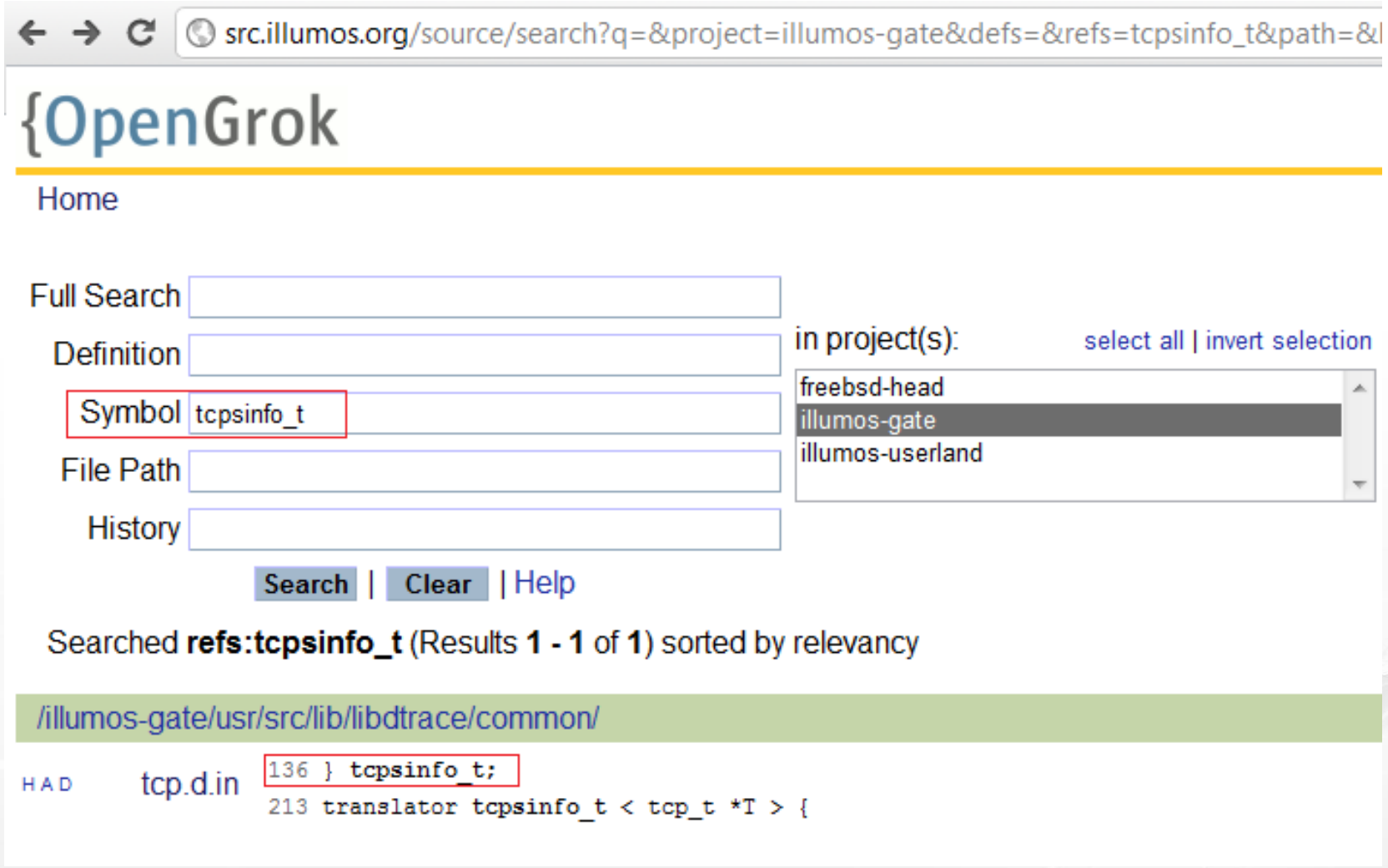
args[0]: pktinfo_t *
args[1]: csinfo_t *
args[2]: ipinfo_t *
args[3]: tcpsinfo_t *
args[4]: tcpinfo_t *

What is a “tcpsinfo_t ”?

