



# Eclipse Plug-ins Development Linux Tools

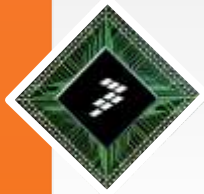
Catalin Udma





## AGENDA

- Linux Tools – command line tools
  - Perf
  - Valgrind
  - Gdb
- Eclipse Linux Tools Project
  - Usability improvement
  - Use cases and examples



# Linux Tools - perf

- Perf is a performance analysis tool that is based on the perf\_events interface made available in Linux Kernels Version 2.6 and higher
- Perf exposes its functionality via a simple but powerful command line interface.
- “perf” command:

```
usage: perf [--version] [--help] COMMAND [ARGS]
```

Examples:

```
perf stat: obtain event counts
```

```
perf record: record events for later reporting
```

```
perf report: break down events by process, function, etc.
```

```
perf annotate: annotate assembly or source code with event counts
```

```
perf top: see live event count
```



# Perf: performance evaluation

```
#define L 20000
#define C 20000
int a[L][C];

unsigned long long f1() {
    unsigned long long ret = 0;
    int x, y;
    for (x = 0; x < L; x++)
        for (y = 0; y < C; y++)
            ret += a[x][y];
    return ret;
}
```

```
#define L 20000
#define C 20000
int a[L][C];

unsigned long long f2() {
    unsigned long long ret = 0;
    int x, y;
    for (y = 0; y < C; y++)
        for (x = 0; x < L; x++)
            ret += a[x][y];
    return ret;
}
```

The same algorithm, the same result.

The same performance ?



# Perf: performance evaluation

```
$ perf stat -e cache-misses ./f1
```

```
Performance counter stats for './f1':
```

```
    4007 cache-misses
```

```
 1.855771646 seconds time elapsed
```

```
$ perf stat -e cache-misses ./f2
```

```
Performance counter stats for './f2':
```

```
 11401273 cache-misses
```

```
 5.551568045 seconds time elapsed
```



# Perf: performance evaluation

```
$ perf record -e cpu-clock test_perf
```

```
!!!Hello World!!!
```

```
[ perf record: Woken up 1 times to write data ]
```

```
[ perf record: Captured and wrote 0.009 MB perf.data
```

```
$ perf report
```

#	Overhead	Command	Shared Object	Symbol
#	.....	.....	.....	.....
	76.67%	perf	perf	[.] function2
	16.67%	perf	perf	[.] function1
	6.67%	perf	[kernel.kallsyms]	[k] do_page_fault



## Linux Tools - valgrind

- <http://valgrind.org/>
- Valgrind is an instrumentation framework for building dynamic analysis tools.
- A collection of tools for dynamic analysis:
  - Memcheck detects memory management problems
  - Cachegrind – a cache profiler
  - Massif – a heap profiler
  - Helgrind – thread debugger which finds data races in multithreaded programs



# Linux Tools - valgrind

```
$ valgrind --tool=memcheck --leak-check=yes ./test_valgrind
```

```
==20699== Use of uninitialised value of size 4
==20699==    at 0x2B1BFB: _itoa_word (in /lib/libc-2.5.so)
==20699==    by 0x2B5390: vfprintf (in /lib/libc-2.5.so)
==20699==    by 0x2BCE42: printf (in /lib/libc-2.5.so)
==20699==    by 0x80483F0: main (test_valgrind.c:9)

==20699== Invalid read of size 4
==20699==    at 0x8048406: main (test_valgrind.c:12)
==20699== Address 0x401608C is 4 bytes after a block of size 40 alloc'd
==20699==    at 0x40053C0: malloc (vg_replace_malloc.c:149)
==20699==    by 0x80483FC: main (test_valgrind.c:11)

==20699== 4 bytes in 1 blocks are definitely lost in loss record 1 of 2
==20699==    at 0x40053C0: malloc (vg_replace_malloc.c:149)
==20699==    by 0x80483D0: main (test_valgrind.c:7)
```





# Linux Tools - GDB

- <http://www.gnu.org/software/gdb/>
- what is going on `inside' another program while it executes -- or what another program was doing at the moment it crashed
  - current execution point context:
  - program's source code
  - registers
  - stack frames
  - program memory
  - variables
  - change program execution
  - Run control: run, suspend, step in/out/over



# Linux Tools - GDB

```
$ gdb
```

```
GNU gdb (Ubuntu/Linaro 7.4-2012.04-0ubuntu2) 7.4-2012.04  
Copyright (C) 2012 Free Software Foundation, Inc.
```

```
(gdb) run
```

```
Continuing.
```

```
Program received signal SIGSEGV, Segmentation fault.
```

```
0x10000488 in compute_data (number=10) at data.c:18
```

```
18          p_data->a = number;
```

GDB commands:

- run/Ctrl-C(stop), next, step, nexti, stepi, breakpoint
- backtrace, frame, print, x, dump
- set, call, jump, return



