Networking as a Resource

- Job brokerage is critical function of WMS
  - We currently consider storage and CPU resources only in brokerage
  - Networking is assumed – why not treat as a resource to be brokered
  - First, we should try to use network information for site selection
  - Second, can we use provisioning to improve workflow
  - Third, can we improve/modify paths dynamically
  - Plenty of opportunity for future work

Summary

- In the past WMS assumed:
  - Network is available and ubiquitous
  - As long as we implement timeouts, workflow will progress smoothly
  - Computing models can tell us how to design workflows
- What we learned from the LHC:
  - Flexibility in WMS design is more important than computing model
  - Network evolution drives WMS evolution
  - We should start thinking about Network as resource
  - WMS should use network information actively to optimize workflow
  - Resource provisioning could be important for the future
- The future:
  - **A**dvanced **N**etwork **S**ervices for **E**xperiments (ANSE), NSF funded (Caltech, Michigan, Vanderbilt and U Texas Arlington)
  - Next Generation Workload Management and Analysis System for Big Data, PANDA integration with networking, DOE funded (BNL, U Texas Arlington)

Transfers: network-awareness? [1/2]

Where data management could become network-aware?

Level 1: "*high*"-level i.e. **activity planning**

- in some sense, above both Data and Workload Management
- the planning (e.g. dependencies, completion times, ..) drive workflow scheduling and executions
  - network bandwidth reservation could be triggered in advance based on planning details/needs

Level 2: "*medium*"-level i.e. **transfer "routing"**

- *(NOTE: "routing" here intended at the experiment application level, not at the network level)*
- static subscriptions are executed by selecting the "best" source(s) to a destination
- the choice is now based on internal transfer stats (e.g. transfer rates, failures, .. over last days/hrs)
  - network information could be used instead, or additionally

Level 3: "*low*"-level i.e. **file-level transfer**

- could be at the transfer agent level (e.g. FileDownload for CMS PhEDEx) or indeed the underlying file transfer service (FTS)
- all subscriptions and routing would be done in a traditional, network-unaware manner
  - bandwidth allocation may be triggered when the file transfer service needs to deal with a long transfer queue on a link (e.g. threshold?)

Examples? See next slide.

Transfers: network-awareness? [2/2]

Level 1: "*high*"-level i.e. **activity planning**

- *subscriptions in Rucio may be an interesting candidate for a choice at this level?*
  - replica management based on **Replication Rules** defined on datasets/containers. Each rules is owned by a Rucio "**account**", and defines the minimum # of replicas that have to be available on a Rucio Storage Element (**RSE**), i.e. a storage space with attributes. RSEs can be grouped in logical ways (e.g. CLOUD=US, or Tier=1). Accounts manage (and are charged) for their own data with replication rules defined on datasets/containers and lists of RSEs
  - *Could a translation of such a rule into a concrete list of transfer tasks be engineered to be optimized on the basis of network-aware information? (e.g. naively: "choose the source RSE with best connection to the destination RSE"?)*

