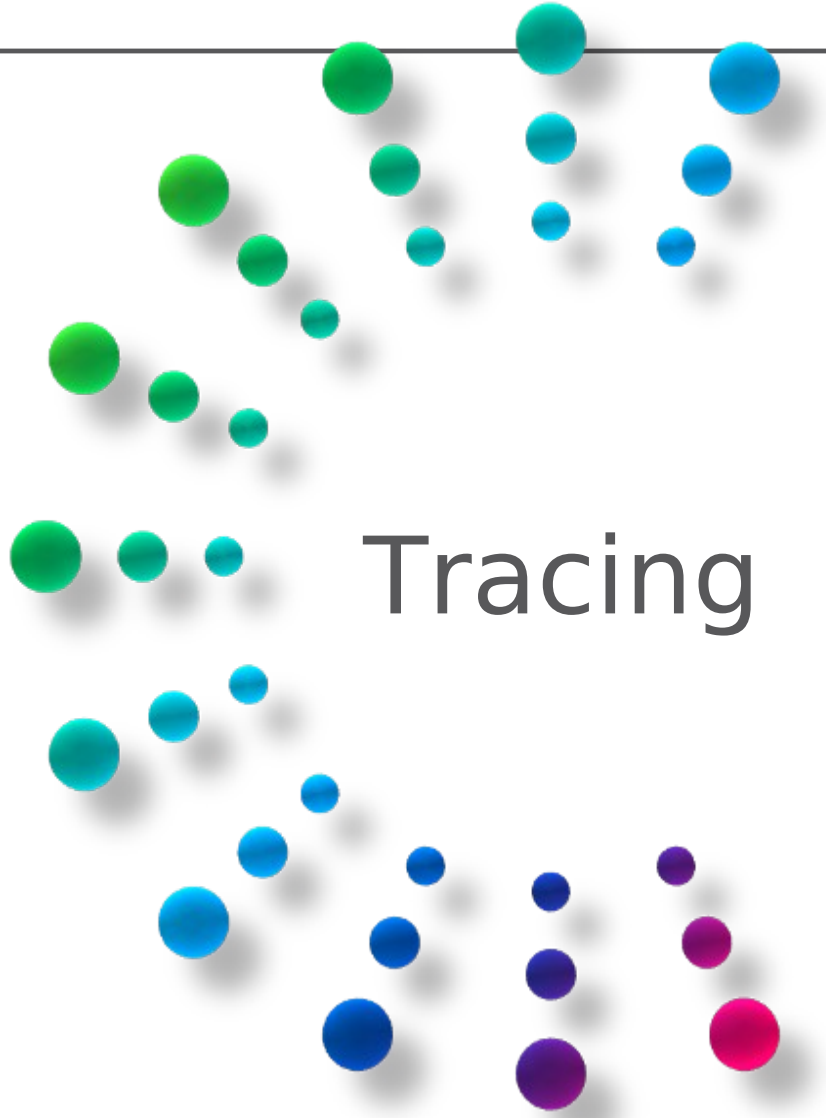
