

# Linktime Optimizatoin BOF

# Changes since last year

- Symbol table is more on less on place
- Partitioning cleaned up and made more robust
- Symbol renaming happens only when necessary
- Type merging rewritten to tree merging and significantly sped up  
(thanks to Richard Biener)
- Inliner heuristic now handle better big programs
- Devirtualization now use one-definition-rule
- Number of bugfixes (we can build more stuff!)

# What works

- Bootstrap
  - We should avoid rebuilding every compiler binary twice for checksum
  - We should default to `-fno-fat-lto-objects` (needs libtool update?)
  - We should avoid exporting all symbols for plugins
- Spec2k/Spec2k6
  - Even with nice performance & size gains
- Firefox (joint work with Taras Glek, needs small patch for symbols used by asm statements)
- Chromium (thanks to Martin Liška, also needs patch)
- Libreoffice (thanks to Martin Liška)
- Linux Kernel (thanks to Andi Kleen, needs moderate patch with some hacks)

# What works

- Bootstrap
  - We should avoid rebuilding every compiler binary twice for checksum
  - We should default to `-fno-fat-lto-objects` (needs libtool update?)
  - We should avoid exporting all symbols for plugins
- Spec2k/Spec2k6
  - Even with nice performance & size gains
- Firefox (joint work with Taras Glek, needs small patch for symbols used by asm statements)
- Chromium (thanks to Martin Liška, also needs patch)
- Libreoffice (thanks to Martin Liška)
- Linux Kernel (thanks to Andi Kleen, needs moderate patch with some hacks)

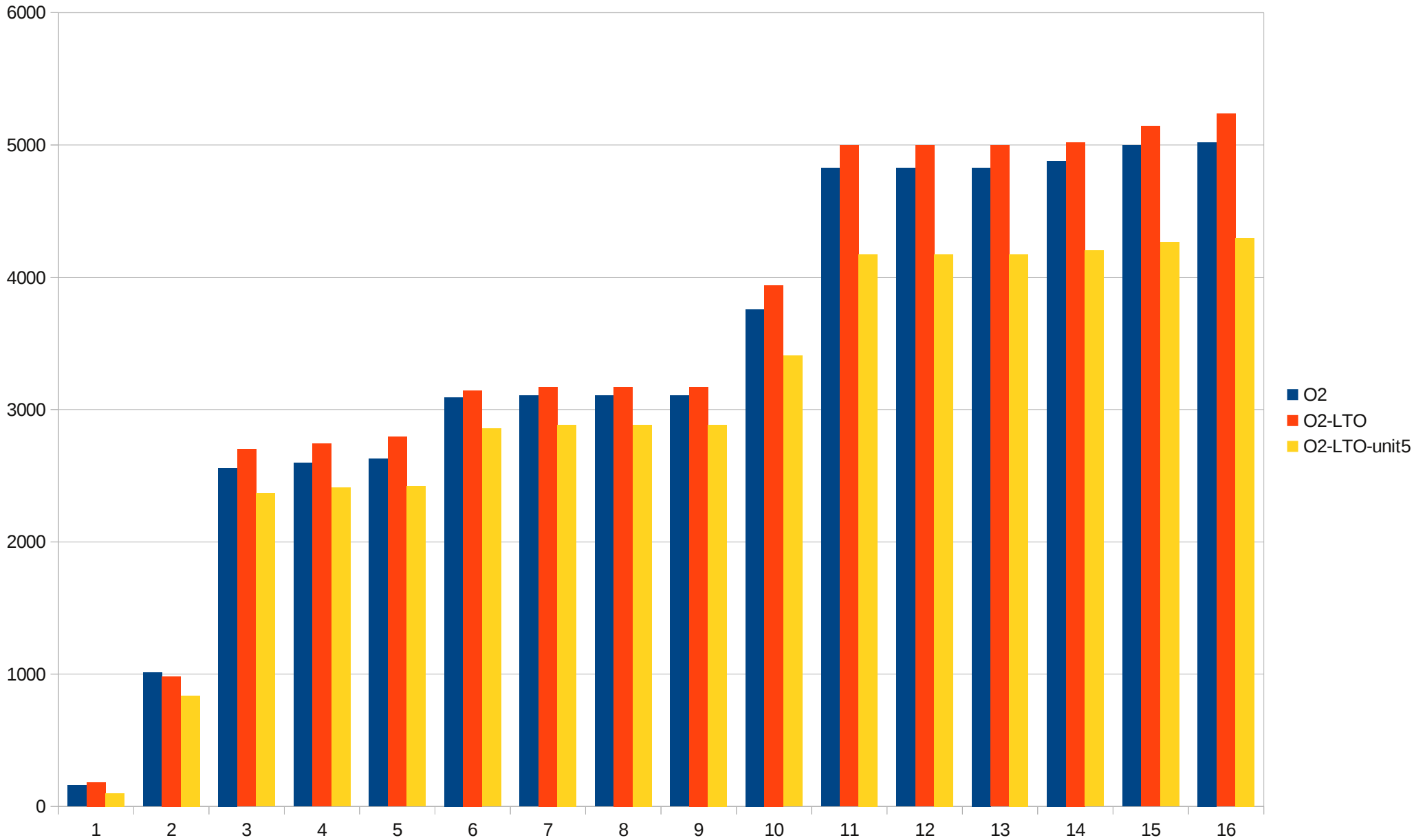
# How well it works?