Level 2: "*medium*"-level i.e. **transfer "routing"**

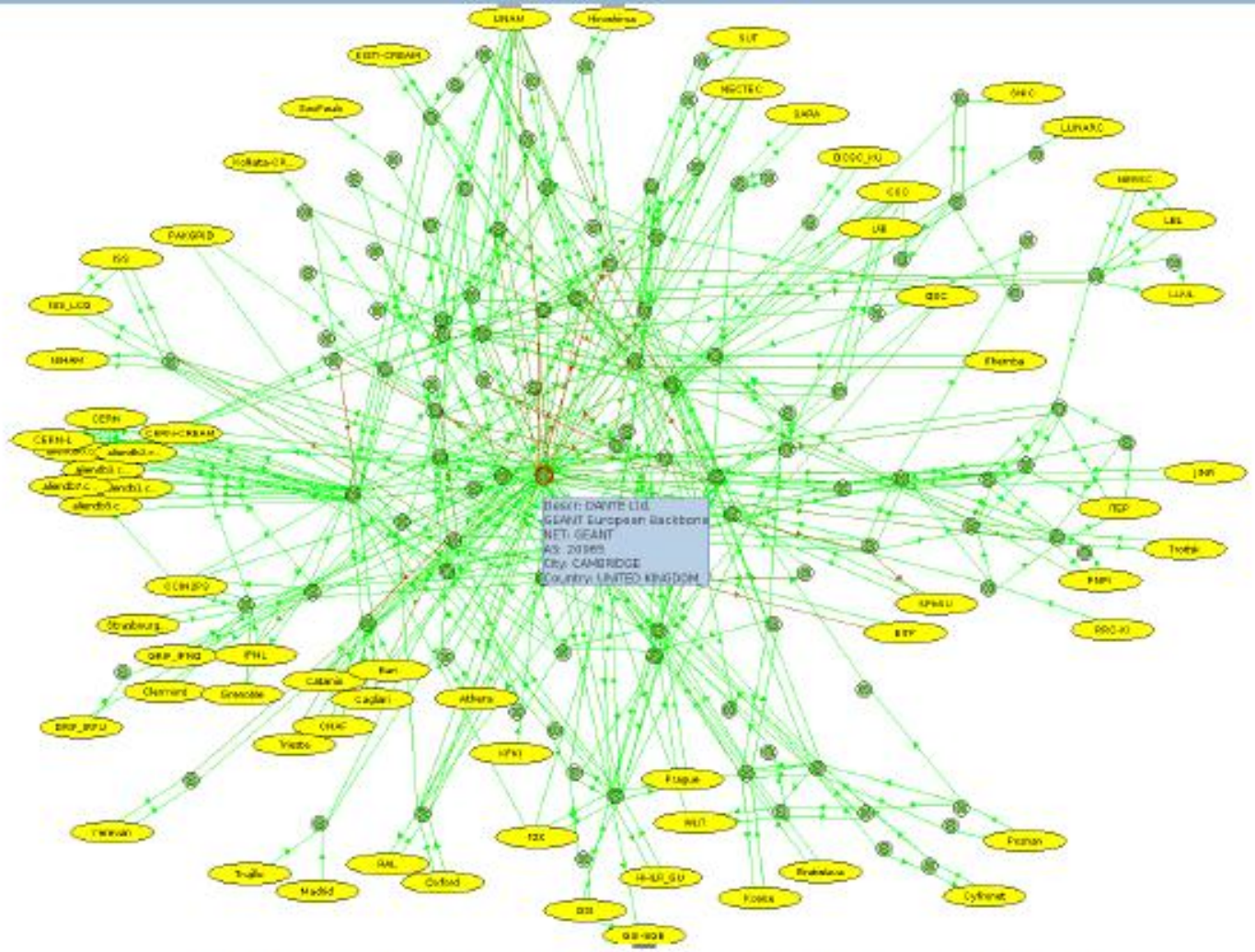
- *ATLAS Site Services or PhEDEx FileRouter could use network info at this level?*