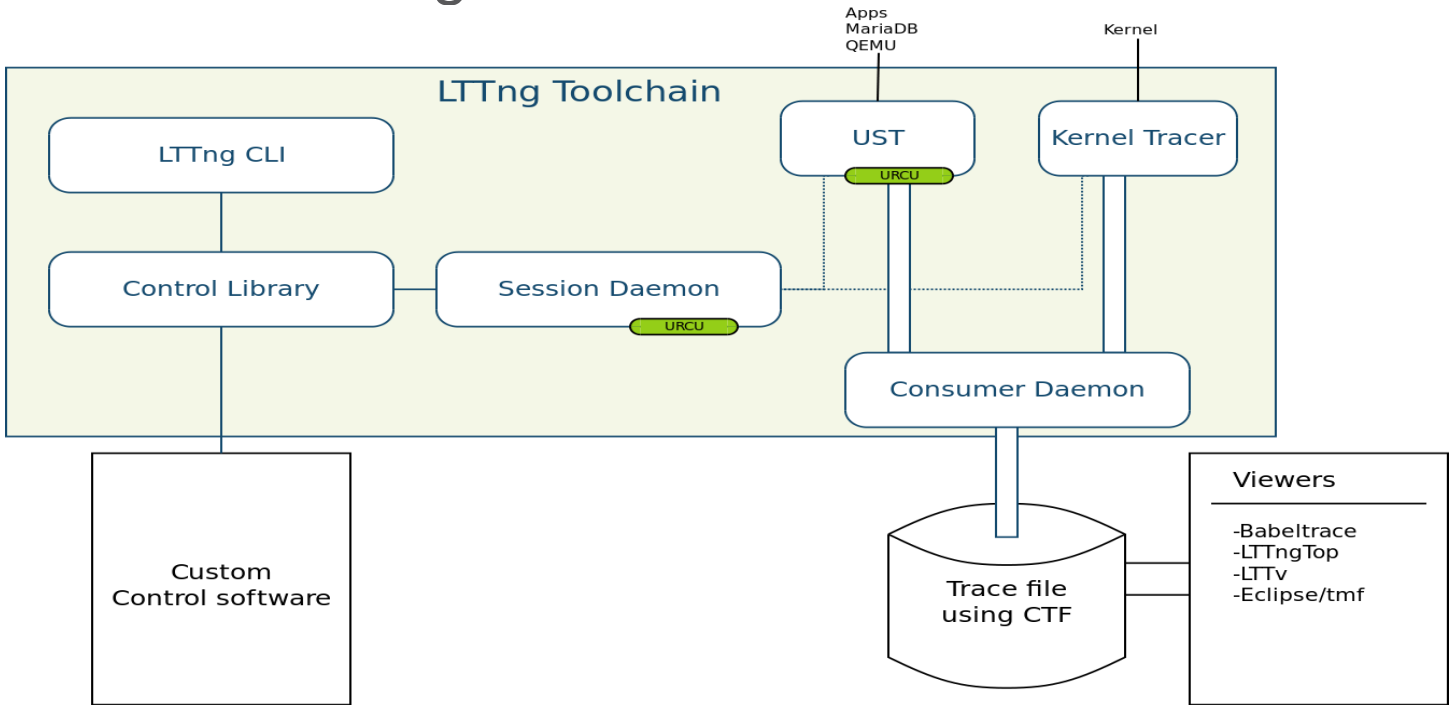


