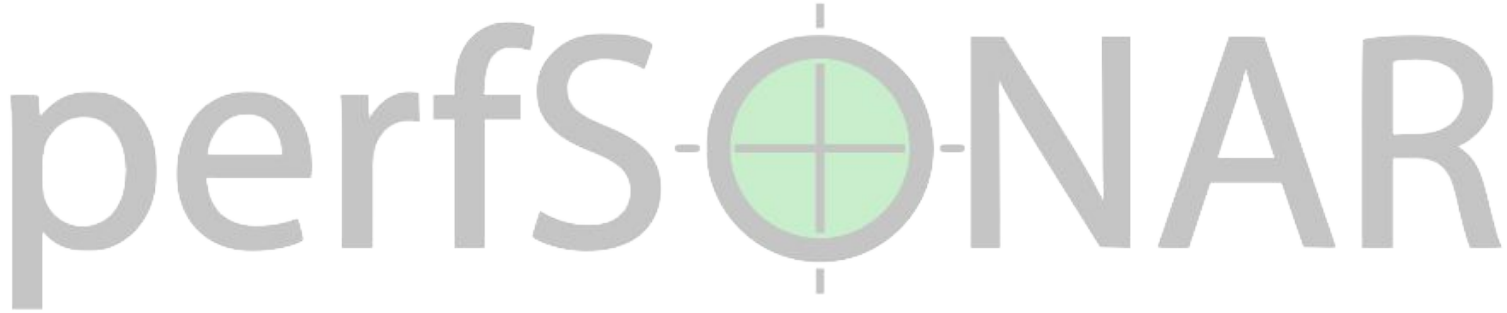


pScheduler extensibility and applicability for the GÉANT Community

5th SIG PMV Meeting, Manchester

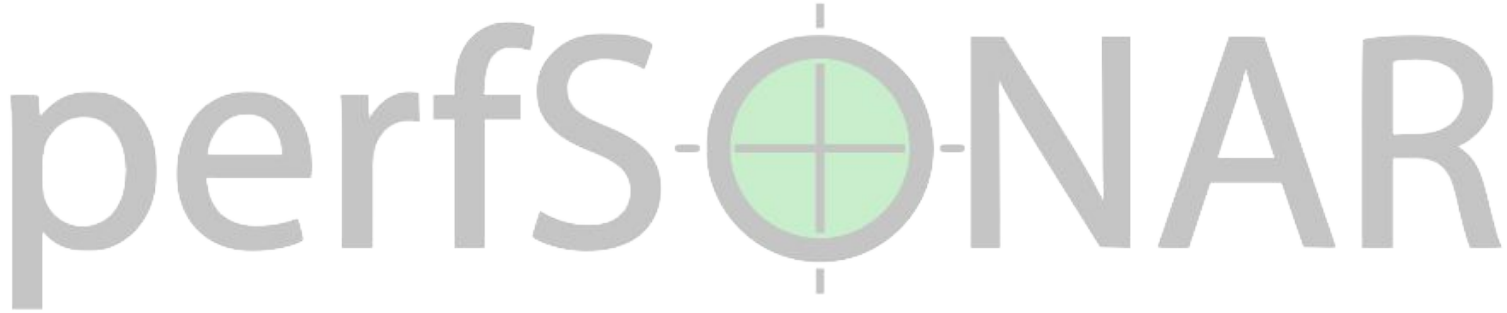
Antoine Delvaux — antoine.delvaux@man.poznan.pl

GÉANT perfSONAR Service Manager, perfSONAR developer

October 23th, 2018

Outline

- perfSONAR
- pScheduler
- pScheduler extensibility in practice
- pSConfig
- Applicability for the GÉANT Community



perfSONAR

Performance focused **S**ervice **O**riented **N**etwork monitoring **AR**chitecture

