# TimeMap Update

**Fabio Farina (GARR)**
**And GN5-1 WP6 T3 team**
TimeMap-dev@lists.geant.org

# Outline

- Why TimeMap
- Current status
- Recent developments on
  - Data acquisition
  - Anomaly detection

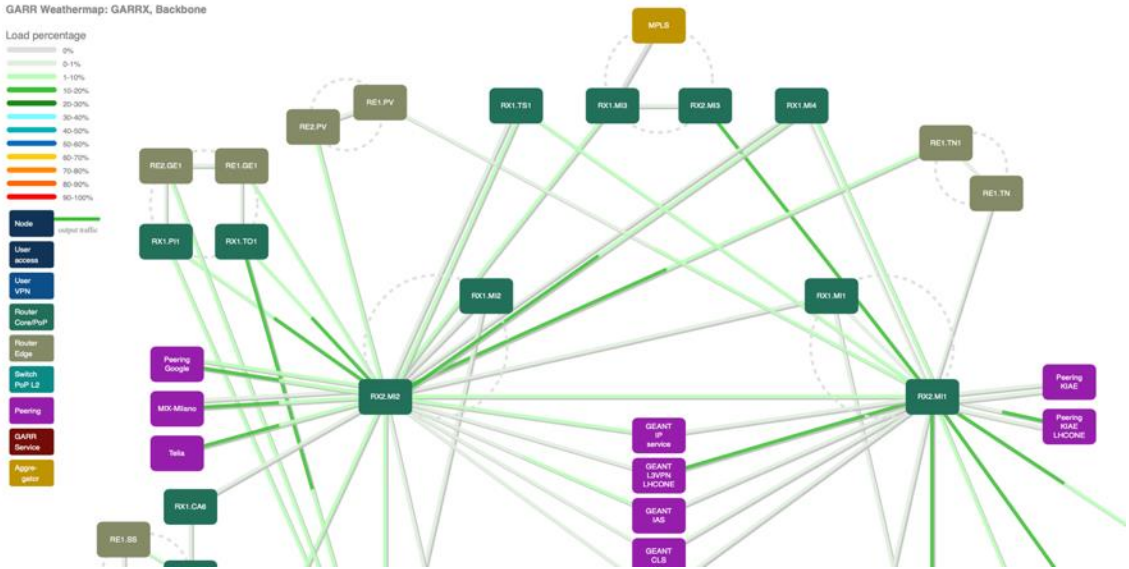
# Why TimeMap: the road-trip analogy

**How is the road ahead today?**
**And how is it is in average?