Tracing

Matthew Khouzam – IM&T Tools Delivery (Ericsson)

LTTng distinctive features

- Multi-session support with per host or per user daemon
- Algorithms based on RCU verified by model checking
- Designed to meet real-time constraints
- Supports live streaming of the trace data



Common Trace Format

- Ericsson and Linux Foundation CE Linux Workgroup
- Reviewed by Linux kernel developers and SystemTAP communities
- In collaboration with
  Multi-Core Association Tool Infrastructure Workgroup

  Freescale, Mentor Graphics, IBM, IMEC, National Instruments, Nokia Siemens Networks, Samsung, Texas Instruments, Tilera, Wind River, University of Houston, Polytechnique Montréal, University of Utah, …

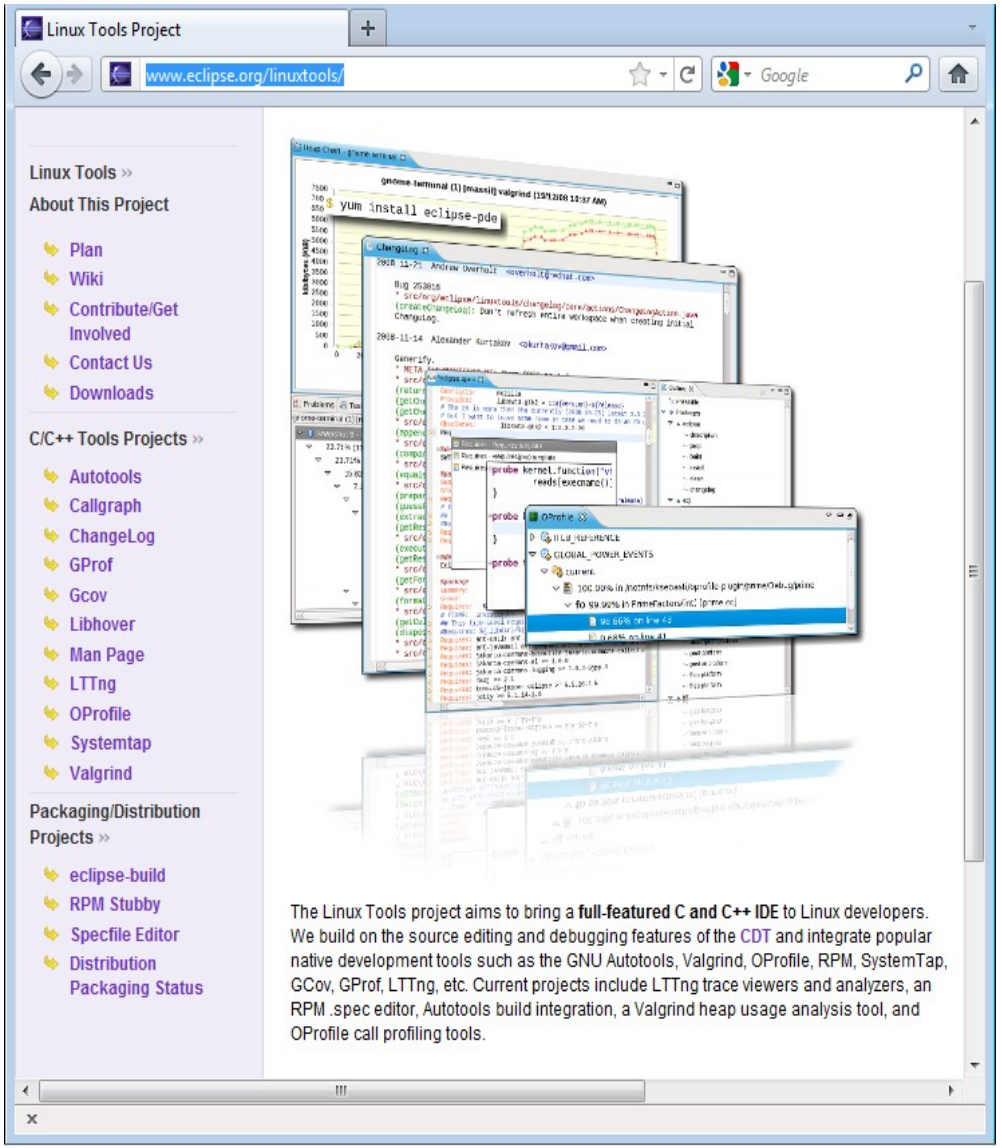
- Requirement, specification, reference implementation http://www.efficios.com/ctf

Common Trace Format

- Self describing
- Very compact binary trace format
- System-wide and multi-system trace representation in a common format, for integrated analysis:
  - Software traces
    - Across multiple CPUs
    - Across the software stack, e.g. hypervisor, kernel, library, applications
  - Hardware traces
    - DSPs, device-specific tracing components.
    - GPUs.