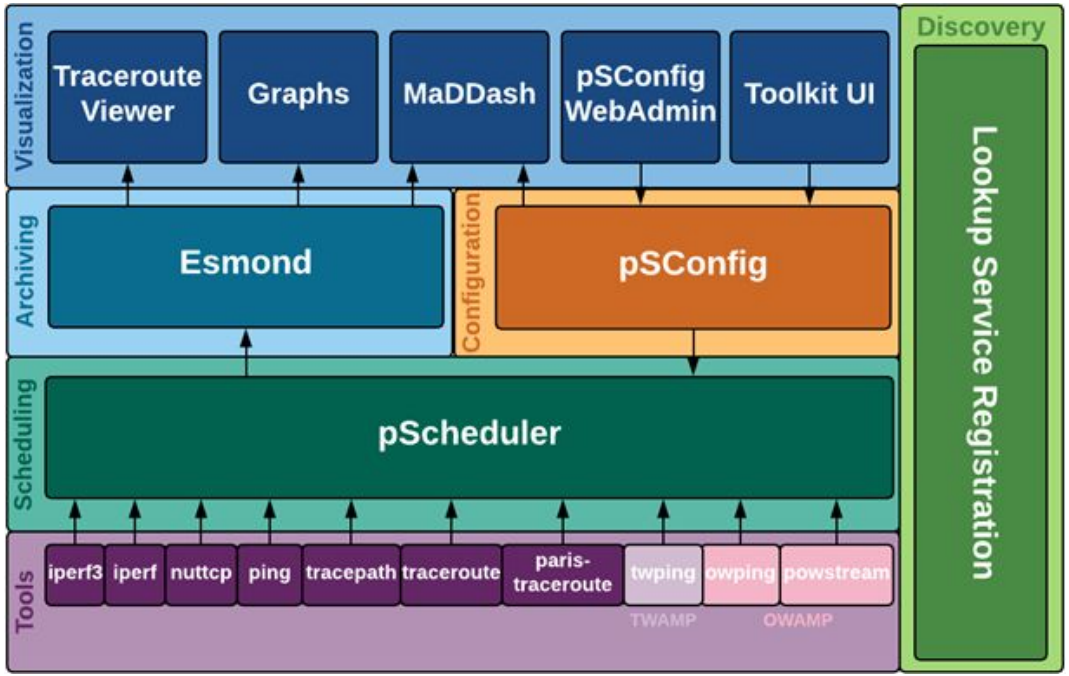
ESnet ENERGY SCIENCES NETWORK GÉANT INDIANA UNIVERSITY INTERNET2 UNIVERSITY OF MICHIGAN

perfSONAR

- Software product to monitor networks for performance discrepancies, on an end-to-end basis (multi-domain), ensuring friction-free use for scientific applications
  - measurement tools, archives, visualisation, GUI
- Developed for the R&E community, by the R&E community
- Running on 3 major linux distros
  - CentOS 7
  - Debian 8 and 9
  - Ubuntu 14, 16 and 18
- Open Source (Apache License): https://github.com/perfsonar/

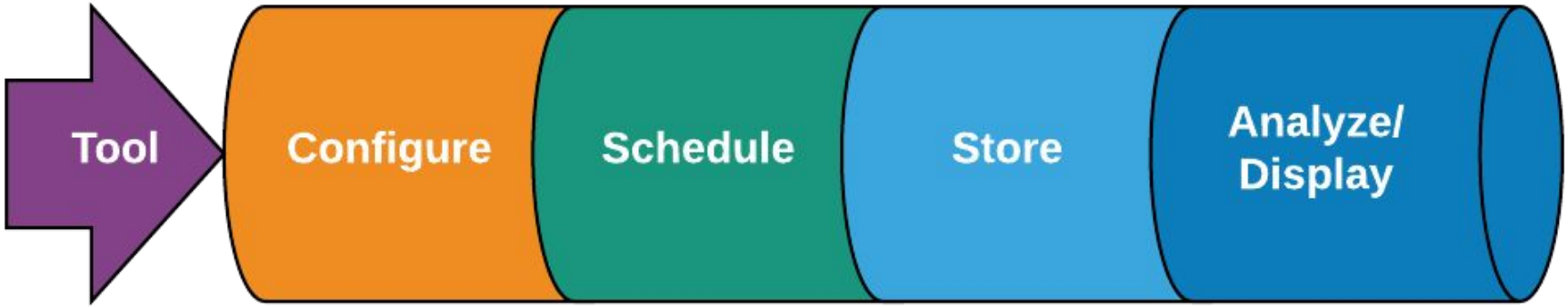
ESnet ENERGY SCIENCES NETWORK GÉANT INDIANA UNIVERSITY INTERNET2 UNIVERSITY OF MICHIGAN

