Emulation at Very Large Scale with Distem

Tomasz Buchert, Emmanuel Jeanvoine, Lucas Nussbaum

Agenda

Large-scale systems

Large-scale systems are:

- difficult to manage
- faulty
- stress platforms to their limits

Parallel command execution

How to execute a command on a set of nodes efficiently?

Parallel command execution consists of:

1. running a command on each given node
2. collecting output, including return codes

Easy to understand, but difficult to perform.

Parallel command execution (2)

Parallel command execution is a building block:

- verify connectivity
- install software
- launch and stop services
- monitor

However, as size of the system increases, problems appear:

- failing nodes
- time complexity of the operation
- storage and memory limitations
- network saturation

Parallel command execution (2)

Parallel command execution is a building block:

- verify connectivity
- install software
- launch and stop services
- monitor

However, as size of the system increases, problems appear:

- failing nodes
- time complexity of the operation
- storage and memory limitations
- **network saturation**

ARP protocol

The ARP protocol is used to map IP addresses to MAC addresses:

- fundamental protocol in IPv4/Ethernet stack
- very simple
- intrusive
- unscalable:
  - small tables in hardware (e.g., 8,192 entries)
  - small tables in software (by default)
  - a *cache stampede*
  - similar to *congestive collapse*

```
Dec 20 00:41:01 fw03 kernel: Neighbour table overflow.
Dec 20 00:41:01 fw03 kernel: printk: 100 messages suppressed.
```

Virtual Extensible LAN

VXLAN is a network virtualization technology:

- still a draft (last updated: 2014-05-20)
- created by VMWare, Arista Networks and Cisco
- plays between Layer 2 and Layer 3
- encapsulates Ethernet frames in UDP packets
- uses multicast to contain broadcast and multicast
- available in Linux 3.7 or newer

Advantages:

- usage of the existing IP network infrastructure
- many more VLANs than 802.1Q (2^{24} vs. 2^{12})
- no addresses of machines present in Ethernet switches

Introducing: Distem

Distem is an emulator for distributed systems:

- take your real application
- prepare your system parameters with Distem
- run your application inside emulated platform

Distem features

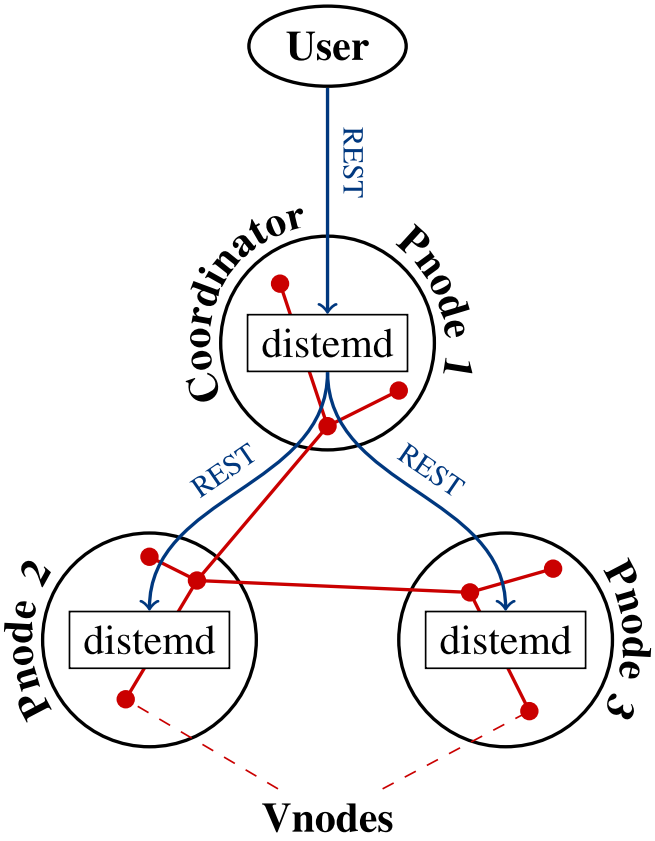
The features of Distem include:

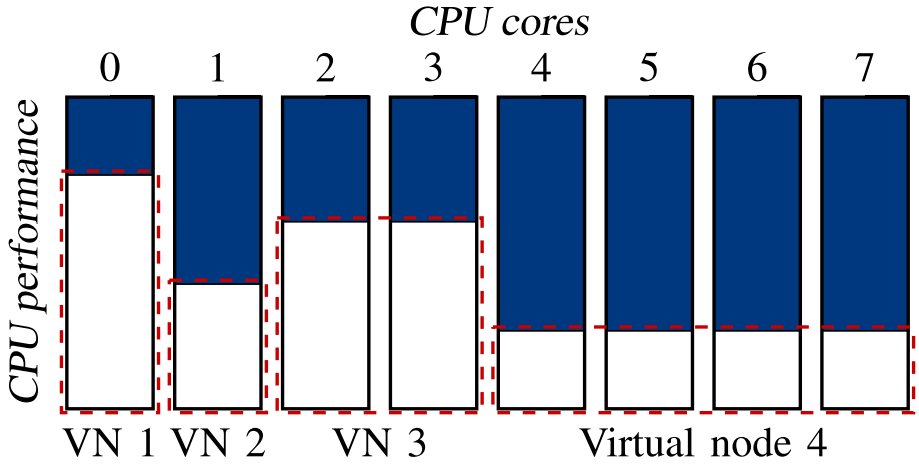
- running many *virtual nodes* an each *physical node*
- emulation of CPU performance
- emulation of network topologies

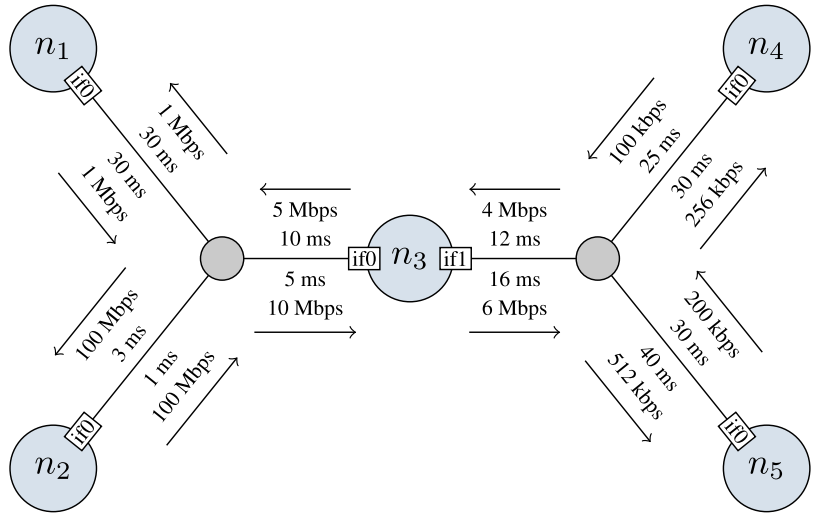
Distem uses modern Linux functionality:

- Linux containers
- control groups
- CPU frequency scaling
- traffic control

Distem features





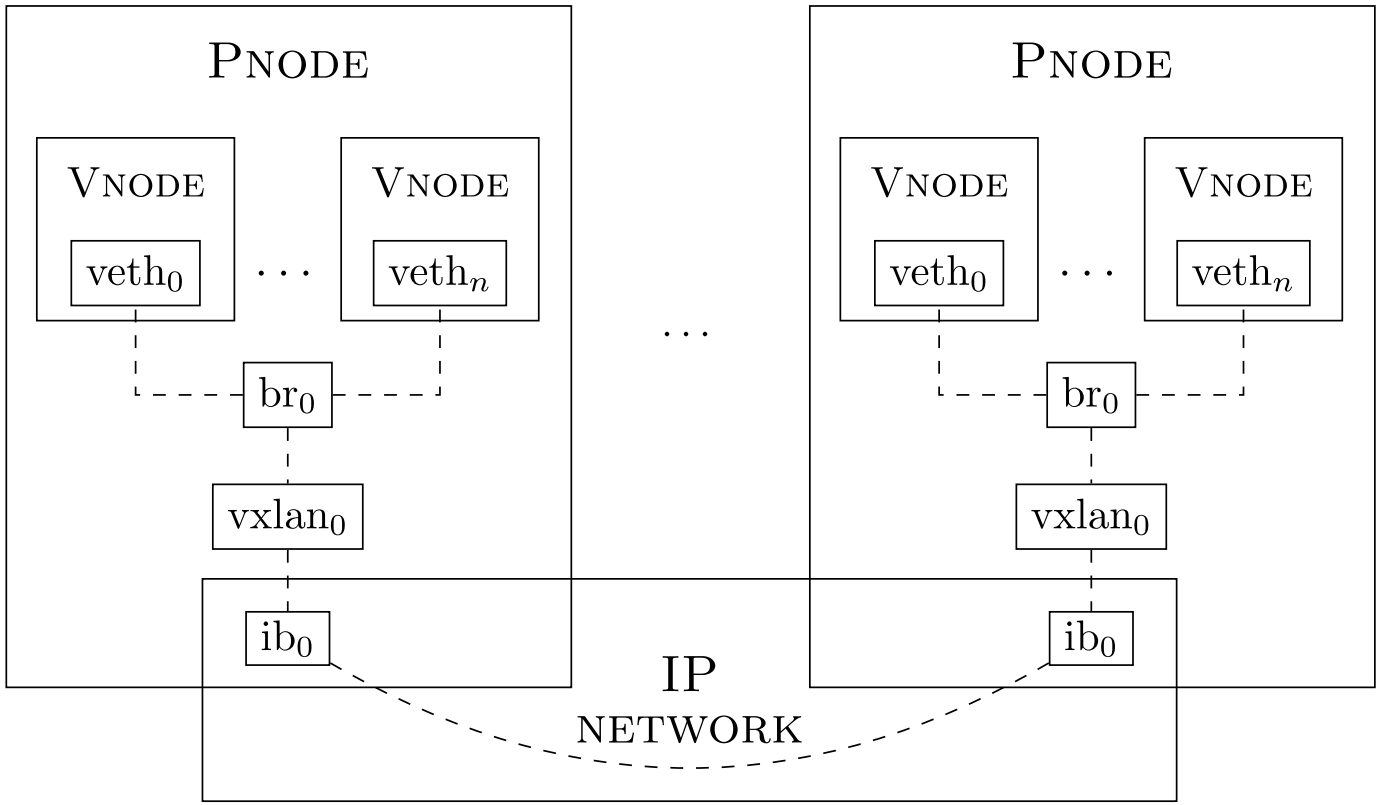


VXLAN in Distem

We extended Distem to use VXLAN:

- each physical node has a VXLAN interface
- VXLAN interface is bridged with interfaces of virtual nodes
- physical network does not know about virtual nodes

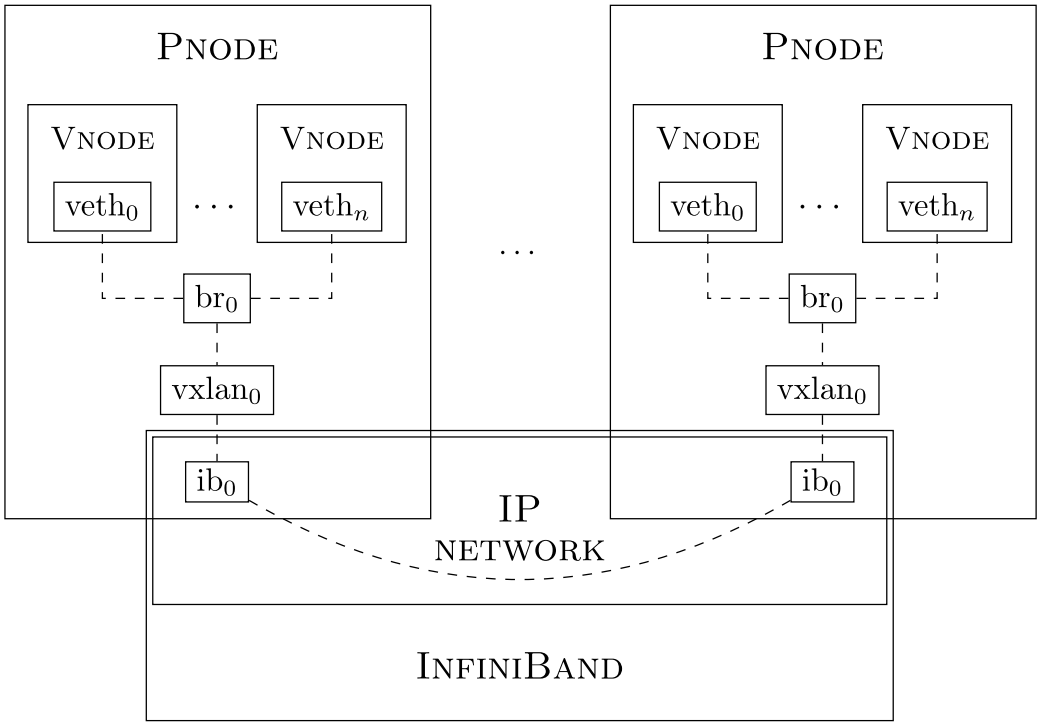
Network stack



Use of InfiniBand

VXLAN carries aggregated traffic of all nodes.