Eclipse Tracing monitoring framework (TMF)

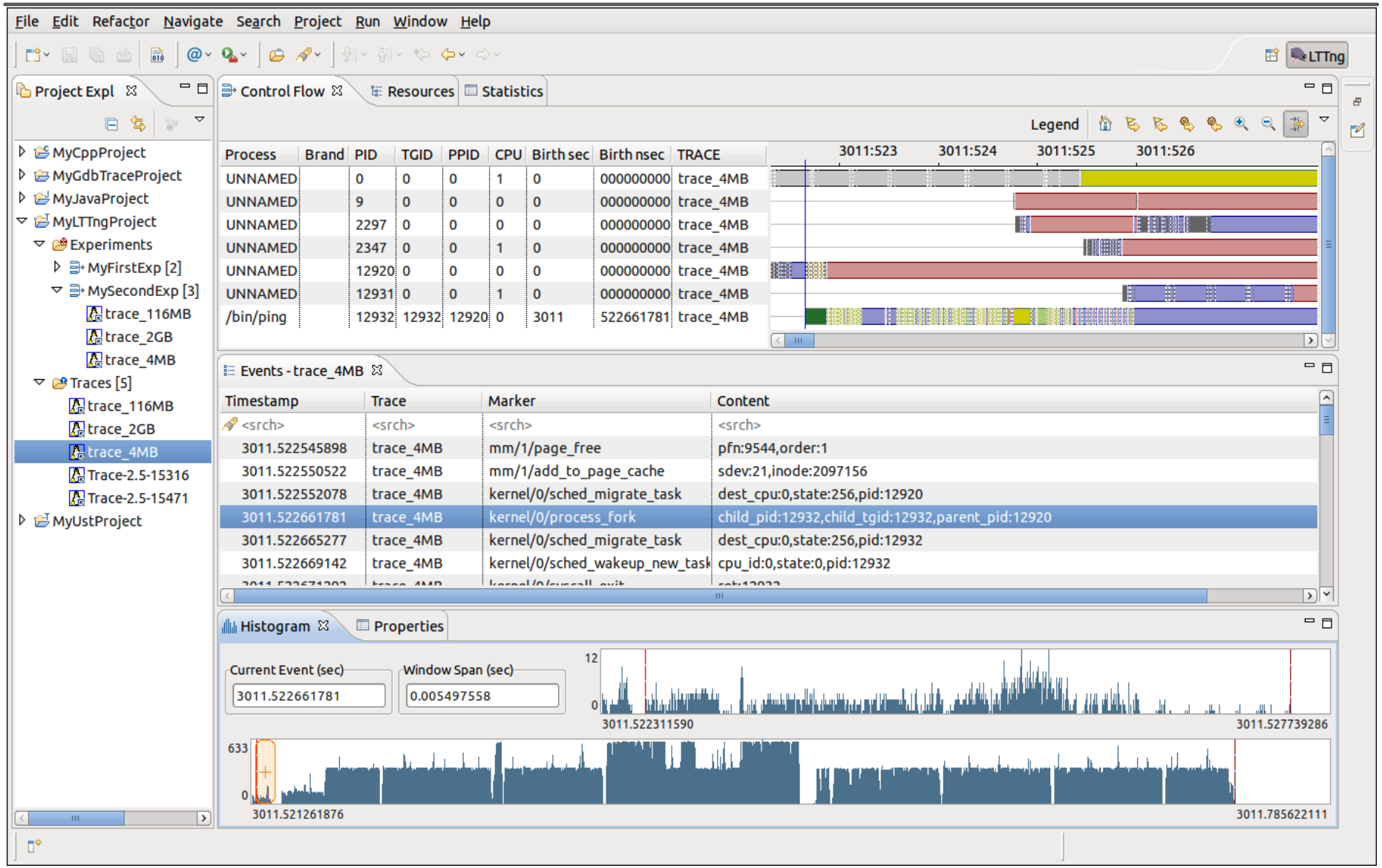
- Eclipse Linux Tools Project
- Framework to build trace visualization and analysis tool
- Scalability allows to handle traces exceeding memory
- Enable trace analysis from different sources
- LTTng Eclipse integration is an implementation on top of TMF



