

perfs--NAR

Introduction to MaDDash

Andy Lake, ESnet

Training Workshop for High-Speed Networks

July 23, 2019



Overview

- Motivation
- What is MaDDash?
- Configuration Strategy
- Advanced Topic: Notifications

So you've got
these fluffy new
perfSONAR
nodes...





...but there is
such a thing as
too much
fluffiness

The Need

- Users need a clear way to visualize a large number of tests, whether it is within an organization or across multiple institutions
- The tests perfSONAR supports primarily have two endpoints
- Tests can often be grouped together by purpose (e.g. test connectivity between all these hosts dedicated to project X)

How do we capture the known relationships between tests and leverage this knowledge to find patterns representing network performance issues?

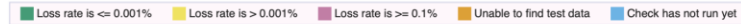
Existing solutions?

- Lots of good “host” monitoring solutions exist, all of which are inherently one dimensional

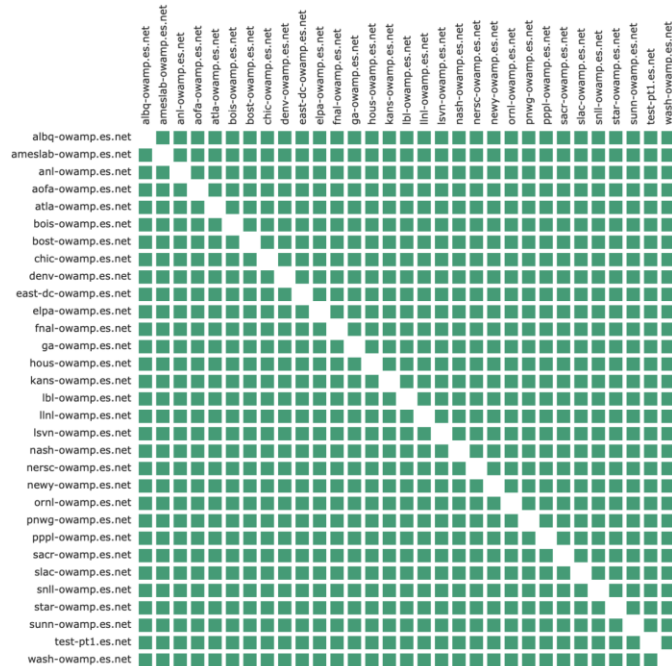


Enter MaDDash

ESnet - ESnet to ESnet Packet Loss Testing - Loss



