



ellexus

How to speed up your EDA flows

Accelerating Verification NMI 2014

Richard Jordan
Ellexus Ltd

Introduction

How to speed up your EDA flows



ellexus

Ellexus make tools to help you solve the problems that arise when deploying and tuning complex Linux applications

This is done through dependency analysis and IO profiling

The company was founded to improve the way we optimise our EDA flows, but we now work across the HPC space.

This presentation is mainly based on the work that we have been doing with ARM and other HPC customers

We have been working with them to detect common misconfiguration problems and to find IO bottlenecks

Who am I?

Disclaimer: this is not my day job



ellexus

I am a senior developer at Ellexus

I work on our dependency tracing and profiling technology and on our build and test procedures

I am giving this presentation on behalf of our CEO, Rosemary Francis

I will try to answer your questions as best I can, but if there is anything I can't answer I can put you in touch with Rosemary afterwards

Dependency Analysis

File-system tracing and profiling



ellexus

Breeze

Traces file system dependencies
Profiles file IO and network IO

Originally designed to solve installation and config problems

Breeze is now mainly used to solve performance issues and optimise complex flows

ARM

Synopsys

Cadence

Cancer Research UK

Sanger Institute

Ansys

Breeze

Tracing the application



ellexus

The screenshot displays the Ellexus Breeze application interface. The main window is titled "Ellexus - Breeze, sViz and *Trace". The "Graph View" shows a call graph for "firefox". The graph starts with a "firefox" node (blue) that reads a "firefox" node (green). This green node then executes several other nodes: "sed", "sed", "basename", "uname", "mozilla-xremote-client", "mkdir", "mozilla-plugin-config", and "run-mozilla.sh". The "mozilla-plugin-config" node reads a "mozilla-plugin-config" node (blue) and executes "plugin-config" (red) and "uname" (orange). The "run-mozilla.sh" node reads a "run-mozilla.sh" node (blue) and executes "uname", "firefox", "basename", and "dirname".

The "Event View" on the right shows a list of events for the process "/usr/lib64/firefox/firefox - pid_26580 pid_26685". The events are as follows:

Time	Source PId	Event	Target
0us	26580	Exec'ed by	/usr/lib64/firefox/run-mozilla
7ms 775us	26580	ReadFileInfo	/usr/lib64/firefox/ (Directory
19ms 442us	26580	ReadFileInfo	/usr/lib64/xulrunner-2/libxpc
19ms 746us	26580	Open	/usr/lib64/firefox/application
24ms 139us	26580	Read	/usr/lib64/firefox/application
24ms 257us	26580	ReadFileInfo	/usr/lib64/xulrunner-2/libxpc
24ms 367us	26580	Open	/usr/lib64/xulrunner-2/deper
30ms 804us	26580	Read	/usr/lib64/xulrunner-2/deper
31ms 223us	26580	Load	/usr/lib64/xulrunner-2/libmo
31ms 525us	26580	Load	/usr/lib64/xulrunner-2/libmo
74ms 720us	26580	Load	/lib64/libasound.so.2 (Share

The "Event View" also includes a timeline visualization showing the sequence of events over time, with a scale from 0us to 4s. A specific event at 17ms 241us is highlighted in purple.

The "Commandline (bash)" window at the bottom shows the command `trace firefox` being executed in a terminal session at `/home/rosemary>`. The interface also includes a "Select run:" dropdown menu showing "17 - /usr/bin/firefox" and a "Find Next" button in the "Event View".