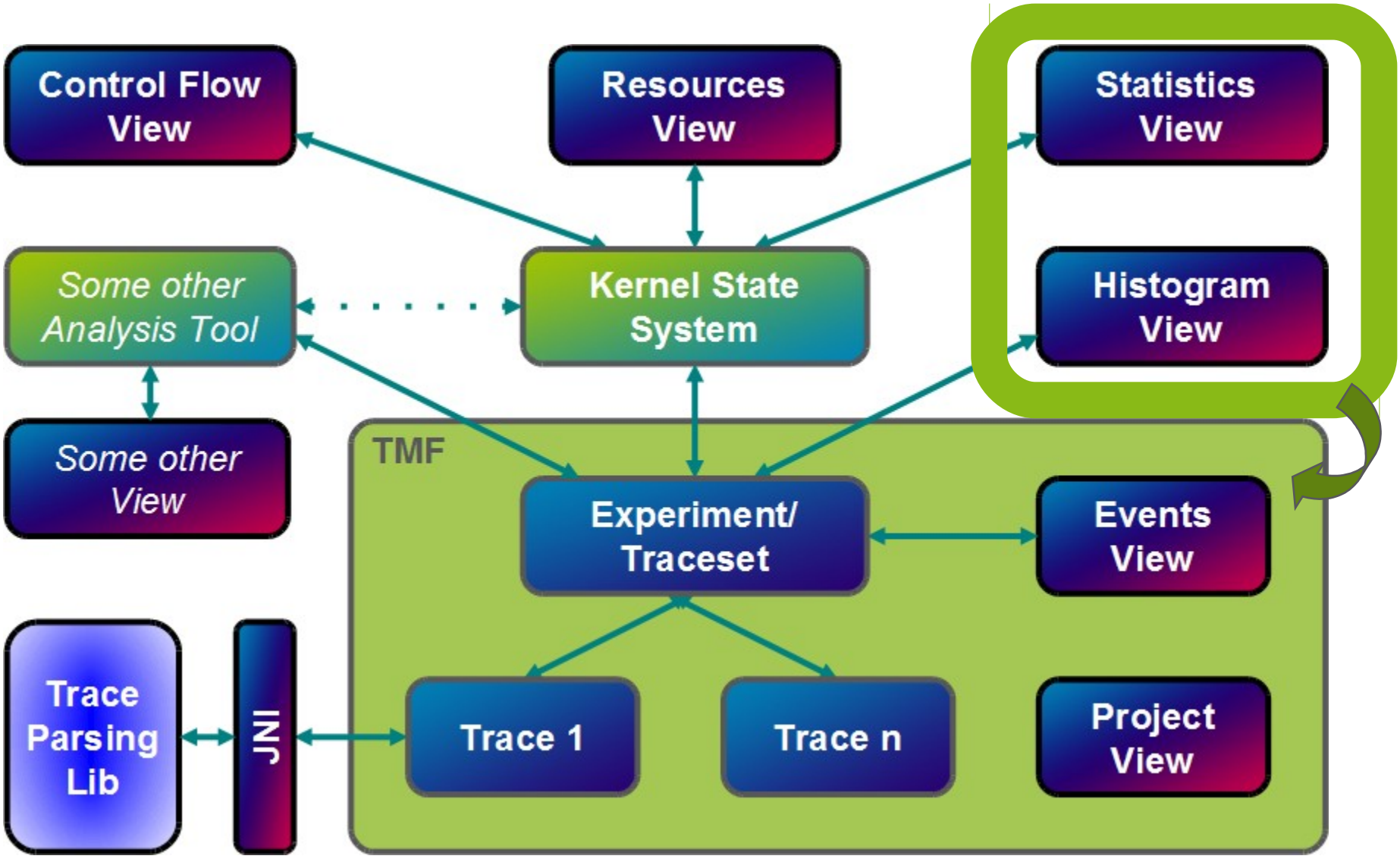
What the framework provides

- A trace and event data model
- Extension point to add new trace types
- Reusable views and widgets
- Integration into common navigator framework (e.g. project explorer)
- An event filter model
- Time window and event synchronization
- Ability to hook own analysis tools
- Custom text & XML parser wizards (no code required!)
- Trace control and streaming for LTTng 0.x (extra steps required)