perfSONAR Architecture



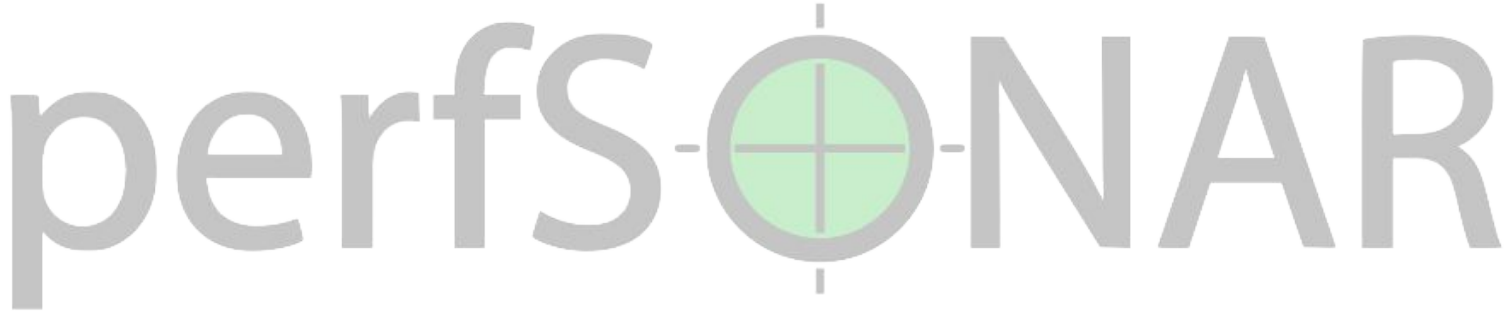
ESnet ENERGY SCIENCES NETWORK GÉANT INDIANA UNIVERSITY INTERNET2 UNIVERSITY OF MICHIGAN

Building the perfSONAR Pipeline



Why do we care about adding new capabilities?

- Evolving network environments
  - Cloud, containerization and virtualization
  - Embedded network devices
- New Types of Measurements
  - Research platforms really interested in things like disk-to-disk transfer
- Data Analytics
  - We want perfSONAR data to be used in new and interesting ways
  - This includes integration of data in projects looking at more than just perfSONAR results
- Meshes Continue to Grow
  - Can we build smarter meshes than we do today to avoid over-testing as collaboration get larger