**

**it may be nice …**


**Or lots of Stop & Go**

# Network Traffic: what do we usually have?

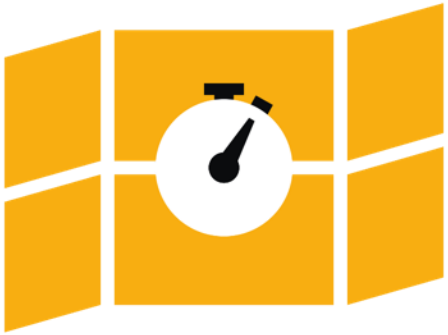



**But this is OK for bulk data transfers**

# NOT for real time applications which are sensitive to Latency & Jitter!

## We need to monitor ”the hidden”:

- latency
- jitter



TimeMap

## We need to keep track of ”the hidden”:

- historic series

## We need to find anomalies in “the hidden”

- machine learning
- alarms
- call the right NOC for the right network segment



GÉANT

# TimeMap instance for the GÈANT backbone

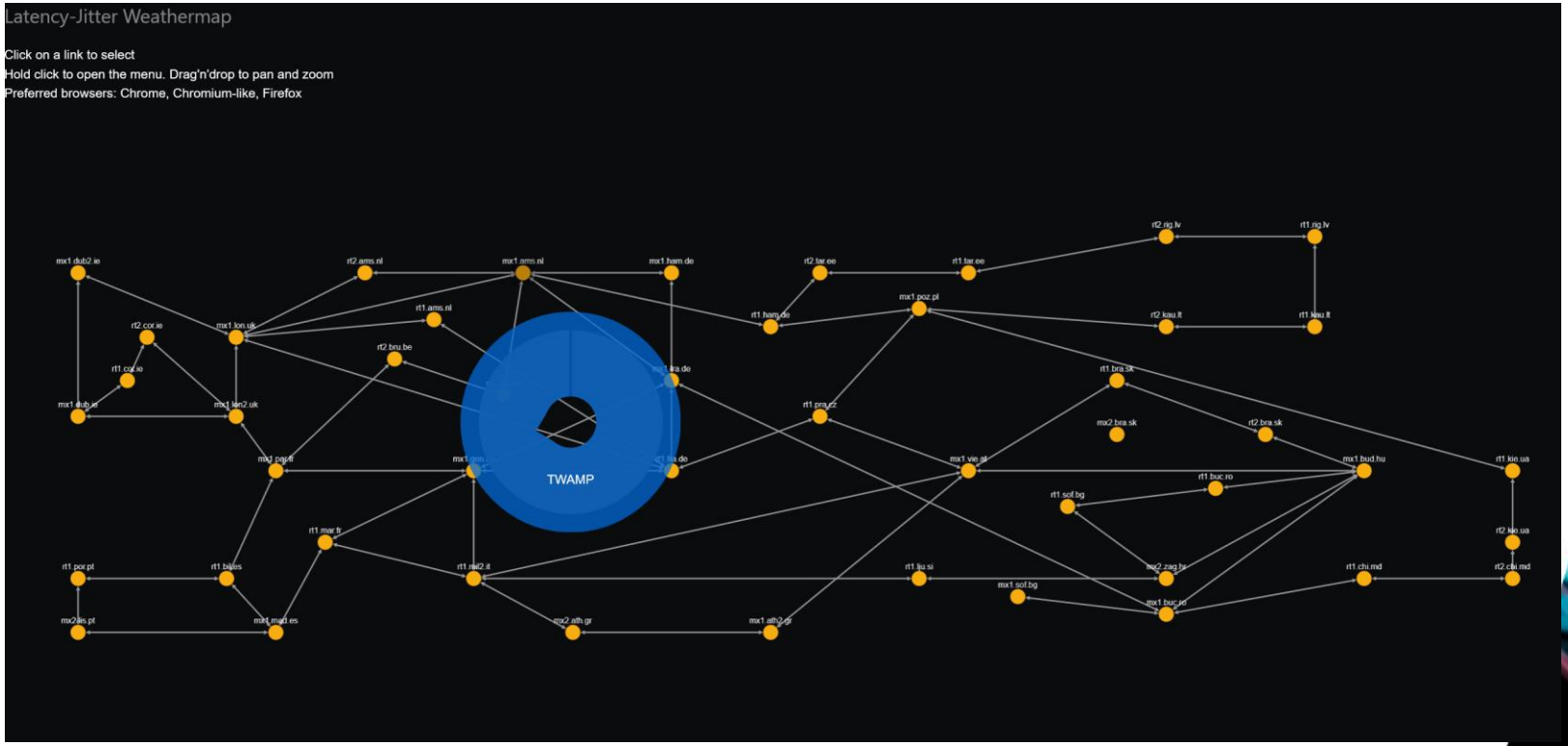
- The service on GEANT backbone

**https://timemap.geant.org/**

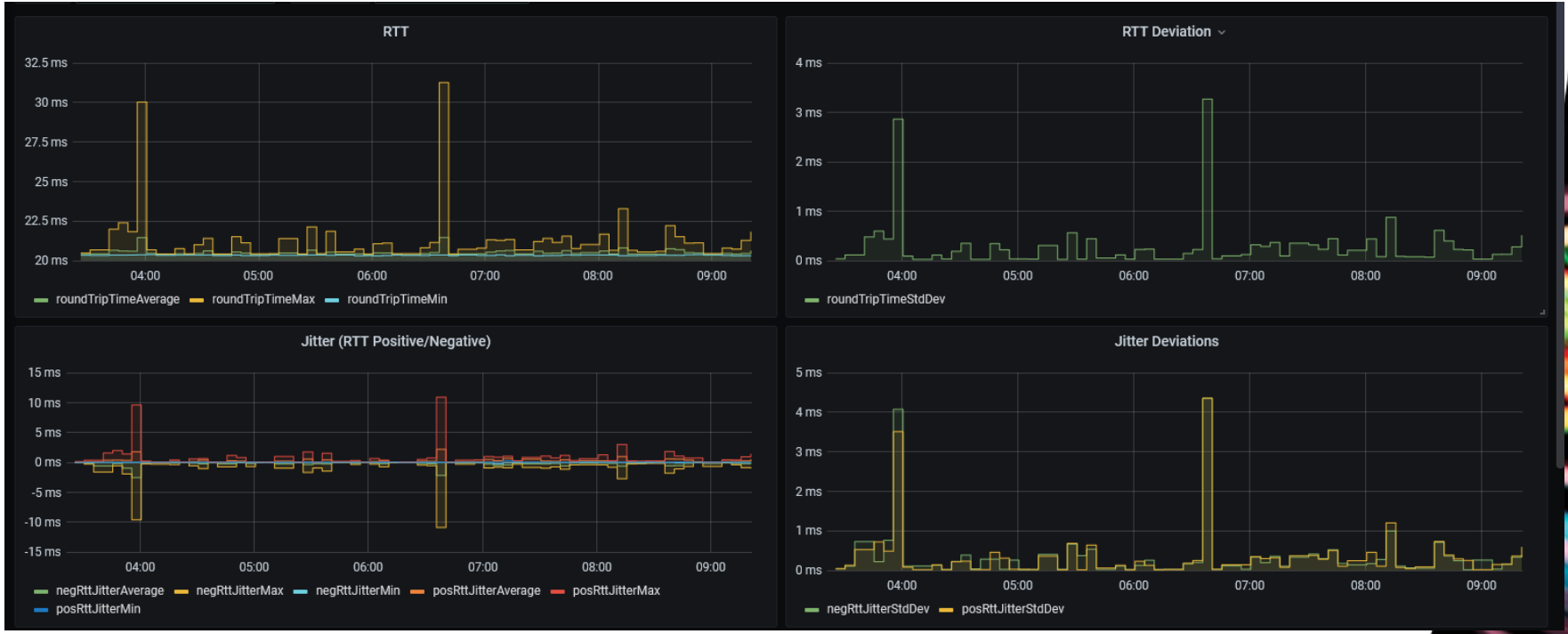
- Documentation: source code, user and admin guides, customization