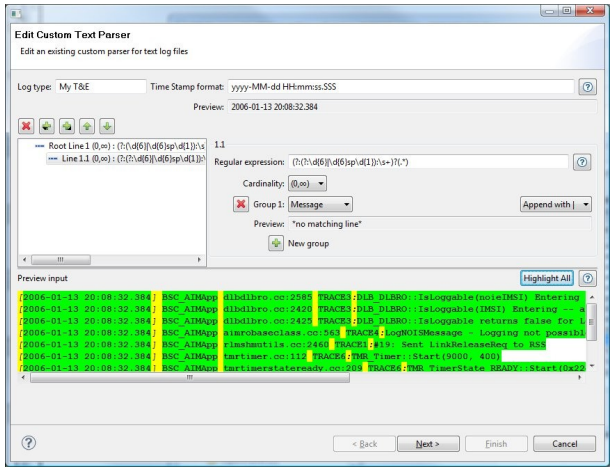
Lttng perspective



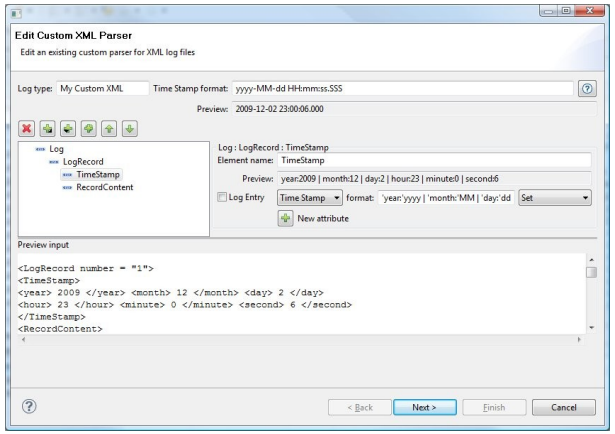
Eclipse Tracing Monitoring Framework & LTTNG architecture