pScheduler

the perfSONAR Scheduler

ESnet ENERGY SCIENCES NETWORK GÉANT INDIANA UNIVERSITY INTERNET2 UNIVERSITY OF MICHIGAN

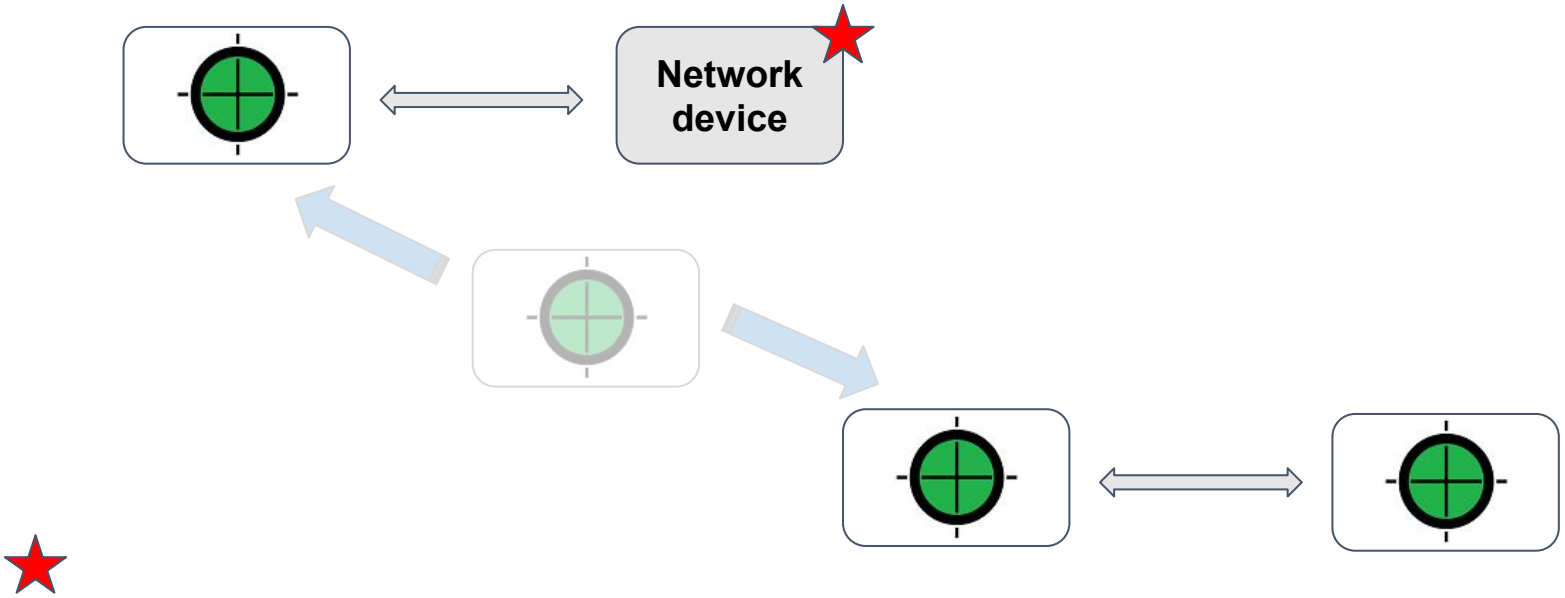
pScheduler: the perfSONAR Scheduler

- Software for scheduling, supervising and archiving measurements
  - Central to the perfSONAR architecture
- Plug-in architecture, allows integration of new
  - Tests — *Things to measure*
  - Tools — *Things to do the measurements*
  - Archivers — *Ways to dispose of results*
  - Contexts — *Environments in which to run the measurements*
- Aim is easily bring new applications to perfSONAR
- Core development team doesn't need to be involved other than in an advisory role

pScheduler usage

- From the CLI
  - `pscheduler command [ arg … ]`
  - `pscheduler task [ task-opts ] testname [ test-opts ]`
  - `pscheduler task --tool toolname testname [ test-opts ]`
  - `pscheduler --help`
- Integrated with the perfSONAR toolkit GUI
  - Add measurements/tests to the schedule
- Integrated with pSConfig
  - Manage the measurement schedule of multiple perfSONAR nodes

Measurement scenarios



Depending on test type and on network device support

Tasks classification

- Exclusive
  - Cannot run at the same time of another exclusive or normal test
  - Ex: throughput
- Normal
  - Multiple normal tests can run at the same time
  - Ex: latency
- Background (and background-multi)
  - Are running at the same time of all other
  - Ex: rtt, trace, clock, latencybg

Plot task classification and schedule

- To check current schedule
  - `pscheduler schedule`
  - `pscheduler plot-schedule > schedule.png`
- More info:
  - http://docs.perfsonar.net/pscheduler_ref_tests_tools.html#pscheduler-ref-tests-tools-test-classifications

Archiving results

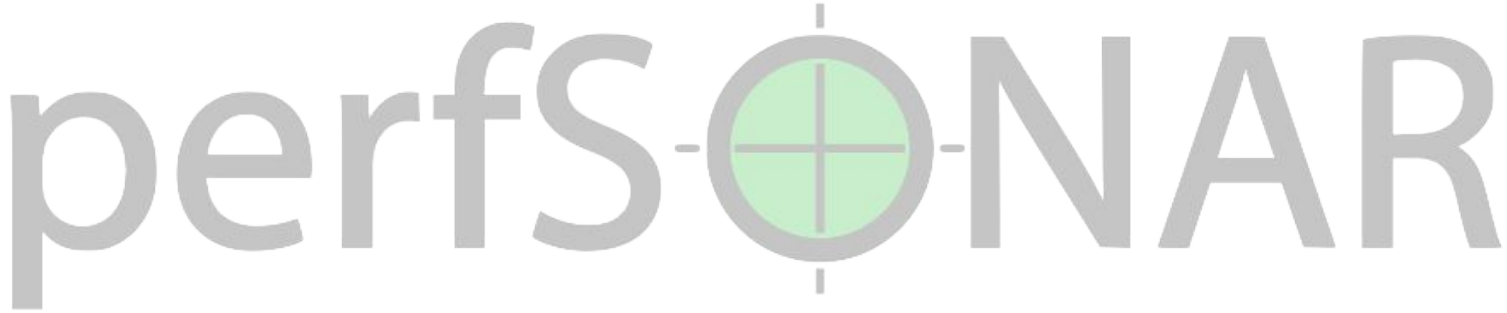
- Different archivers are existing:
  - Esmond
  - Syslog
  - HTTP
  - RabbitMQ
- Can easily add others, probably the simplest plugins are archivers
- A single perfSONAR node can use multiple archivers
  - depending on measurement type
  - storing measurement data in different places

Task context

- pScheduler can setup a context before running a task
- Contexts are another type of plugin
- Currently, the only context plugin existing is Linux NetName Spaces (LNNS) switching
  - pScheduler switches LNNS before running the task
  - Can involve other virtual interfaces, different routing
- Do you need any context setup before running a task?

Other interesting bits about pScheduler

- Limits system to control resources usage and who can access your measurement point
- API provides access to different statistics
- Troubleshooting and diagnosis tools included
- Written in Python 2, will be ported to Python 3 next year
- Future features
  - Priorities, tasks preemption
  - Resources pooling (mainly ports)



Extensibility in practice

How can we leverage pScheduler extensibility?