**https://gitlab.geant.org/gn4-3-wp6-t1-lola/timemap_public**

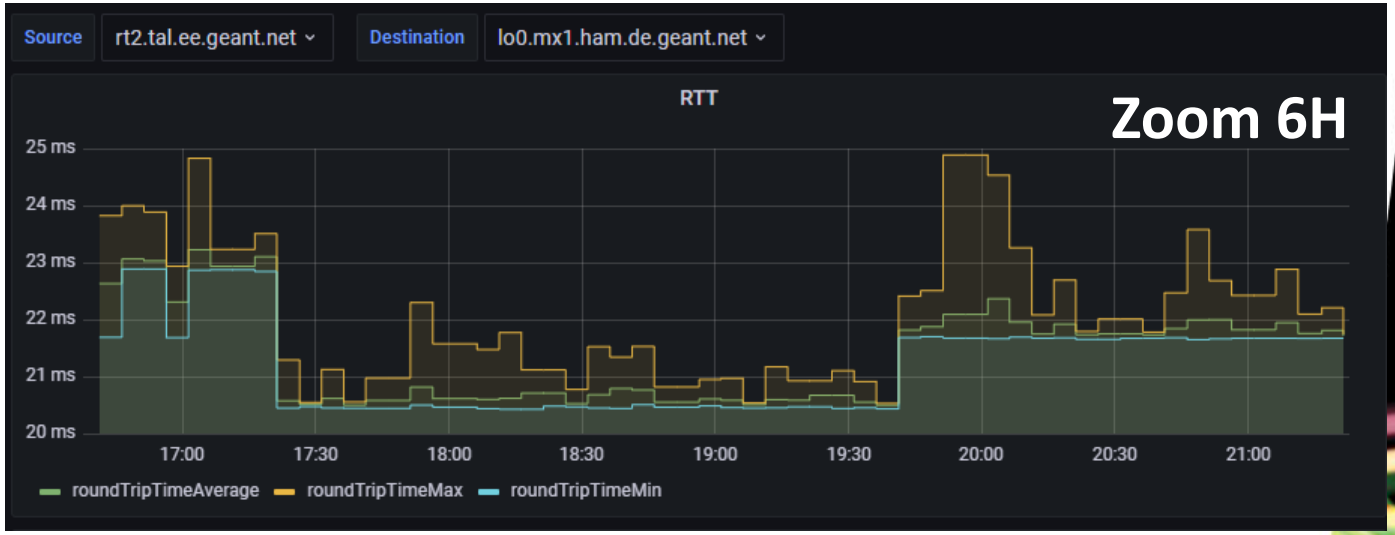
# The entry map page: click on link



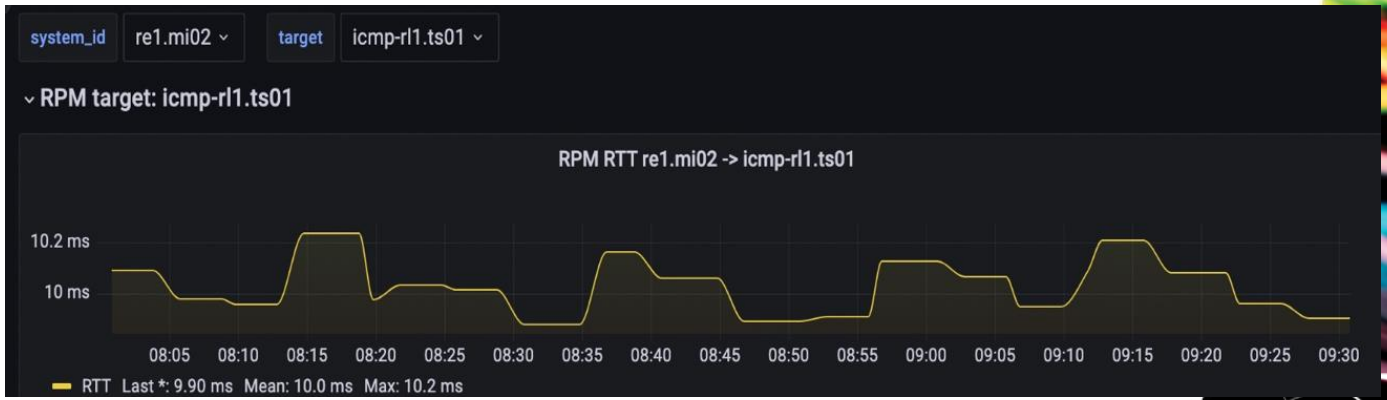
# Observations



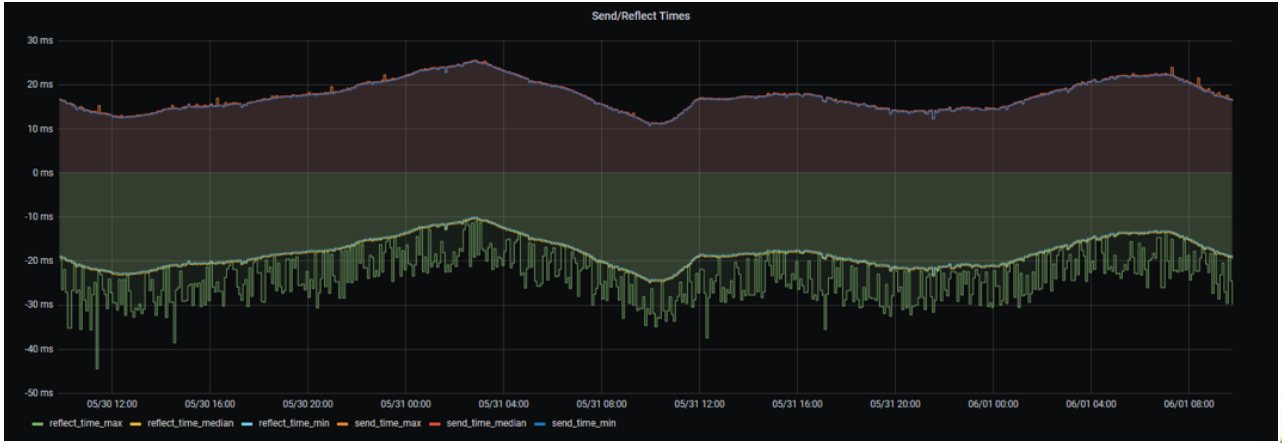
