BKK19-114
EAS Unit Testing
Steve Muckle

Monday, April 1st 2019
Agenda

- motivation
- existing resources
- work in progress
motivation
common sources of customization

- cpufreq
- scheduler
typical validation

- power
- perf
- U/X
some issues observed

- schedutil gov not responding quickly
- spurious wakeups by schedutil gov
- stale CPU util values used by schedutil gov
- scheduler wakeup path logic
- sync flag functionality
- overutilized flag
- scheduler bias to prev cpu
existing resources
rt-tests.git

- backfire/sendme: signal latency from driver
- cyclicstest: sched latency
- hackbench: benchmark/load generator
- pi_stress: priority inheritance stress test
- pmqtest: measure POSIX msg queue latency
- ptsematest: measure POSIX mutex latency
- rt-migrate-test: ensure highest prio tasks run
- signaltest, sigwaittest: signal latency
- svsematest - SYSV semaphore latency
LTP

- **testcases/realtime**
  - sched latency, priority inheritance priority preemption, signal latency
- **testcases/kernel/sched**
  - bug fix verification (autogroup, nptl)
  - general stress tests (hackbench, process_stress, pth_str0*, sched_stress)
- **testcases/kernel/device-drivers/cpufreq**
  - one test to verify overclocking
- **testcases/kernel/power_management**
  - basic cpufreq, cpuidle, sched domain
- **testcases/kernel/syscalls**
kselftest

- cpufreq
  - stress governor switching
  - various driver/governor module insertion, removal cases
  - hotplug/scaling_min_freq torture test
LISA

- developed and maintained by ARM
- existing suite of EAS-oriented unit tests
  - eas placement
  - load tracking
- powerful analysis tools
- heavyweight
  - composed of multiple repos
  - integrated test runner, device support
  - know python for best results
  - has its own shell
work in progress
small standalone EAS tests

- C-based
- currently designed for LTP
- relies on its own trace parsing library
- merged into aosp/external/ltp
tests currently implemented

- eas
  - one small task
  - one big task
  - big to small
  - small to big
  - task toggles between small and big
  - 2 big tasks and 3 small tasks
tests currently implemented

- schedtune boost
  - manipulate boost values, observe task util
tests currently implemented

- schedutil governor
  - test latency to ramp to fmax then to fmin *
  - check num of sugov thread wakeups *
tests currently implemented

- generic scheduler tests
  - test task bw as affected by CFS priorities
  - test DL task exec correctness (runtime/deadlines)
  - test DL/RT scheduling latency
  - test RT FIFO/RR priority scheduling
current test results - pixel 3 w/aosp and 4.9

<table>
<thead>
<tr>
<th>test</th>
<th>pass/total</th>
</tr>
</thead>
<tbody>
<tr>
<td>eas_one_small_task</td>
<td>10/10</td>
</tr>
<tr>
<td>eas_one_big_task</td>
<td>10/10</td>
</tr>
<tr>
<td>eas_small_to_big</td>
<td>0/10 (upmigration latency = 115-130ms)</td>
</tr>
<tr>
<td>eas_big_to_small</td>
<td>9/10</td>
</tr>
<tr>
<td>eas_small_big_toggle</td>
<td>0/10 (upmigration latency = 130-140ms)</td>
</tr>
<tr>
<td>eas_two_big_three_small</td>
<td>3/10 (small tasks getting scheduled on big)</td>
</tr>
</tbody>
</table>
current test results - pixel 3 w/aosp and 4.9

<table>
<thead>
<tr>
<th>test</th>
<th>pass/total</th>
</tr>
</thead>
<tbody>
<tr>
<td>sugov_latency</td>
<td>10/10</td>
</tr>
<tr>
<td>sugov_wakeups</td>
<td>10/10</td>
</tr>
<tr>
<td>sched_boost</td>
<td>8/10</td>
</tr>
</tbody>
</table>
### current test results - pixel 3 w/aosp and 4.9

<table>
<thead>
<tr>
<th>test</th>
<th>pass/total</th>
</tr>
</thead>
<tbody>
<tr>
<td>sched_cfs_prio</td>
<td>10/10</td>
</tr>
<tr>
<td>sched_dl_runtime</td>
<td>10/10</td>
</tr>
<tr>
<td>sched_latency_dl</td>
<td>8/10</td>
</tr>
<tr>
<td>sched_latency_rt</td>
<td>9/10</td>
</tr>
<tr>
<td>sched_prio_3_fifo</td>
<td>9/10</td>
</tr>
<tr>
<td>sched_prio_3_rr</td>
<td>10/10</td>
</tr>
</tbody>
</table>
## current test results - hikey960 w/aosp and 4.14

<table>
<thead>
<tr>
<th>test</th>
<th>pass/total</th>
</tr>
</thead>
<tbody>
<tr>
<td>eas_one_small_task</td>
<td>0/10 (70%-80% on small CPU)</td>
</tr>
<tr>
<td>eas_one_big_task</td>
<td>10/10 (100% on big CPU)</td>
</tr>
<tr>
<td>eas_small_to_big</td>
<td>1/10 (small task runs on big too much)</td>
</tr>
<tr>
<td>eas_big_to_small</td>
<td>9/10</td>
</tr>
<tr>
<td>eas_small_big_toggle</td>
<td>3/10 (max upmigration time &gt; 100ms)</td>
</tr>
<tr>
<td>eas_two_big_three_small</td>
<td>0/10 (small tasks run on bigs too much)</td>
</tr>
</tbody>
</table>
## current test results - hikey960 w/aosp and 4.14

<table>
<thead>
<tr>
<th>test</th>
<th>pass/total</th>
</tr>
</thead>
<tbody>
<tr>
<td>sugov_latency</td>
<td>6/4 (latency right around 70ms)</td>
</tr>
<tr>
<td>sugov_wakeups</td>
<td>0/10 (150-580 wakeups/10sec)</td>
</tr>
<tr>
<td>sched_boost</td>
<td>10/10</td>
</tr>
</tbody>
</table>
## current test results - hikey960 w/aosp and 4.14

<table>
<thead>
<tr>
<th>test</th>
<th>pass/total</th>
</tr>
</thead>
<tbody>
<tr>
<td>sched_cfs_prio</td>
<td>10/10</td>
</tr>
<tr>
<td>sched_dl_runtime</td>
<td>10/10</td>
</tr>
<tr>
<td>sched_latency_dl</td>
<td>0/10 (latency 270-310 usec)</td>
</tr>
<tr>
<td>sched_latency_rt</td>
<td>10/10 (latency 6-23 usec)</td>
</tr>
<tr>
<td>sched_prio_3_fifo</td>
<td>10/10</td>
</tr>
<tr>
<td>sched_prio_3_rr</td>
<td>10/10</td>
</tr>
</tbody>
</table>
future plans

- exposure to more targets/topologies, support
- investigating failures
- improving robustness
- upstreaming
- more test cases
  - sync flag
- potential inclusion in VTS
- move to libtraceevent?
thank you