To ensure low latency and high throughput we use IP over InfiniBand (20Gbit/s).



Experimental setup

We study execution time of parallel command execution.

The command `true` is run on varying number of nodes and three methods:

- Clush (sliding window)
- TakTuk with arity 2 (tree topology)
- TakTuk with arity 3 (tree topology)

Each measure is repeated 3 times.

Experimental setup - platform

The hardware setup is:

- Grid'5000 platform (Nancy site)
- two clusters with InfiniBand:
  - *Graphene*: 144 nodes (1 CPU Intel X3440 @2.53 GHz, 4 cores/CPU, 16GB RAM on each node)
  - *Griffon*: 92 nodes (2 CPU Intel L5420 @2.50GHz, 4 cores/CPU, 16GB RAM on each node)
- 20Gbit/s InfiniBand interconnect
- Debian Jessie, Linux 3.12

Distem setup

The Distem setup is:

- 162 physical nodes
- 246 virtual nodes per each
- 39,852 nodes in total
- a shared filesystem
- no network nor CPU emulation

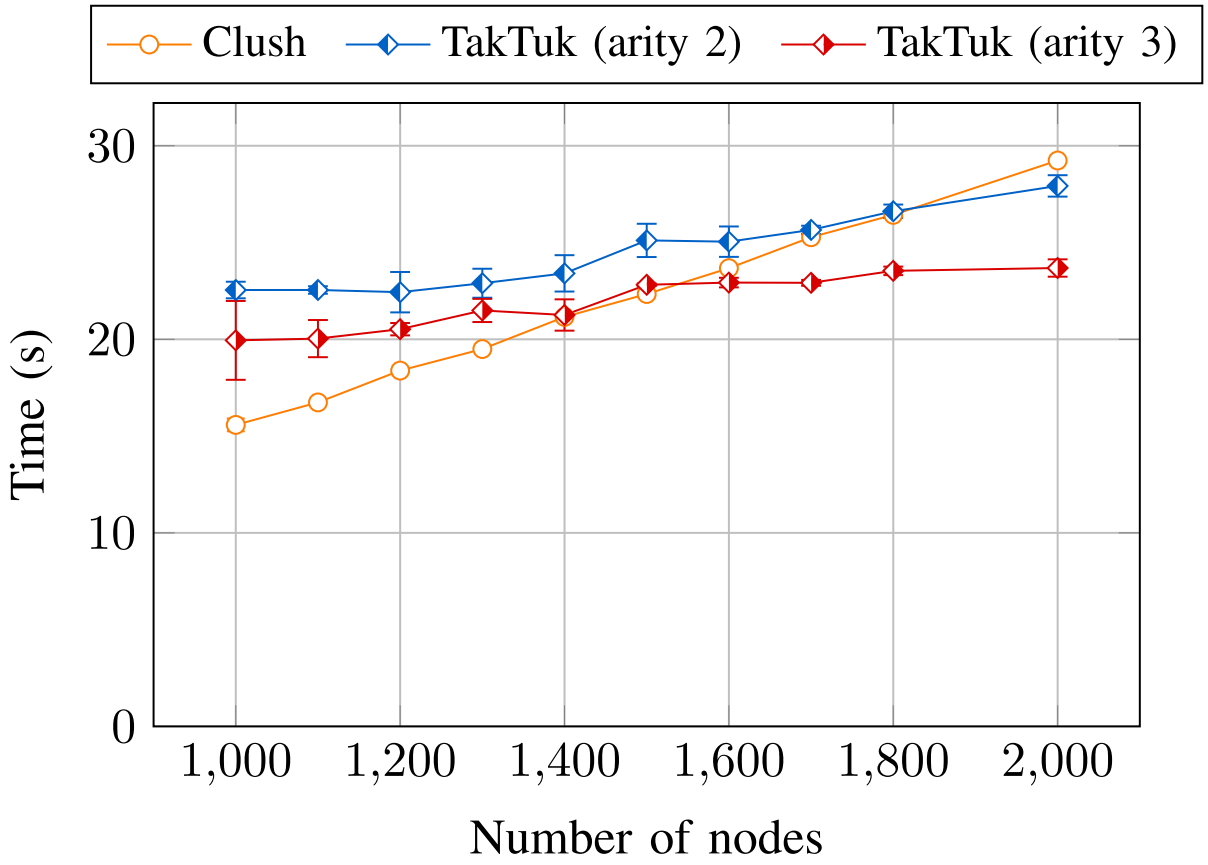
Experiment execution

To run our experiment we used XPFlow:

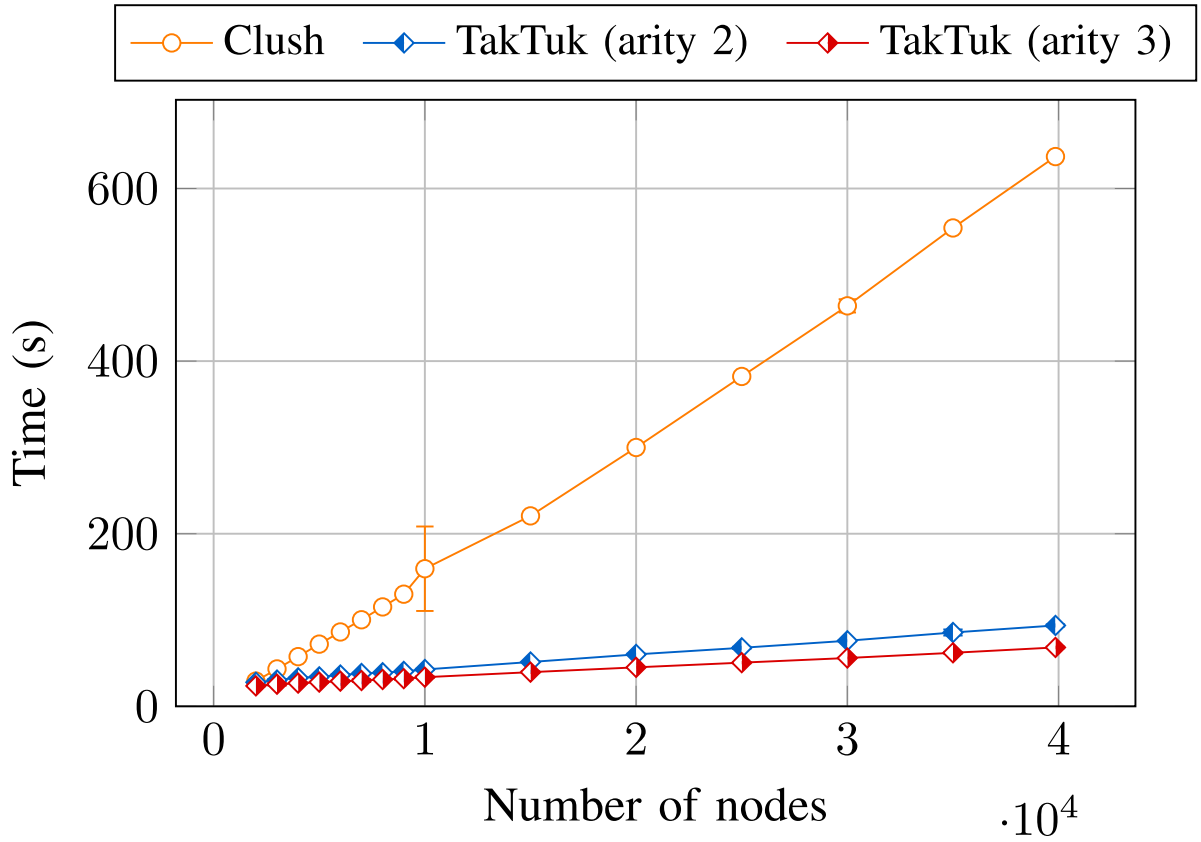
- experiment control engine,
- models experiments with business workflows (from BPM)
- has robust failure model
- allows as to recover from failures (e.g., bad IB interface)

See `http://xpflow.gforge.inria.fr` for more information.

Results (between 1,000 and 2,000)



Results (up to 40,000)



Conclusions

We were able to run a command distributed on nearly 40,000 nodes:

- Distem used to emulate the platform
- scalable network thanks to VXLAN over InfiniBand
- XPFlow for the experiment execution

Future work will include:

- even bigger scale of experiments
- visualization

Thank you for your attention. Questions?

Contact: tomasz.buchert@inria.fr

Results - curve fitting

By mapping the obtained results to a model:

$$T(n) = A \cdot n + B \cdot \log(n) + C$$

we found that:

$$\begin{aligned}
\text{Clush} : T_C(n) &= 0.01649 \cdot n - 7.55 \cdot \log(n) + 52, \\
\text{TakTuk 2} : T_2(n) &= 0.00146 \cdot n + 3.55 \cdot \log(n) - 4, \\
\text{TakTuk 3} : T_3(n) &= 0.00099 \cdot n + 2.27 \cdot \log(n) + 4.
\end{aligned}$$