Breeze

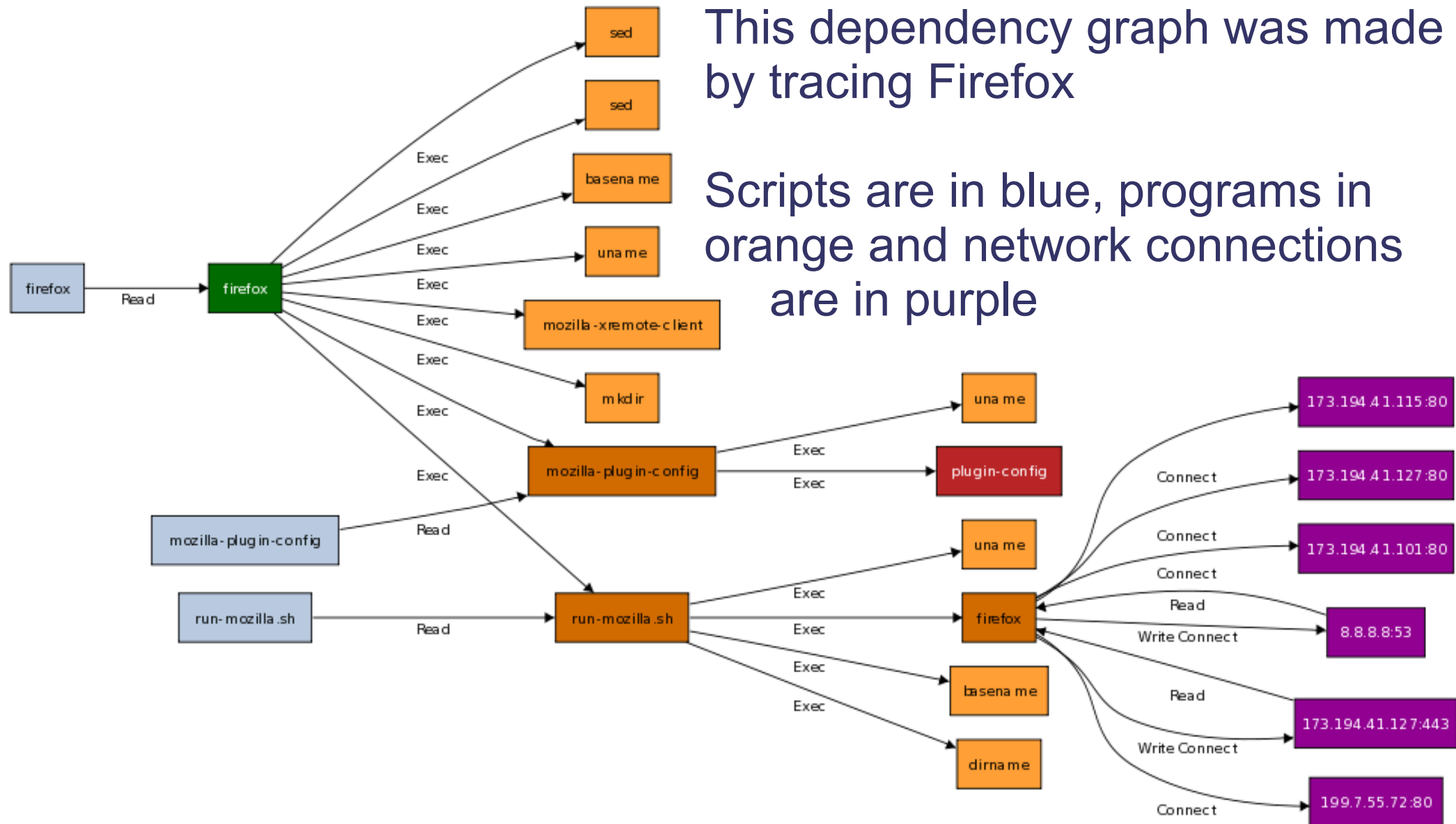
Tracing applications like Firefox



ellexus

This dependency graph was made by tracing Firefox

Scripts are in blue, programs in orange and network connections are in purple





ellexus

How to speed up your EDA flows

Accelerate your flows

Know where your files are



ellexus

Are your files stored in the right place?

Storing temporary files in shared storage can affect scalability
eg in your /home directory