✓ No problems found in grid

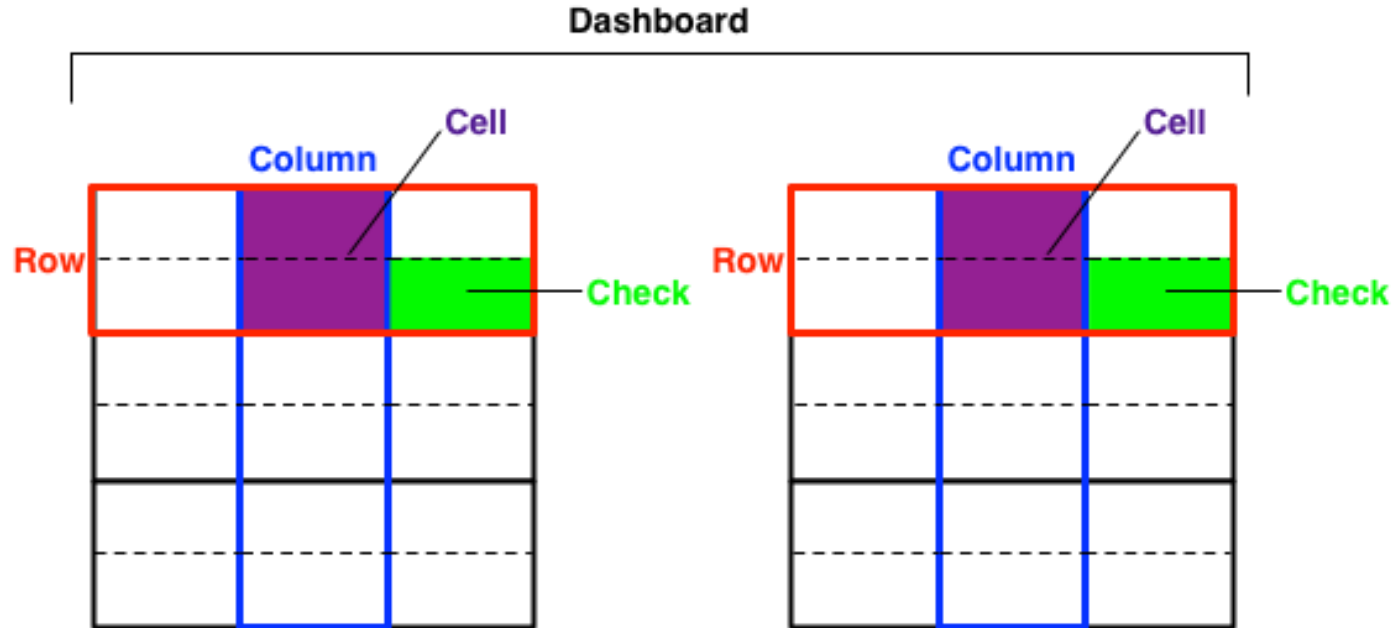


- Builds 2-dimensional grids of measurements
- Rows and columns represent endpoints of tests - though generic enough they could be other things
- Each cell is colored based on result compared to a threshold: e.g. **OK**, **Warning**, **Critical**
- Highly customizable
 - Definable thresholds
 - Custom rules to match common patterns
 - Customizable colors

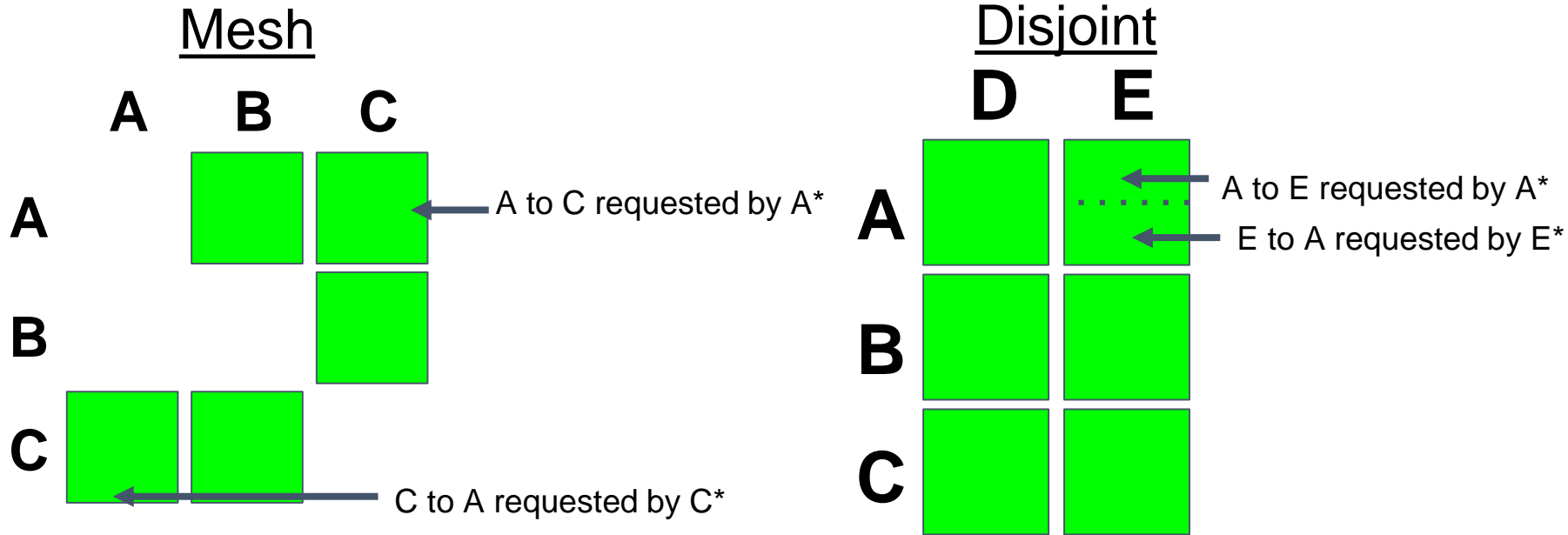
How MaDDash works

- **MaDDash does not execute any network measurements.**
- MaDDash periodically queries archives for measurements that have already run.
- Two main components:
 - maddash-server
 - Java process that aggregates results on schedule, applies thresholds and looks for patterns
 - Dashboards defined in YAML configuration file (which most people never touch..more later)
 - Capable of sending various types of notifications
 - REST API for looking at summarized results
 - maddash-webui
 - Web frontend that queries maddash-server REST API and displays results