ESnet ENERGY SCIENCES NETWORK GÉANT INDIANA UNIVERSITY INTERNET2 UNIVERSITY OF MICHIGAN

pScheduler API

- Well documented REST API:
  - https://github.com/perfsonar/pscheduler/wiki/REST-API
- Standardized, documented data formats using JavaScript Object Notation (JSON)
- Your tool can easily call pScheduler

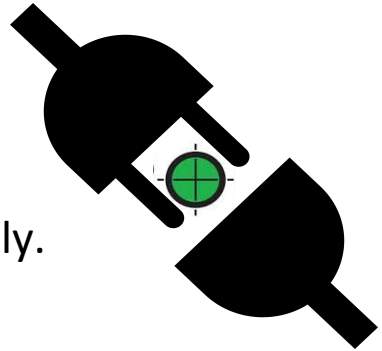
JQ scripting

- Lightweight and flexible command-line JSON processor
- Can manipulate any JSON document
- **sed** for JSON data
- JQ scripts can be used in different places in perfSONAR:
  - Archivers can transforms the result before archiving it
  - pScheduler limits can be applied depending on JQ script result
  - Tasks can be rewritten when received and before being evaluated
  - pSConfig templates can use JQ scripts, for example to set an Esmond authentication key

pScheduler packaging

- pScheduler is designed to be standalone
- Test, tool and archiver plugins are individually-installable packages
- Can add plugins to systems that need them
  - And 2 meta packages to contain them all for ease of installation
- Removing a plugin package renders pScheduler unaware that it exists

pScheduler SDK



Goal

Make perfSONAR plugin development efficient and developer friendly.

How to achieve that?

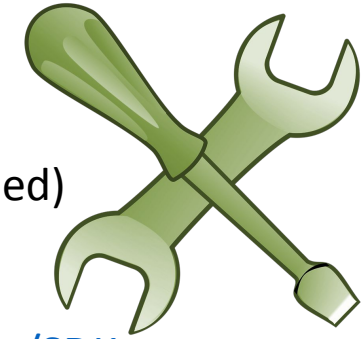
By scripting search and new plugin creation steps and distributing the scripts as a part of perfSONAR source.

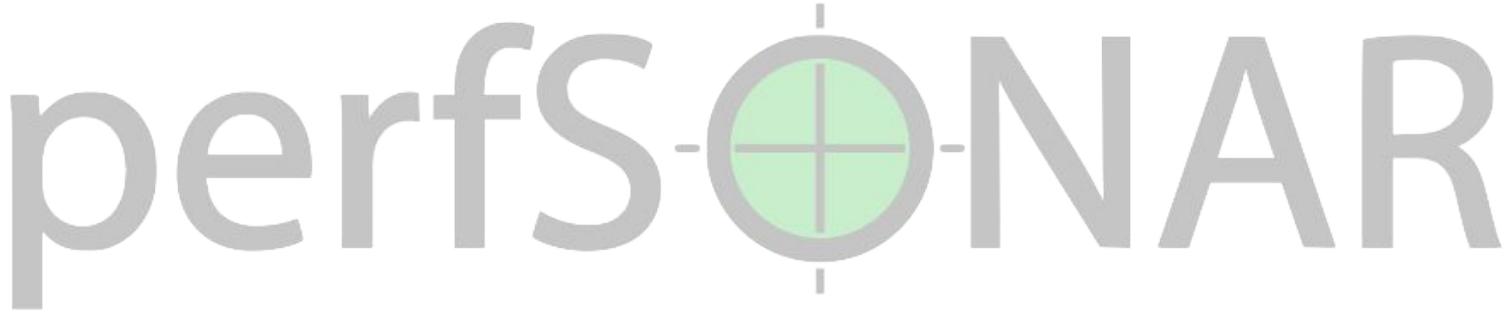
Benefits

Quick start: ability to create perfSONAR plugins without extensive knowledge of perfSONAR.

SDK usage

- Clone and build pscheduler
  - `pscheduler/scripts/SDK/plugin_dev [test|archiver] [name]`
  - `pscheduler/scripts/SDK/plugin_dev [tool] [test] [name]`
- Write your plugin: edit those files:
  - `'can-run'` and `'enumerate'` (see `'EDIT ME'` tags)
- Test written plugin:
  - `sudo make cbic`
- In the test and tool directories look for additional changes (if needed)
  - `grep -R Order *`
- More details at
  - https://github.com/perfsonar/pscheduler/tree/master/scripts/SDK