# Eclipse Linux Tools



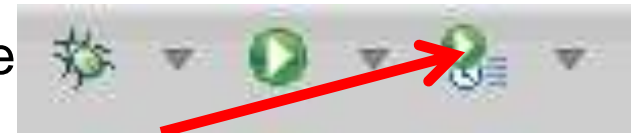
# Eclipse Linux Tools Project

- Full featured C/C++ IDE for Linux Developers
  - Project: build, editor, debugger (GDB based)
  - Linux Tools integrates Linux native tools:
    - Profiling: perf, oprofile, GProf
    - Dynamic analysis tools: valgrind
    - Tracing : LTTng, SystemTap
    - Other: RPM, ChangeLog, Man Page, Gcov



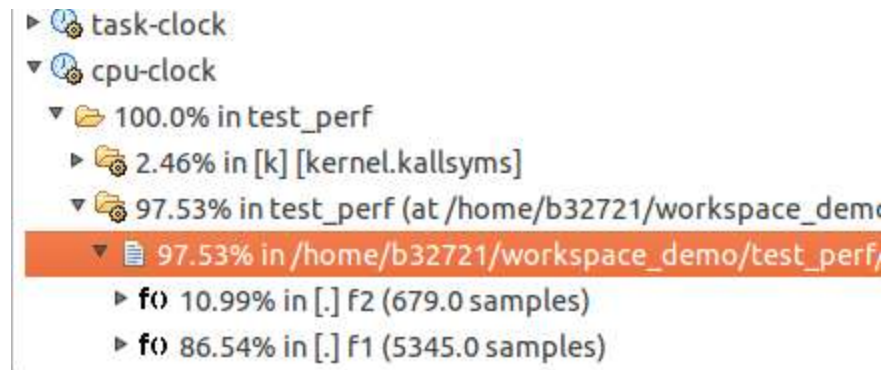
# Eclipse Linux Tools - perf

- Eclipse Linux Tools Project
- Eclipse – one click launch
  - Right click on the project->Profile As -> Profile



- Customized profiling – Profile configuration
  - Perf options
  - Perf event selection SW/HW: cycles, cache-misses, hw counters

- View the results
  - Perf profile view





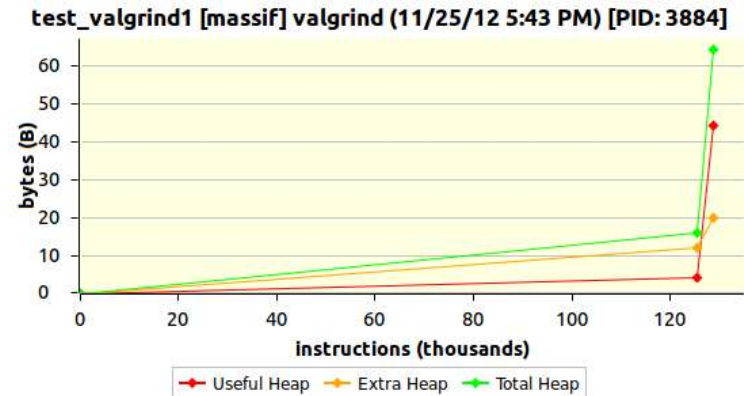
# Eclipse Linux Tools - valgrind

- A collection of tools for dynamic analysis:
  - Memcheck detects memory management problems
  - Cachegrind – a cache profiler
  - Massif – a heap profiler
  - Helgrind – thread debugger which finds data races in multithreaded programs
- One click launch and profile configuration
- View the results

```

▶ W Invalid read of size 4 [PID: 3946]
▼ W 4 bytes in 1 blocks are definitely lost in loss record 1 of 2 [PID: 3946]
    at 0x4028876: malloc (vg_replace_malloc.c:236)
    by 0x8048428: main (test_valgrind.c:7)
▶ W 40 bytes in 1 blocks are definitely lost in loss record 2 of 2 [PID: 3946]

```



test\_valgrind1 [massif] valgrind (11/25/12 5:43 PM)

Snapshot	Time (i)	Total (B)	Useful Heap (B)	Extra Heap (B)	Stacks (B)
0	0	0	0	0	0
1	125,448	16	4	12	0
2	128,753	64	44	20	0



# Eclipse Linux Tools - gdb

## GDB – Linux application debug

Resume (F8)

Quick Access

C/C++ Debug

Debug

test\_gdb [C/C++ Application]

- test\_gdb [6118] [cores: 0]
  - Thread [1] 6118 [core: 0] (Suspended : Breakpoint)
    - compute\_data() at test\_gdb.c:27 0x80483f4
    - main() at test\_gdb.c:22 0x80483d3

gdb

Variables

Name	Type	Value
number	int	3
p_data	struct data_t *	0x804a058

Breakpoints

- test\_gdb.c [line: 27]

No details to display for the current selection.

```
24 }
25 void compute_data(int number) {
26     struct data t *p_data = get_next_data();
27     p_data->a = number;
28     p_data->b = number % 256;
29 }
30 struct data t* get_next_data() {
31     static int cnt = 0;
32     if (cnt < 10)
33         return &my_data[cnt++];
```