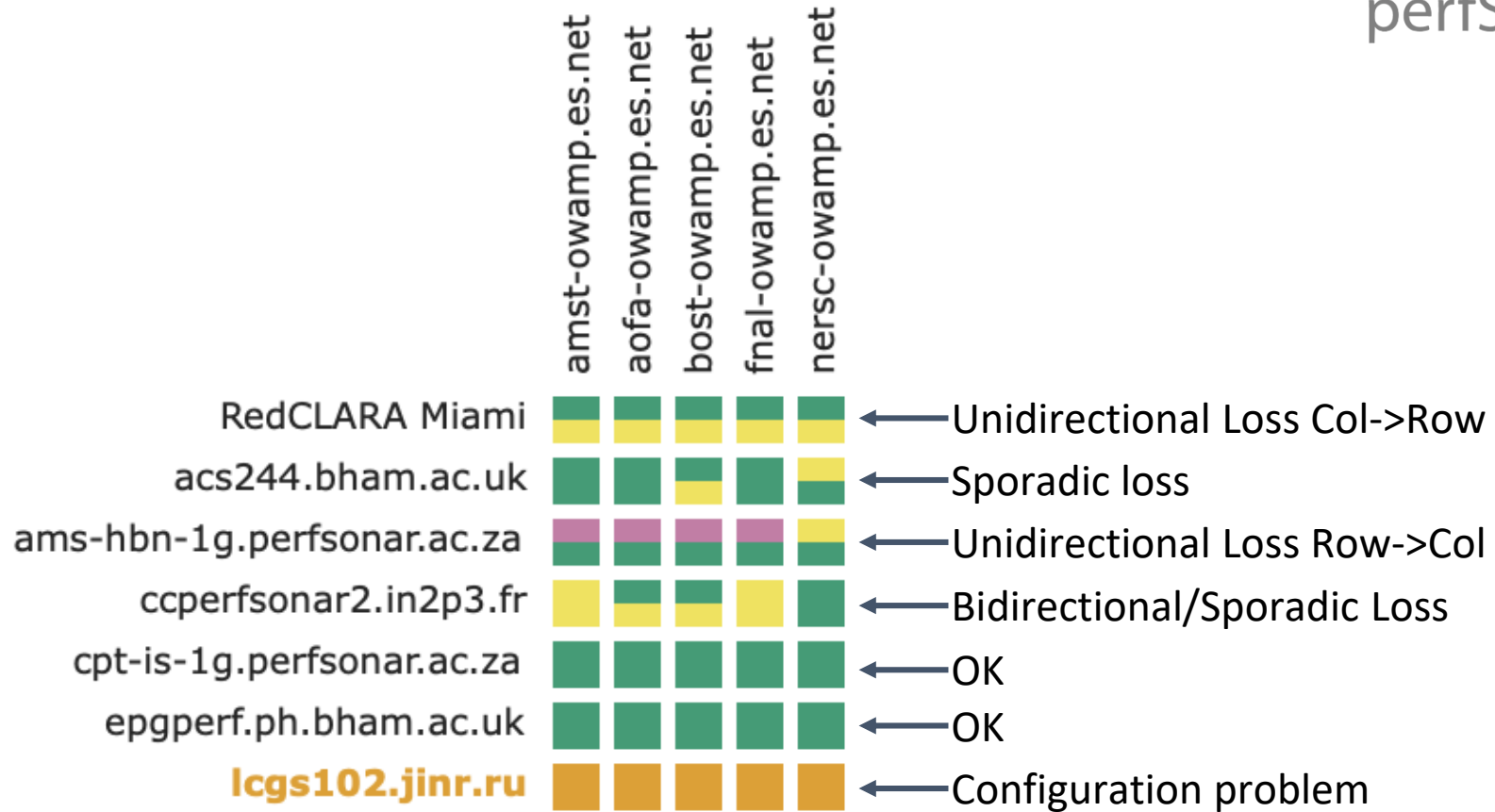
Anatomy of a Dashboard: Basics



Anatomy of a Dashboard: Grids



**If requester is marked "no-agent" then other side requests test*

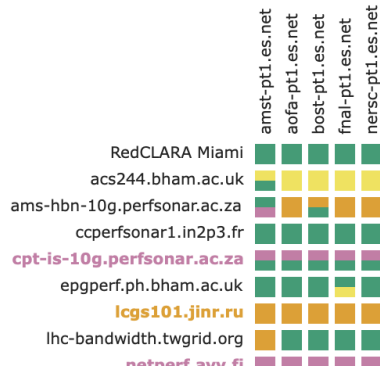
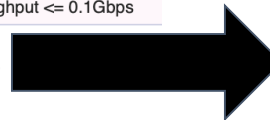


Anatomy of a Dashboard: Reports

ESnet - ESnet to International Throughput Testing - Throu



! Found a total of 13 problems involving 13 hosts in the grid



ESnet - ESnet to International Throughput Testing - Throughput

perfsonar-test1.kek.jp

✖ Unable to find any throughput measurements

Category: CONFIGURATION

Potential Solutions:

- Verify the host is up
- Verify that httpd is running on the host ('systemctl status httpd')
- Verify that cassandra is running on the host ('systemctl status cassandra')
- Verify that the pSConfig pScheduler Agent is running ('systemctl status psconfig-pscheduler-agent')
- Look for any pscheduler errors with the command 'pscheduler troubleshoot'
- Verify the host is pointing at the correct pSConfig template with the command 'psconfig remote list'
- Verify when the host last updated its measurement configuration with 'psconfig pscheduler-stats'
- Verify the host allows access to TCP port 443 and 5201 required for the measurement to run
- Look for errors in /var/log/perfsonar/psconfig-agent.log
- Look for errors in /var/log/pscheduler/pscheduler.log

netperf.ayy.fi

✖ Throughput is below warning or critical thresholds to a majority of sites

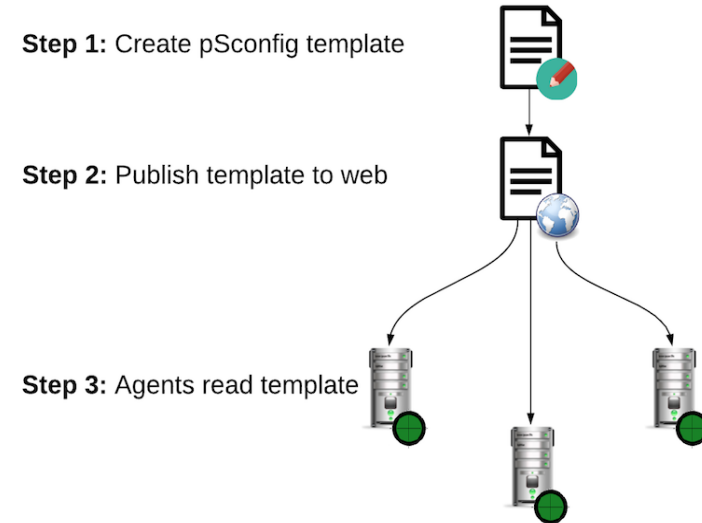
Category: PERFORMANCE

sampaps02.if.usp.br

✖ Throughput is below warning or critical thresholds to a majority of sites

Building Dashboards with pSConfig

- **pSConfig** is a framework for defining the measurements you want run among one or more hosts
- A single source of truth for both measurement hosts and your dashboards
- Currently two agents:
 - **pscheduler-agent**: It reads the template file(s) and generates pScheduler tasks
 - **maddash-agent**: It reads the the template file(s) and generates a maddash.yaml file



MaDDash/pSConfig Quickstart

1. Define pSConfig JSON file (by hand, using pSConfig Web Admin, etc)
2. Point at the JSON file by running the following on each measurement host:

```
# psconfig remote add --configure-archives "https://10.0.0.1/example.json"
```

3. Point at the JSON file by running the following on MaDDash host:

```
# psconfig remote add "https://10.0.0.1/example.json"
```

MaDDash/pSConfig Quickstart:

<https://docs.google.com/document/d/1k7FT66MKPy3JjpD5k00FAFITpSdFmZ6huhTUDQ2rGGY/edit>

Advanced: MaDDash Notifications

- MaDDash can send reports from previous slides to generate notifications
- Supported notification types:
 - Email
 - ServiceNow (Coming in v4.2 - currently in beta)
- Requires editing maddash.yaml file directly (commented example in default file)

notifications:

-

name: "My Email Report"

type: "email"

schedule: "0 * * * ?"

problemReportFrequency: 86400

minimumSeverity: 1

parameters:

dashboardUrl:

"http://dashboard.domain.example"

mailServer:

address: "127.0.0.1"

port: 25

from: "dashboard@domain.example"

to:

- "email1@domain.example"

- "email2@domain.example"

Closing remarks

- MaDDash's goal is to make it easier to analyze the results of a large number of point-to-point measurements
- It is integrated with other perfSONAR tools like pSConfig to make keeping the dashboards and actual measurements aligned while trying to minimize the steps required to setup
- MaDDash can highlight a lot of issues, but it is just the messenger - still significant work to solve those issue
- MaDDash is another tool in the toolbox, and can hopefully help identify issues in conjunction with other measurement/monitoring information

Further Information

Quickstart Guide:

<https://docs.google.com/document/d/1k7FT66MKPy3JjpD5k0OFAFITpSdFmZ6huhTUDQ2rGGY/edit>

Official Docs: <http://docs.perfsonar.net/#displaying-measurements-with-maddash>

ESnet Dashboard: <http://ps-dashboard.es.net>

Support List: perfsonar-user@perfsonar.net