Reading files repeatedly or one byte at a time can be slow

Searching for files can be slow and can affect other programs

It's not just about performance, it's about **reliable** performance

Customer case study

Using shared storage for temporary files

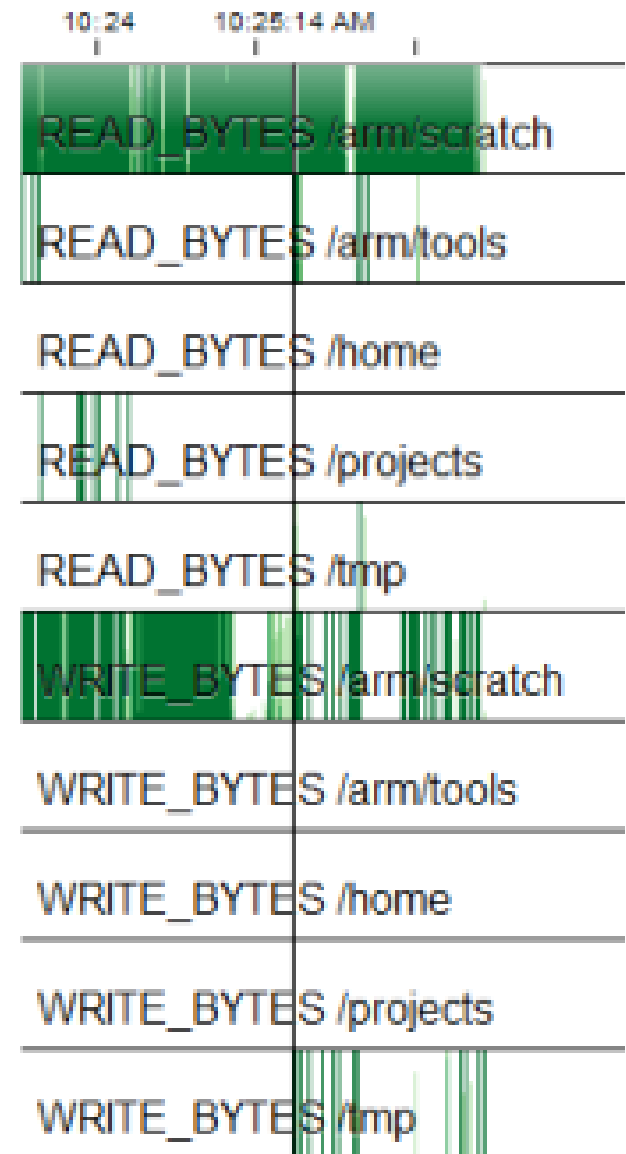


ellexus

This is one of the traces that ARM shared with us

Lots of data is written to
`/arm/scratch`

Almost nothing is written to
`/tmp`



Accelerate your flows

Think about your PATH set up



ellexus

The way your PATH variable is set up affects the way applications search for resources.

`export PATH=/home/richard/tools;${PATH}` wrong!

`export PATH=${PATH};/home/richard/tools` right 😊

Customer case study

Calling system programs on the PATH

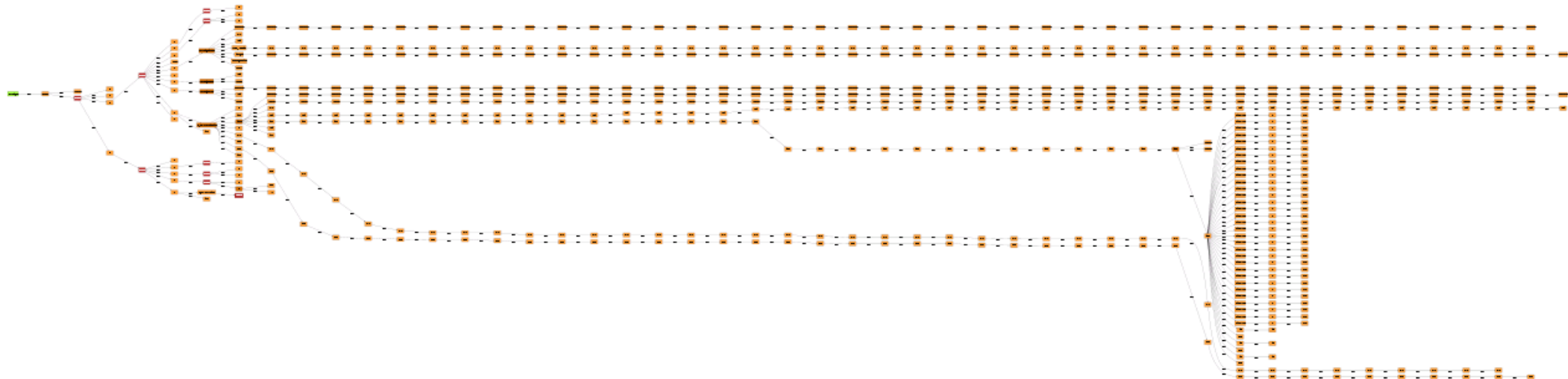


ellexus

The graph below is part of a real EDA flow where the PATH was set up the wrong way