pSConfig

Organising measurement schedules amongst a set of hosts

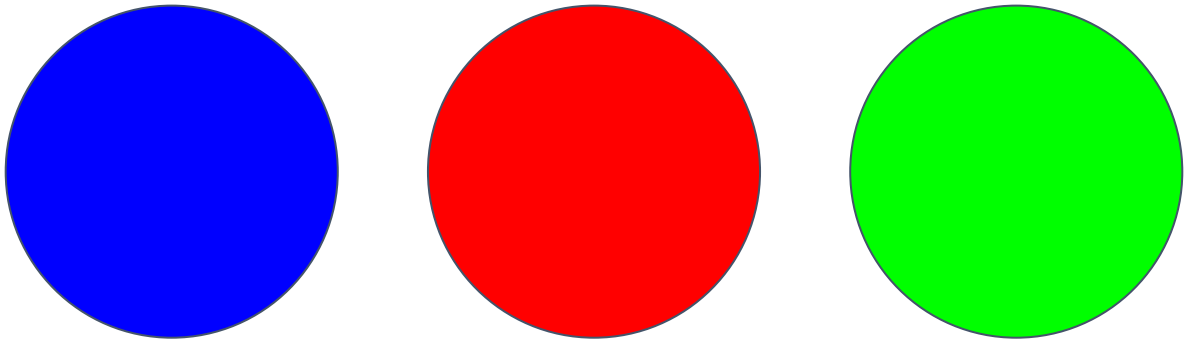
ESnet ENERGY SCIENCES NETWORK GÉANT INDIANA UNIVERSITY INTERNET2 UNIVERSITY OF MICHIGAN

pSConfig

- A **template** framework for describing and configuring a **topology** of pScheduler **tasks**
- pScheduler **tasks**, with all their options and a schedule
- A **topology**: the way in which tasks are interrelated and arranged
- A template: description of the task topology
  - JSON: *pSConfig templates*
  - address: collection of properties, unit of input to a task
  - group: combining addresses together
  - variables

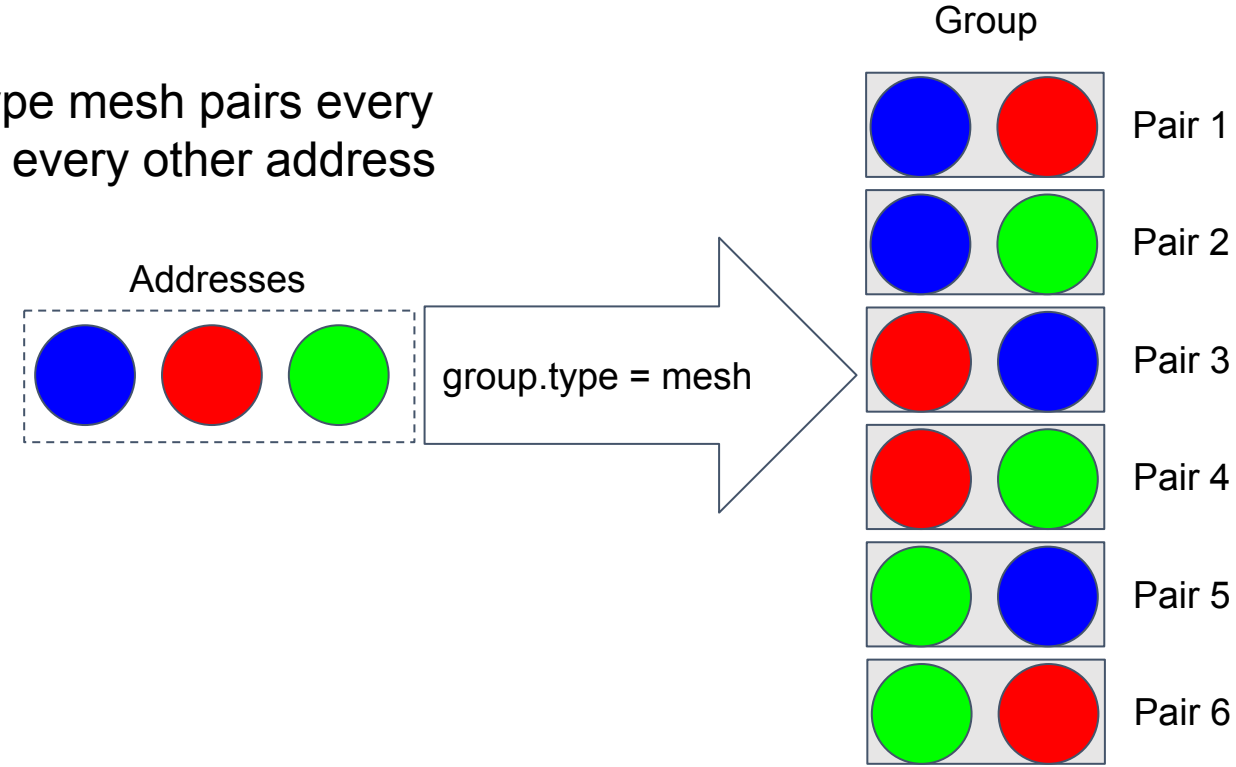
Addresses

An address is an object with properties like the three circles below:



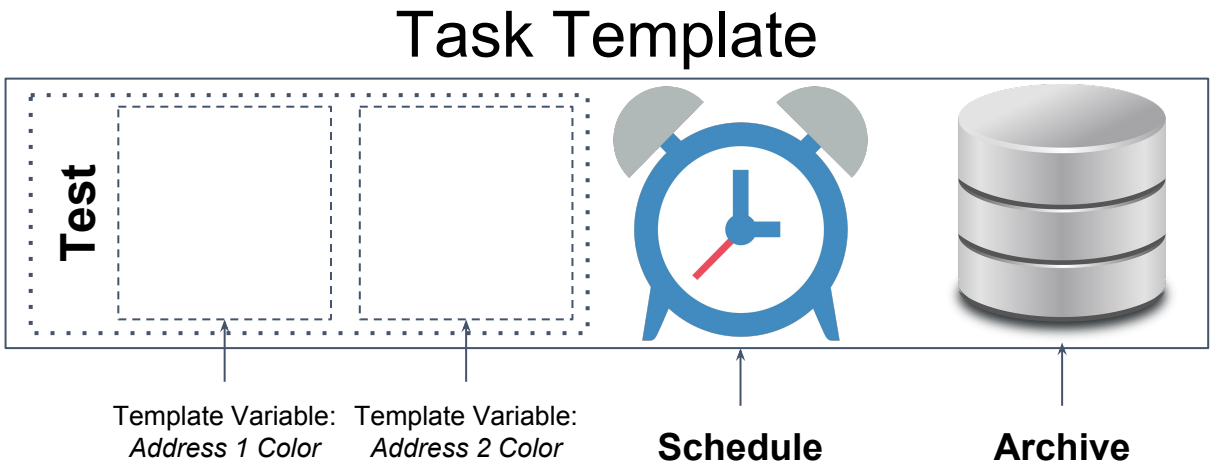
Groups

A group of type mesh pairs every address with every other address in the group



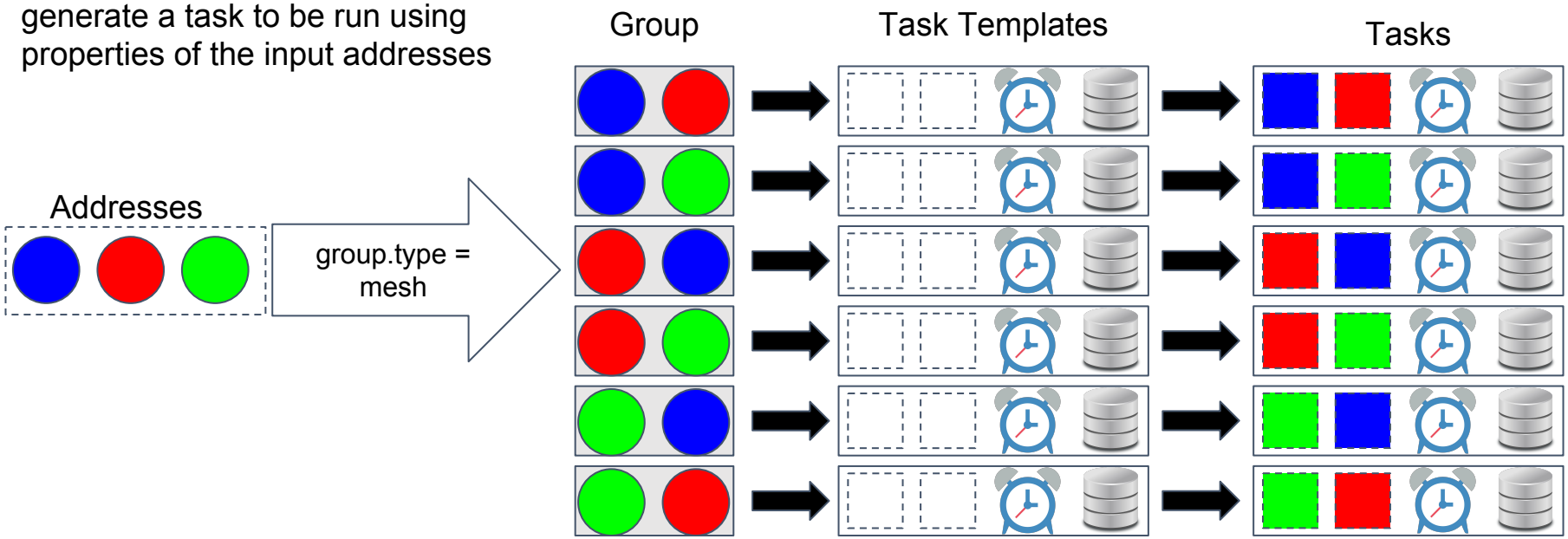
Tasks and Template Variables

We define a template for building tasks which include variables used in the test based on the color of each address object pair input by the group. It can also have things like a schedule of when to run and an archive that are the same across all tasks in this example (we could use variables in those too if we wanted).