Level 3: "*low*"-level i.e. **file-level transfer**

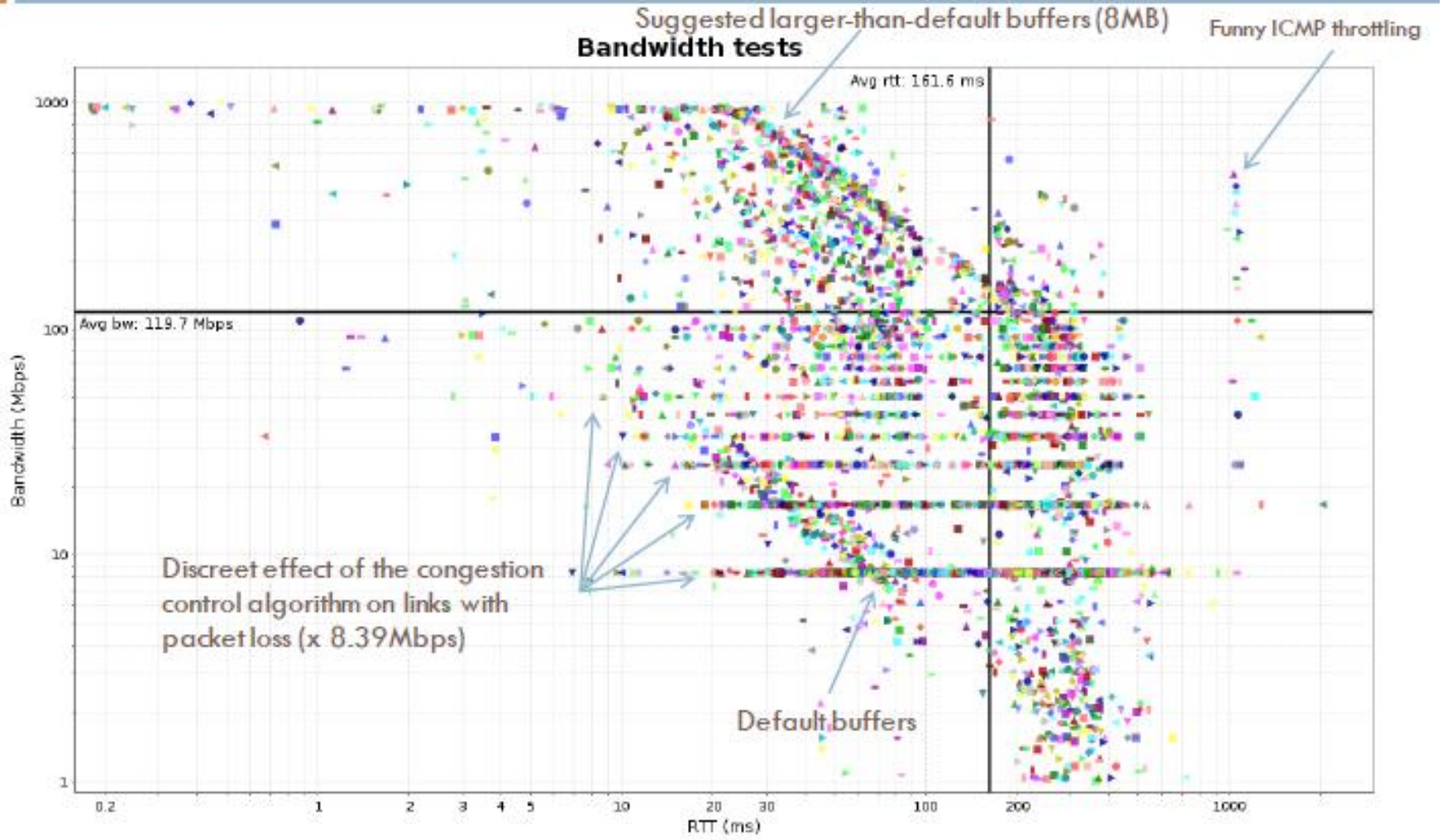
- *e.g. FDT used as the backend in the FileDownload agent in PhEDEx on the /Debug instance on just one link may be an existing proof of concept of a choice at this level?*

Food for thoughts...

Network topology view in MonALISA



Available bandwidth per stream



NSI ‘wishlist’

- Authentication/Authorization
  - Federated? User-based? NREN level? Standardized!
- Pathfinding, topology discovery and exchange
  - Real-time topology (incl capacity)? Per user (screened) ?
  - Path building - chain / tree? Explicit Routing Object?
- Aggregation
  - DIY aggregation in NEXPReS client
- Monitoring
  - Fault tolerance, maintenance, automatic/manual re-routing?
- Status of NSI clients? Full citizens?
- Bandwidth! Reach!