They had about 50 paths in their infrastructure

The whole file system was searched every time a local program was called (eg grep, uname)



Accelerate your flows



ellexus

Detecting file system trawls

Here we show file system trawls for a scripted flow on local disk and on NFS

The trawl length is the number of failed open() calls before the file is found

On local disk **2ms** are wasted
On NFS **36ms** are wasted

This was one run on a quiet network with two dedicated machines

Trawl Length	Local disk	NFS
5	46	748
6	57	695
7	121	829
8	82	771
9	90	991
10	96	990
11	110	950
12	117	877
13	163	5042
13	228	5063
9	201	4341
5	46	1195
6	57	924
7	67	1362
8	80	1361
9	248	1749
10	187	1936
11	108	1744
12	123	2291
13	131	2389
Wasted time (uS)	2358	36248

Accelerate your flows

File system configuration



ellexus

A good file system should hide a lot of mistakes

... but file system design is very complex

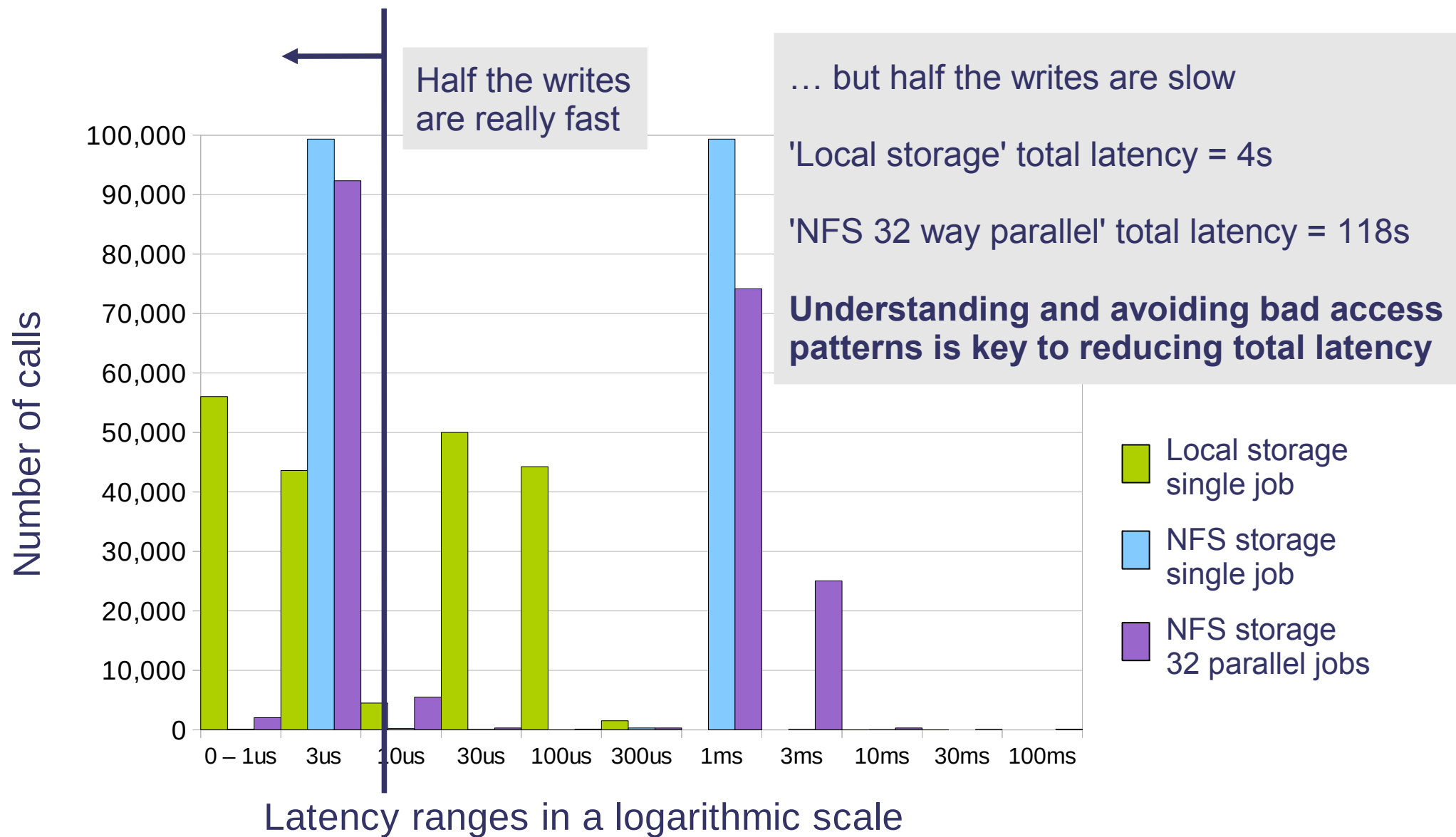
It is easy for your applications to hit a worst case access pattern