# Re-routing



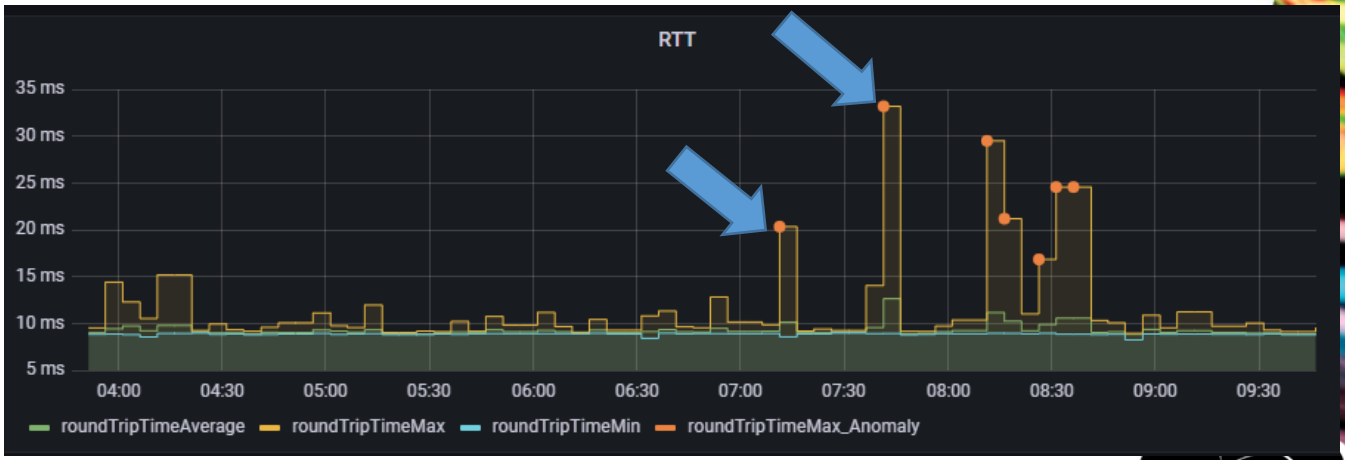

# ECMP effects



# Trends (clocks shifting?)




# Anomaly Detection in action



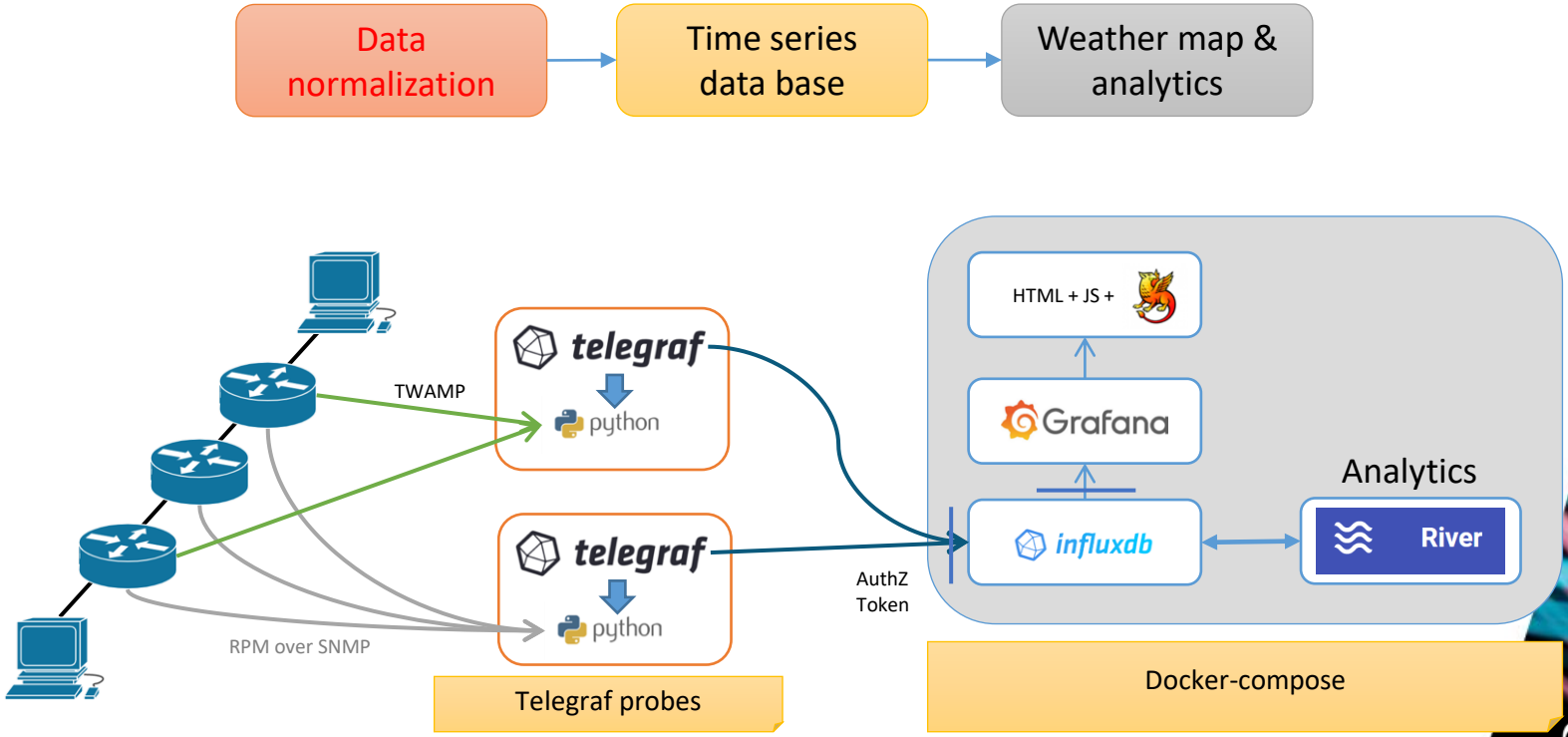
# TimeMap technical requirements

- Scalable micro-services, easy to deploy, **minimal custom code**
- As neutral as possible: **monitoring standards and FOSS**

- Security, with federated access control
  - **eduGAIN** authentication
  - Role Based Access Control, API tokens, multi-tenancy

