

KernelShark 1.0

Transforming the GUI into a toolkit

Yordan Karadzhov

VMware Inc. - OSTC

What is KernelShark?

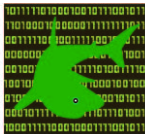


KERNELSHARK

- * **Front end reader of Linux kernel tracing data (Ftrace)**

- * The original version - started in 2009.
- * Written in *Gtk+-2.0*
- * Main goal: analyse and fully understood the performance of the Real-time scheduler.

What is KernelShark?



KERNELSHARK

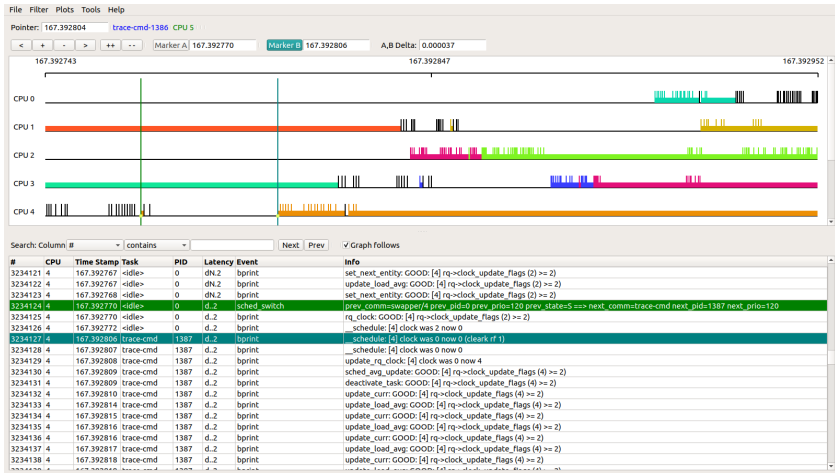
- * **Front end reader of Linux kernel tracing data (Ftrace)**

- * **New KernelShark:** use all lessons learned from the old version.
- * Completely rewritten to use Qt.
- * But not only this ...

The New KernelShark is

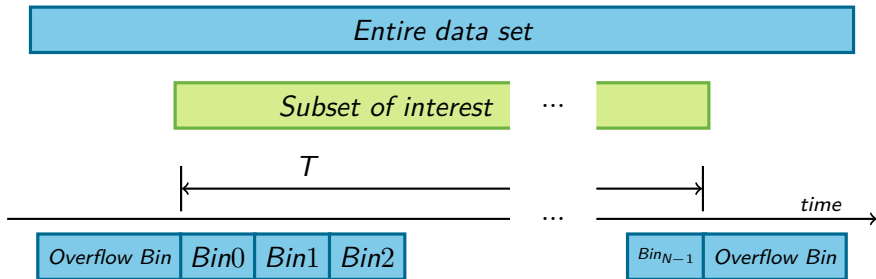
- a. Optimized for processing significantly larger amounts of data.
- b. New scalable data model - $\log(n)$ time complexity.
- c. OpenGL-based visualization.
- d. Preconfigurable - Json config I/O.
- e. User modifiable - plugins.

Kernel Shark



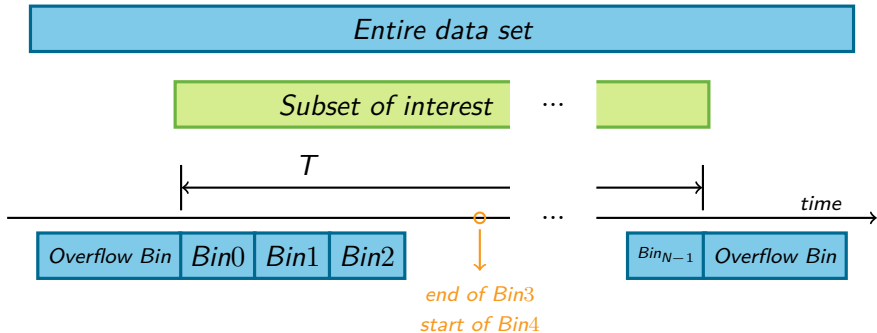
- * Nothing Revolutionary
- * A number of small improvements.

Visualization model - How does it work?