Data courtesy of Martin Liška

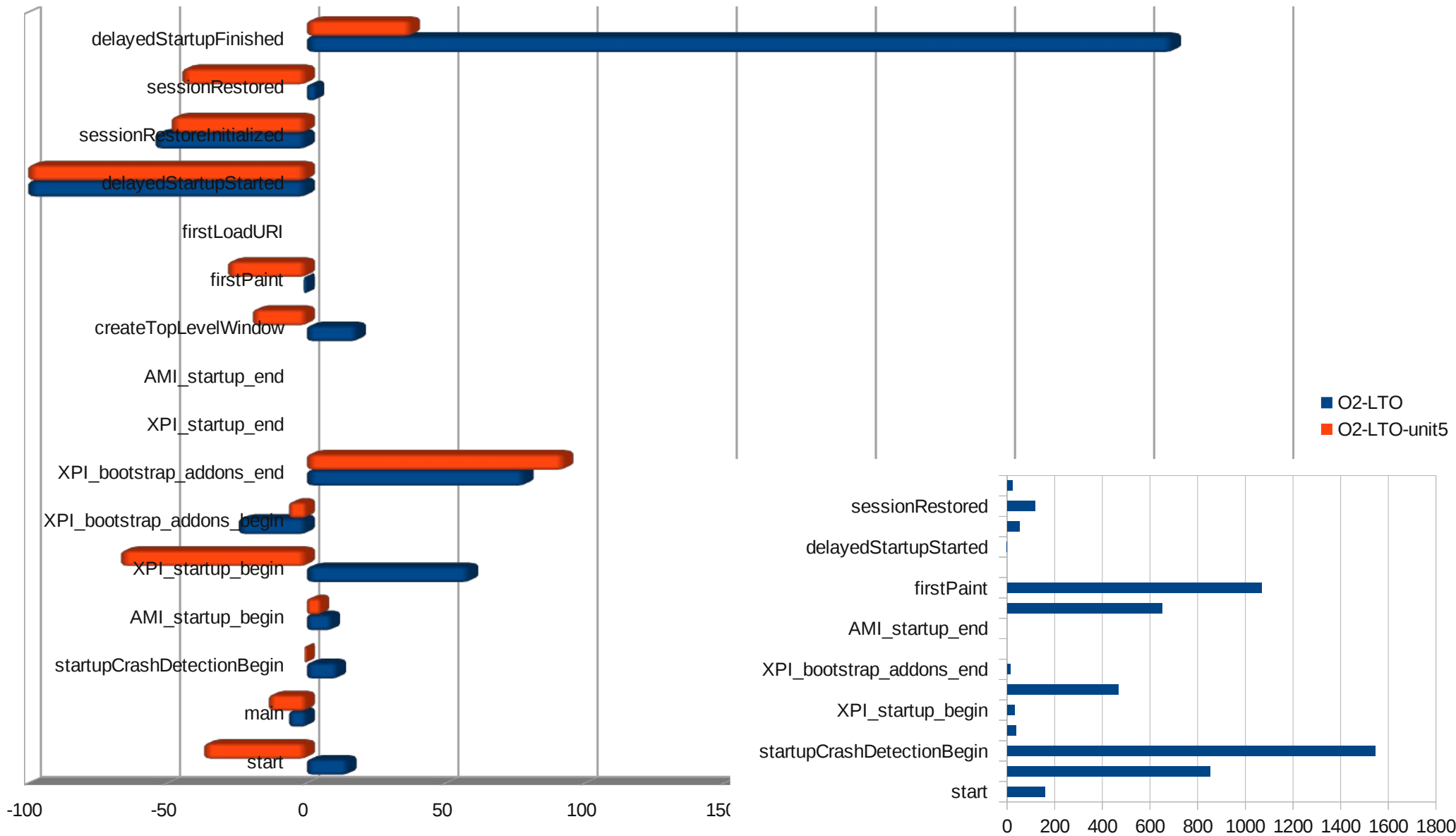
# Firefox sizes

- Non-LTO libxul 50.2MB, 5.0s startup
- LTO libxul 51.85MB, 5.2s startup
- LTO libxul, unit-growth=46.08MB, 4.2s startup