Custom text Parser Wizards

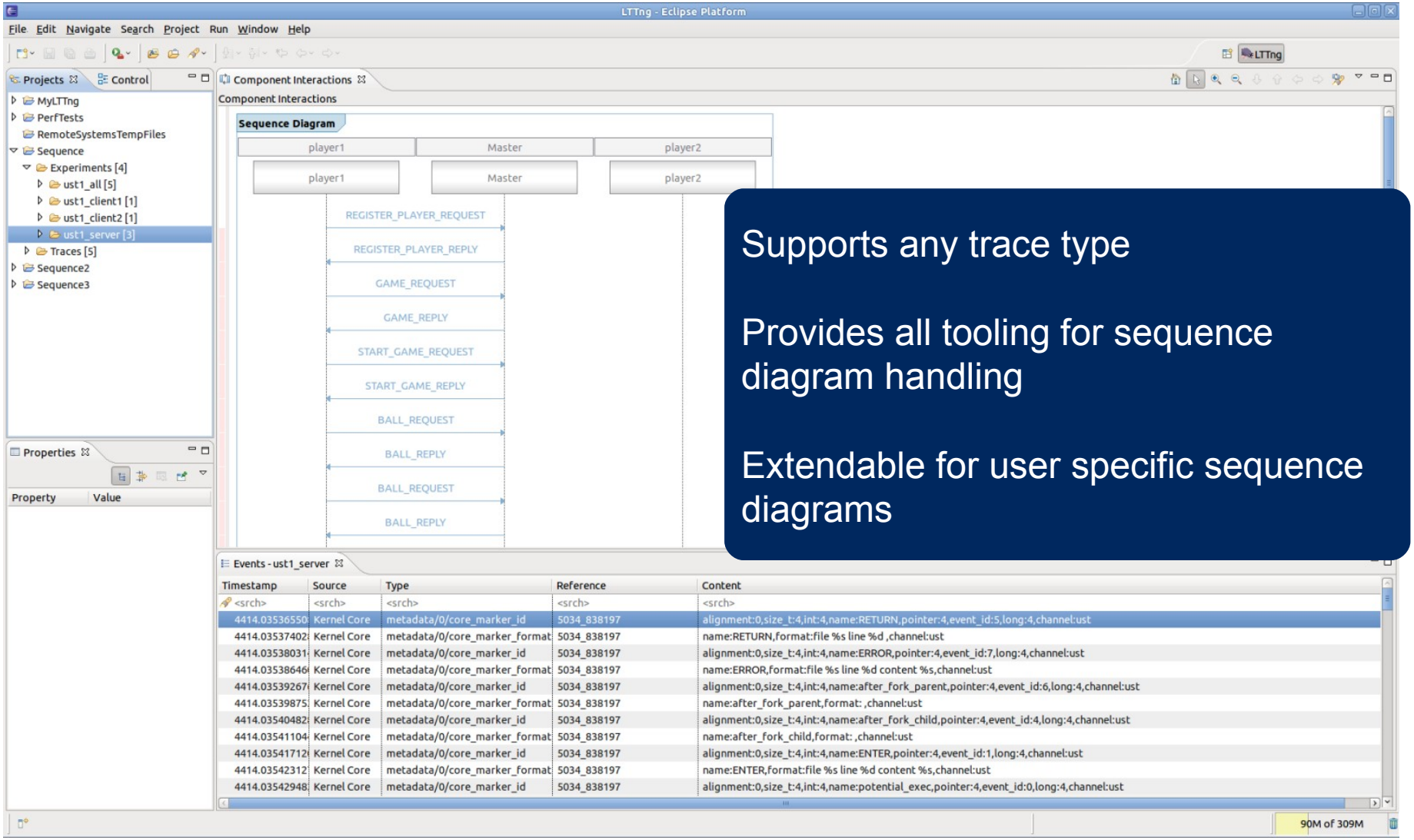


- Custom Text Parsers
  - line based parser with regex
  - allows user to define own parser with extracted data and output fields
  - parser definition created and edited with a wizard
  - parser definitions can be shared by importing / exporting to file



- Custom XML Parsers
  - XML based extracting data from XML elements and their attributes

