



Real-Time Linux

State of the union

Thomas Gleixner

RT-Summit 2017, Prague



Mission

- ☐ RT Mainlining
- ☐ Documentation
- ☐ Establishing community
- ☐ Long term maintenance

Funding

- ☐ Funding for 2017 is secured
- ☐ 1 Platinum member
- ☐ 5 Gold members
- ☐ 4 Silver members



Staffing

- 🗂 5 developers working on the project



Tasks accomplished

- ☐ CPU hotplug rework
- ☐ CPU hotplug locking rework

CPU hotplug

What's wrong?

- ❏ Known to be fragile
- ❏ Duct taped to death
- ❏ Completely fell apart with RT

CPU hotplug

First steps

- ❑ First attempt of redesign in 2012
- ❑ Conversion to a symmetric state machine

CPU hotplug

First steps

- ☐ Ran out of spare time
- ☐ Some \$corp promised to pick it up

CPU hotplug

First steps

- ☞ Ran out of spare time
- ☞ Some \$corp promised to pick it up
- ☞ ... applied more duct tape instead

CPU hotplug

This time for real

- ☞ Revisited in 2015
- ☞ Finished in 2017

CPU hotplug

Groundwork

- ☞ Analyzing all notifiers
- ☞ Documenting ordering requirements
- ☞ Fixing several dozens of existing bugs

CPU hotplug

Grumpwork

- ❏ One by one conversion of notifiers to states
- ❏ Gradual removal of old infrastructure

CPU hotplug

The locking nightmares

- ☞ Rework of hotplug locking
- ☞ Lockdep coverage unearthed yet another pile of bugs
- ☞ Curing is mostly finished by now

CPU hotplug

Lessons learned

- ☞ If you unearth existing bugs, you are expected to fix them
- ☞ The amount of crap you find is immense
- ☞ Don't assume you saw the worst already
- ☞ Don't give up if you have to rewrite stuff all over the place
- ☞ Never expect that corporates keep their promises

CPU hotplug

Lessons learned

- ☞ Estimation of effort is hard
- ☞ Total timespan off by factor 2
- ☞ Total work hours off by factor 3

CPU hotplug

Lessons learned

- ☞ Took 2 years time
- ☞ About 2 man years effort
- ☞ About 1.2 patches per work day

Tasks accomplished

- ☐ CPU hotplug rework
- ☐ CPU hotplug locking rework
- ☐ Timer wheel rework
- ☐ HRTimer rework
- ☐ Futex/RTmutex overhaul
- ☐ Separation of pagefault and preemption
- ☐ Cleanup of task affinity abuse
- ☐ Extending debug facilities



Tasks accomplished

- 📄 About 700 patches merged or queued for 4.15
- 📄 About 50 patches still under review
- 📄 Fixed 40 real and 80 latent bugs

Long term stable versions

- ❏ Linux 3.2 - EOL: May. 2018, rarely updated
- ❏ Linux 3.10 - EOL: Oct. 2017, rarely updated
- ❏ Linux 3.16 - EOL: Apr. 2020, rarely updated
- ❏ Linux 4.1 - EOL: May. 2018
- ❏ Linux 4.4 - EOL: Feb. 2022
- ❏ Linux 4.9 - EOL: Jan. 2019
- ❏ Linux 4.14 - EOL: TBD

Development version

- ❑ Linux 4.13
- ❑ Will be abandoned in favour of 4.14 (LTS)



Most complex development tasks

- ❑ dcache locking
- ❑ softirq modifications
- ❑ memory management interaction
- ❑ local locks and annotations

dcache locking

- ❑ Trylock loops
- ❑ Safe in mainline
- ❑ Lifelock prone in RT
- ❑ The most challenging issue right now



You want to help?

- 📁 Testing
- 📁 Documentation
- 📁 Task list in the RT wiki



Roadmap

Roadmap

Due to the evolutionary nature of Linux the roadmap will be published after the fact, but it will be a very precise roadmap.