- **Dynamic**: almost no changes needed when networks change

# TimeMap architecture – 1+ year of data taking

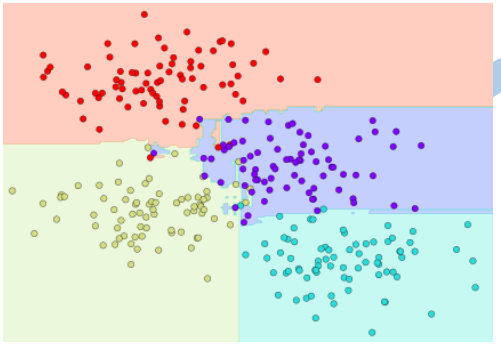


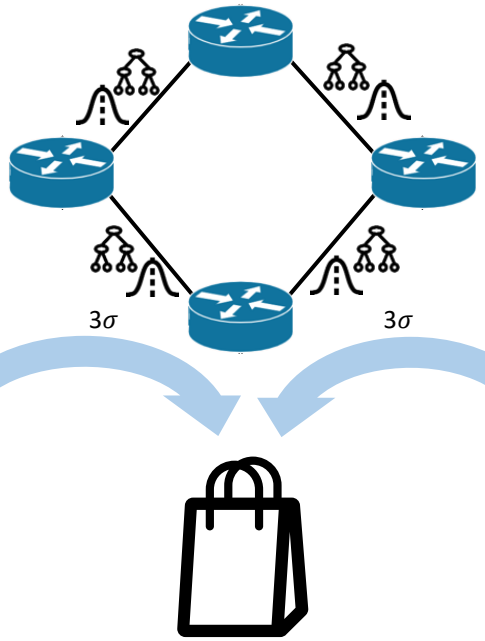
# Anomaly Detection in Timemap – toolset

- Anomaly Detection, in short
  - Std.Dev classification
  - Unsupervised
  - Sensible to overfit

- Streaming Machine Learning
- Light footprint
- Python https://riverml.xyz

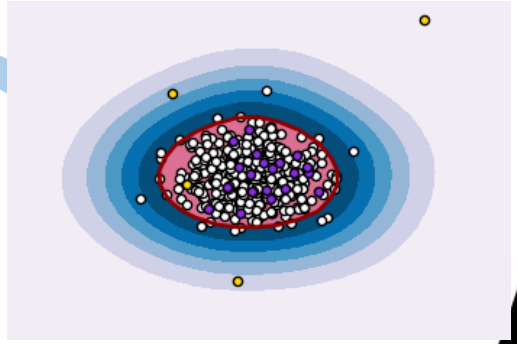
Half-space Random Trees






Model bagging

One-class Support Vector Machine



# On-going – Juniper Twamp probes refactoring

- Benefit from latest libraries for SNMP, gNMI, TWAMP
- Reduce further the ad hoc code
  - Same features, from 250 to 100 lines of code
  - Simpler to read and maintain
- Baseline code for new probes

# On-going: preparing for upcoming new use cases

- Codebase fork for T/F use case
  - New probes for the Flywheels, and other devices if needed
  - Dedicate topology with custom dashboards

- Paving the way for the next GÈANT backbone
  - Adapt Twamp probe to support Nokia devices
  - Interoperability tests
    - Juniper -> Juniper: TWAMP
    - Juniper -> Nokia : TWAMP (Juniper Client, Nokia Server)
    - Nokia -> Nokia : TWAMP-lite

# On-going: Anomaly Detection

- Issues with current models
  - Overfitting and concept drifting
  - Identify when anomalies end
- Enhanced ML models
  - Augment current models with Nowcasting
  - Short range timeseries prediction – training/inference loops
  - Models selection & hyperparameters optimization
- Scouting novel deep learning approaches
  - Digital twin through Temporal Graph Neural Network

GÉANT
Networks • Services • People

# Thank you!

# Questions?

**Fabio.Farina@garr.it**

**timemap-dev@lists.geant.org**

www.geant.org


As part of the GÉANT 2020 Framework Partnership Agreement (FPA), the project receives funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 856726 (GN4-3).