Putting it all together: Creating Tasks

For each pair in the group, we generate a task to be run using properties of the input addresses



Additional remarks on pSConfig

- perfSONAR hosts consume pSConfig templates
  - The host creates the measurements that relates to its interfaces
- A single perfSONAR host can use multiple pSConfig templates
- Local measurements and archivers can exist at the same time
- Adding and removing pSConfig templates to existing hosts can easily be done through Ansible (perfSONAR roles)

Applicability for the GÉANT Community

Can pScheduler and pSConfig be helpful to existing and future GÉANT projects?

Possible applications for GÉANT

- NetMon
  - Scheduling?
    - Call pScheduler to setup tasks
    - Organise all tasks and hosts through pSConfig
  - And integrate OWAMP patches
    - OWAMP (and TWAMP) are maintained by the perfSONAR dev team
- MD-VPN monitoring
  - Multiple interfaces and support for Linux NetName Spaces is helping
  - But can also be done without it, depending on the VPN topology and monitoring setup

And more possible applications for GÉANT

- eduroam monitoring?
- WiFi monitoring?
- Archiving to some other backend
  - Prometheus
  - ElasticSearch

And then … visualisation

- Once archived, measurement visualisation can easily be provided by other frontends
- Kibana (ELK stack)
- Grafana
  - NetSage: https://portal.netsage.global/

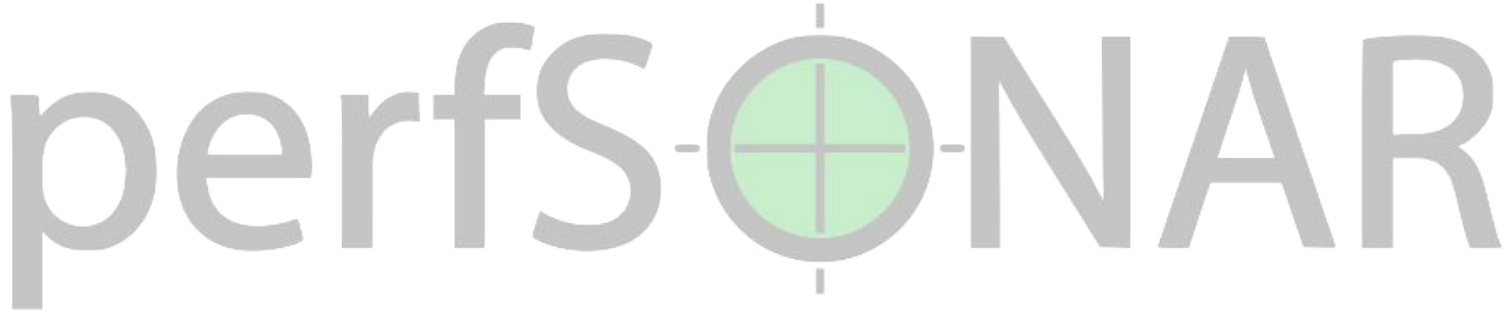
    Contains link utilisation, flow and perfSONAR data
  - Tight integration with Prometheus

Summary

- perfSONAR can be seen as a measurement framework
- pScheduler and pSConfig provides many different ways to incorporate new measurement types, new metrics
  - could even be used for outsid of the performance monitoring world
- There are a lot of extension points and a complete and flexible API
- These benefits can be leveraged to easily build new applications
  - quick prototyping, proof of concept
  - production level application, perfSONAR being a production level tool

Resources

- perfSONAR technical documentation:
  - https://docs.perfsonar.net/
- perfSONAR (and pScheduler) source code:
  - https://github.com/perfsonar/
- pScheduler API:
  - https://github.com/perfsonar/pscheduler/wiki/REST-API
- perfSONAR/pScheduler SDK:
  - https://github.com/perfsonar/pscheduler/tree/master/scripts/SDK
- Users mailing list (pS dev are participating)
  - http://lists.internet2.edu/sympa/info/perfsonar-user



pScheduler extensibility and applicability for the GÉANT Community

5th SIG PMV Meeting, Manchester

Antoine Delvaux — antoine.delvaux@man.poznan.pl

October 23th, 2018

Thanks for content to the whole perfSONAR development team