Accelerate your flows



ellexus

Write call latency on local and NFS storage



Accelerate your flows

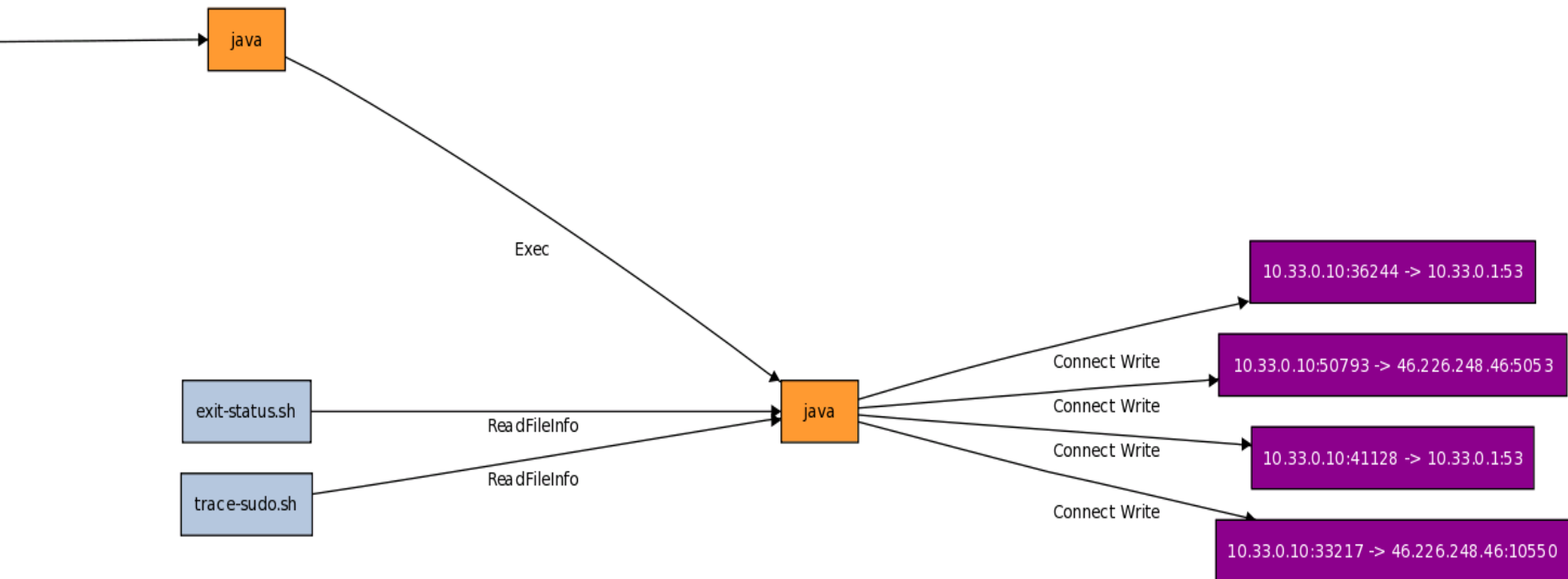
License configuration




ellexus

How much compute time is wasted on the license server?

Are all the settings in your license setup valid or are there servers that no longer exist?





Good Tools
+
Good Infrastructure  Good Performance

File systems are complicated

It is easy to misuse your infrastructure

No program is executed in isolation

Understanding your applications and scripted flows is key



ellexus

Thanks for listening