# Firefox telemetry times

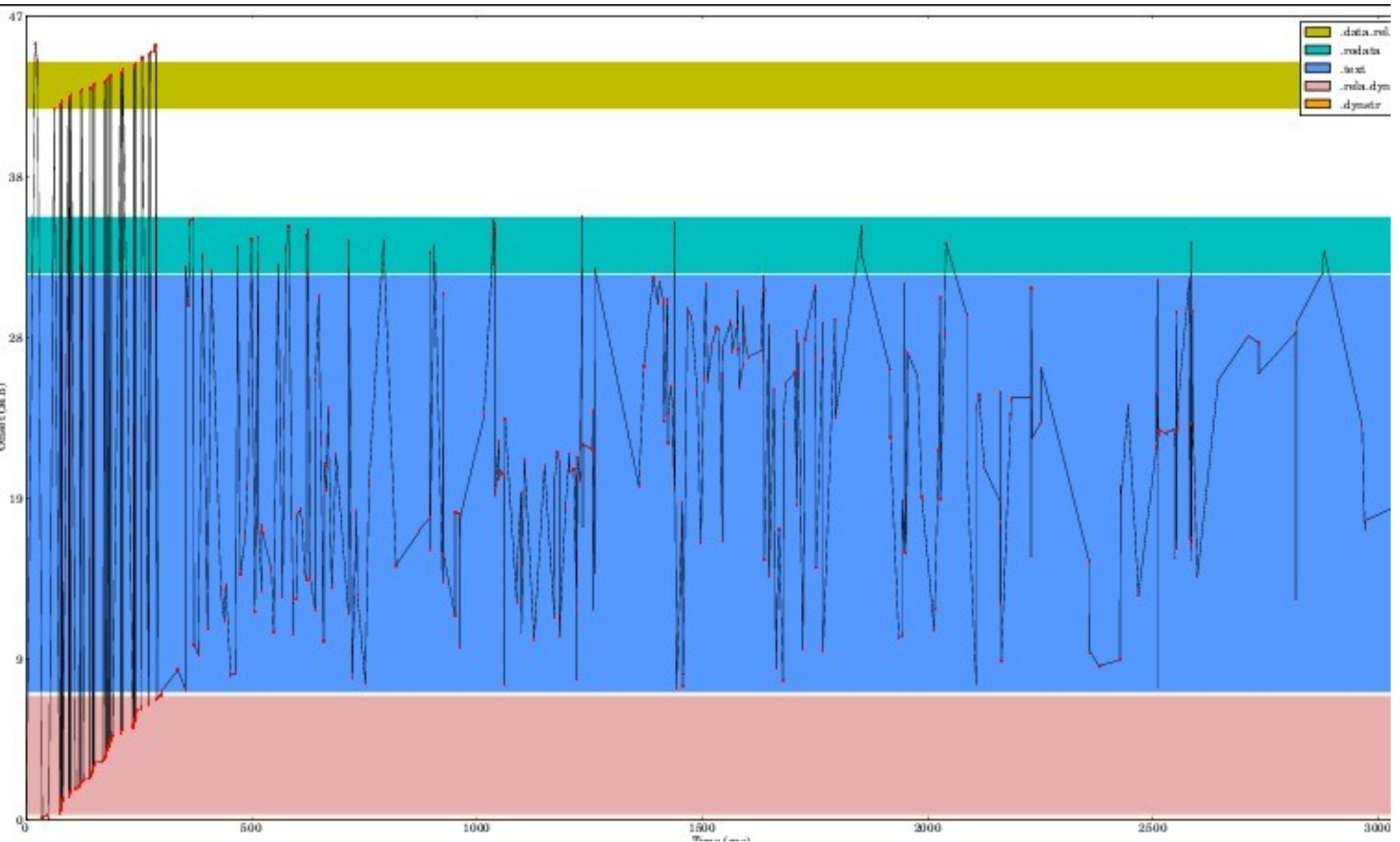


# Slowdowns of individual startup stages by telemetry (and absolute times)





# Firefox startup disk accesses



# TODOS

- Solve merging objects with multiple parameters
  - Currently affects Firefox
- Add `no_reorder` function attribute
  - Needed by Linux Kernel
- Relax section type conflicts? (Andi asked for that)
- Reduce number of trees needed by WPA (Firefox needs 12MB, WPA is still streaming bound for C++)
- Solve debug info issues via early debug?
- Reduce `/tmp` disk usage (currently we need 5GB for Firefox)
- Start doing LTO bootstrap for release builds?
- Default to `-fno-fat-lto-objects`
- `gcc-nm/gcc-ar` wrappers are unhandy. Can binutils load multiple plugins, can we install plugin on the default binutils search path?
- Can we stop holding `stderr` until end of plugin execution?
- One definition rule used for type merging & warnings issues when it is violated?