- Wiki: https://wikis.oracle.com/display/DTrace/tcp+Provider
- Got to scr.illumos.org

DELPHIX™

Find out args for fbt probes: src.illumos.org



DELPHIX™

{OpenGrok

xref: /illumos-gate/usr/src/lib/libdtrace/common/tcp.d.in

Home | History | Annotate | Line # | Navigate | Download | Search only in c

```
107
108 /*
109  * tcpsinfo contains stable TCP details from tcp_t.
110  */
111 typedef struct tcpsinfo {
112         uintptr_t tcps_addr;
113         int tcps_local;                 /* is delivered locally, boolean */
114         int tcps_active;                /* active open (from here), boolean */
115         uint16_t tcps_lport;            /* local port */
116         uint16_t tcps_rport;            /* remote port */
117         string tcps_laddr;              /* local address, as a string */
118         string tcps_raddr;              /* remote address, as a string */
119         int32_t tcps_state;             /* TCP state */
120         uint32_t tcps_iss;              /* Initial sequence # sent */
121         uint32_t tcps_suna;             /* sequence # sent but unacked */
122         uint32_t tcps_snxt;             /* next sequence # to send */
123         uint32_t tcps_rack;             /* sequence # we have acked */
124         uint32_t tcps_rnxt;             /* next sequence # expected */
125         uint32_t tcps_swnd;             /* send window size */
126         int32_t tcps_snd_ws;            /* send window scaling */
127         uint32_t tcps_rwnd;             /* receive window size */
128         int32_t tcps_rcv_ws;            /* receive window scaling */
129         uint32_t tcps_cwnd;             /* congestion window */
130         uint32_t tcps_cwnd_ssthresh;    /* threshold for congestion avoidance */
131         uint32_t tcps_sack_fack;        /* SACK sequence # we have acked */
132         uint32_t tcps_sack_snxt;        /* next SACK seq # for retransmission */
133         uint32_t tcps_rto;              /* round-trip timeout, msec */
134         uint32_t tcps_mss;              /* max segment size */
135         int tcps_retransmit;            /* retransmit send event, boolean */
136 } tcpsinfo_t;
```

DELPHIX™

Variables

- types
  - Aggregates
  - This->
  - Self->
  - Globals
- Variable overhead

type	prefix	scope	overhead	multi-CPU safe	example assignment
aggregation	@	global	low	yes	@x = count();
clause local	this->	clause instance[1]	very low	yes	this->x = 1;
thread local	self->	thread	medium	yes	self->x = 1;
scalar	*none*	global	low-medium	no[2]	x = 1;
associative array	*none*	global	medium-high	no[2]	x[y] = 1;

dtrace.org/blogs/brendan/2011/11/25/dtrace-variable-types/

DELPHIX™

Built in variables

- pid – process id
- tid – thread id
- execname
- timestamp – nano-seconds (walltimestamp)
- cwd – current working directory
- Probes:
  - probeprov
  - probemod
  - probefunc
  - probename

Formatting data

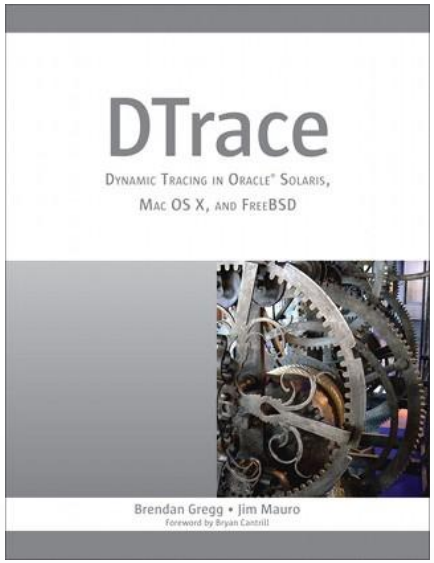
Format in data from DTrace in Perl

In Dtrace:

- No floating point
- No way to access index of an aggregate array
- Can’t divide elements of one array by another
  (ex sum of time by sum of counts)

Resources

- Oracle Wiki
  - wikis.oracle.com/display/Dtrace
- DTrace book:
  - www.dtracebook.com
- Brendan Gregg's Blog
  - dtrace.org/blogs/brendan/
- Oracle examples
  - alexanderanokhin.wordpress.com/2011/11/13
  - andreynikolaev.wordpress.com/2010/10/28/
  - blog.tanelpoder.com/2009/04/24



DELPHIX™

DTrace Book

- Tips and Tricks CH14 p987
  - Time Stamp Column, Postsort
  - Use Perl to Postprocess
    - Sudo mydtrace.d | perl -e '…'
  - Variable Scope and Use
- DTrace Cheat Sheet p 1069