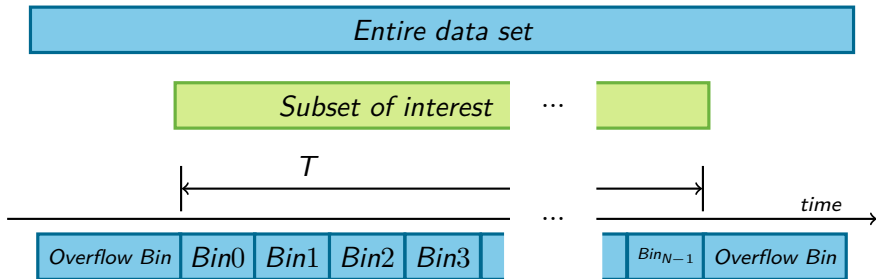
- Break the data-set into **time-bins** \mapsto like a histogram.
- Check only the records at the beginning and at the end of each bin.
 \mapsto constant time.

Visualization model - How does it work?



- Have the trace records, sorted in time.
- Knowing the index of the first record in each *Bin* determines the state of the model.
- But the first element can be found with a binary search $\mapsto \log(n)$ complexity.

Visualization model - How does it work?

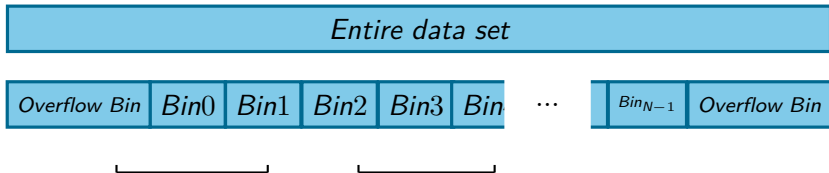


Data Binning provides $O(\log_2(n))$ average time complexity of all operations of the model.

Visualization model & tracing data formats

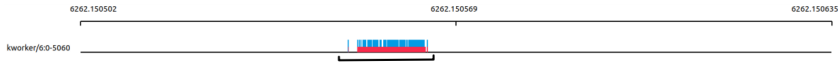
- The KernelShark Visualization model is not coupled to a particular data format.
- Uses KernelShark-specific data structure.
- Contains only the absolute minimum of information need by the model.
- The rest of the information - available on demand (can be slow)

Visualization model & tracing data



- Only one model (data structure) for all graphs.
- Worst-case complexity becomes linear.
- Solution - Data collections.

Visualization model & Data collections.



- Only one model (data structure) for all graphs.
- Worst-case complexity becomes linear.
- Solution - Data collections.

DEMO

Plugins

- Very powerful.
- Can overwrite the content of the data.
- Can plot on top of the existing graphs.

KernelShark: current version 0.9

<https://git.kernel.org/pub/scm/utils/trace-cmd/trace-cmd.git/>

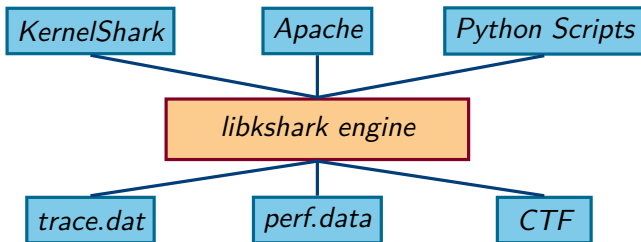
To build the code follow the instructions in

[/trace-cmd/kernel-shark-qt/README](#)

and

[/trace-cmd/README](#)

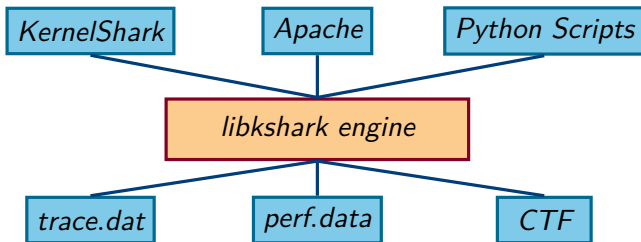
KernelShark is not a GUI. KernelShark is a toolkit.



What's next after KernelShark 1.0?

- KernelShark engine (libkshark.so)
- Available under GNU LGPL v2.1
- Highly customizable (via plugins)
- Will read multiple data formats

KernelShark is not a GUI. KernelShark is a toolkit.



What's next after KernelShark 1.0?

- Any tool will be able to use the library
- Available for Python applications (*libkshark.py*)
- The KernelShark application is just a “shell”.