Unifying trace processing ecosystems with Babeltrace

Linux Plumbers 2019 – Tracing MC (September 9th 2019)
Jérémie Galarneau <jgalar@efficios.com>
**jgalar** on GitHub/IRC

Babeltrace

- Consume, manipulate, convert traces of various formats
  - CTF, perf, pcap, userspace traces/logs, HW traces, etc.
- Correlate multiple trace, from multiple hosts
- Already used in kernel tree (perf to CTF)
- Cross-platform: Linux, macOS, Windows

https://github.com/efficios/babeltrace

Components assembled in a graph

Not unlike gStreamer, ffmpeg, DirectShow, etc.
Split functionality into components and build tools on top.
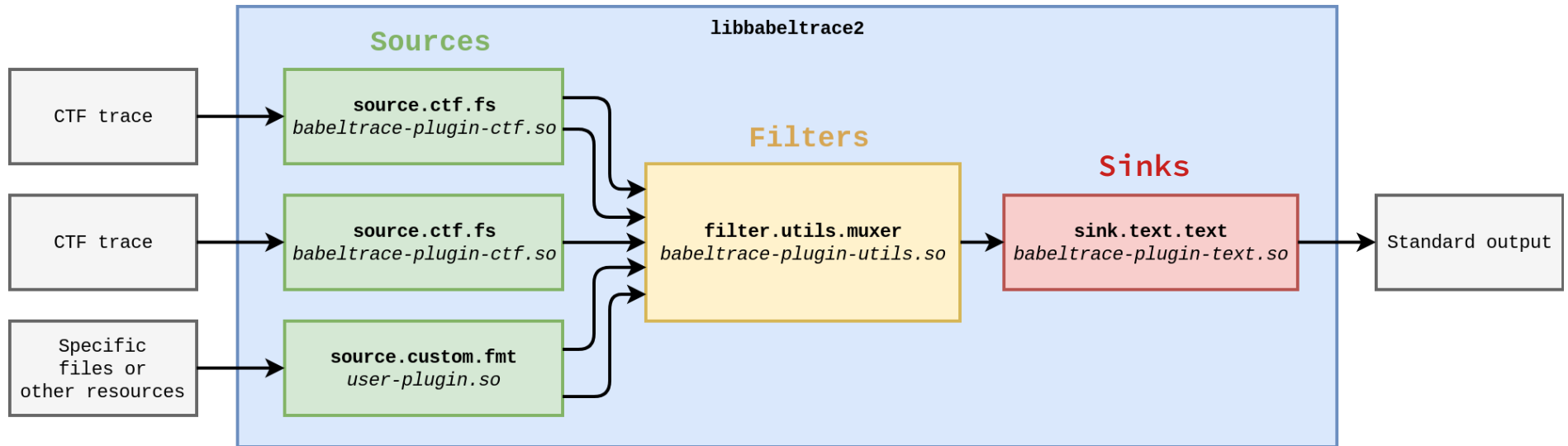
Status update

- All expected API changes for 2.0 are **done**.
- **Documentation** is being written.
- Already usable for all intents and purposes.
- **RC1** is expected within a few weeks.

Discussion

- Trace format stability / interface
  - Source in babeltrace's tree or shipped by tracer?
- Provide common or "correlated" clock sources
  - e.g. LTTng uses `CLOCK_MONOTONIC`, provides boot id, offset to Epoch
  - Trickier for HW traces... external sync is often needed
- Make trace analysis tools tracer-agnostic
  - Live tracing tools (bpftrace) become available offline
  - perf-tools, Trace Compass, Kernel Shark, etc.