Sequence diagram framework



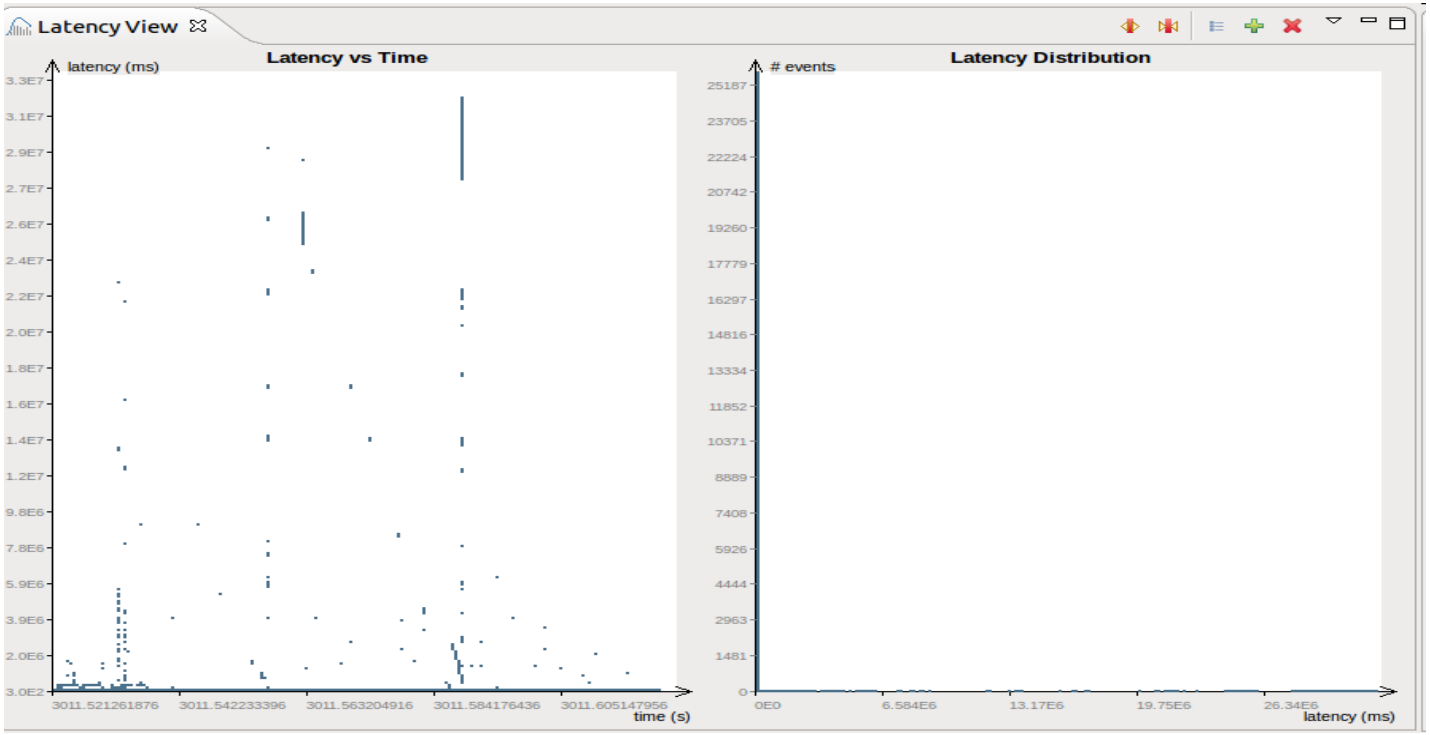
Supports any trace type

Provides all tooling for sequence diagram handling

Extendable for user specific sequence diagrams

Latency Analysis

- For visualizing latency statistics between specific LTTng kernel trace events (Open source soon)



Current development

- Support for LTTng 2.0
  - CTF-based Kernel and UST traces
  - Trace control
  - Session management
  - Support for multiple trace sessions
  - Streaming
- Juno
  - Project graduation work (Linux Tools 1.0)
  - Uplift to Eclipse 4.X
- More analysis tools
  - A better state system.
  - Clock adjustment, trace comparison, etc.

Therefore

- The new state machine integrated into eclipse will allow a higher level of analysis with better abstraction. Saving YOUR time.
- And of course 2012 will be the year of tracing....
  - And the Linux desktop!

Some References

ERICSSON

- Linux Tools
  - Linux Tools: http://www.eclipse.org/linuxtools/index.php
  - Update Site: http://download.eclipse.org/technology/linuxtools/update
- LTTng (Eclipse)
  - LTTng Eclipse Project: http://www.eclipse.org/linuxtools/projectPages/lttng
  - LTTng Eclipse Wiki: http://wiki.eclipse.org/Linux_Tools_Project/LTTng
  - LTTng User Guide: http://wiki.eclipse.org/Linux_Tools_Project/LTTng/User_Guide
  - TMF User Guide: http://wiki.eclipse.org/Linux_Tools_Project/TMF/User_Guide
- LTTng project: http://lttng.org
- For more info, questions, discussions:
  - linuxtools-dev@eclipse.org

Q&A