P2P and DAaM

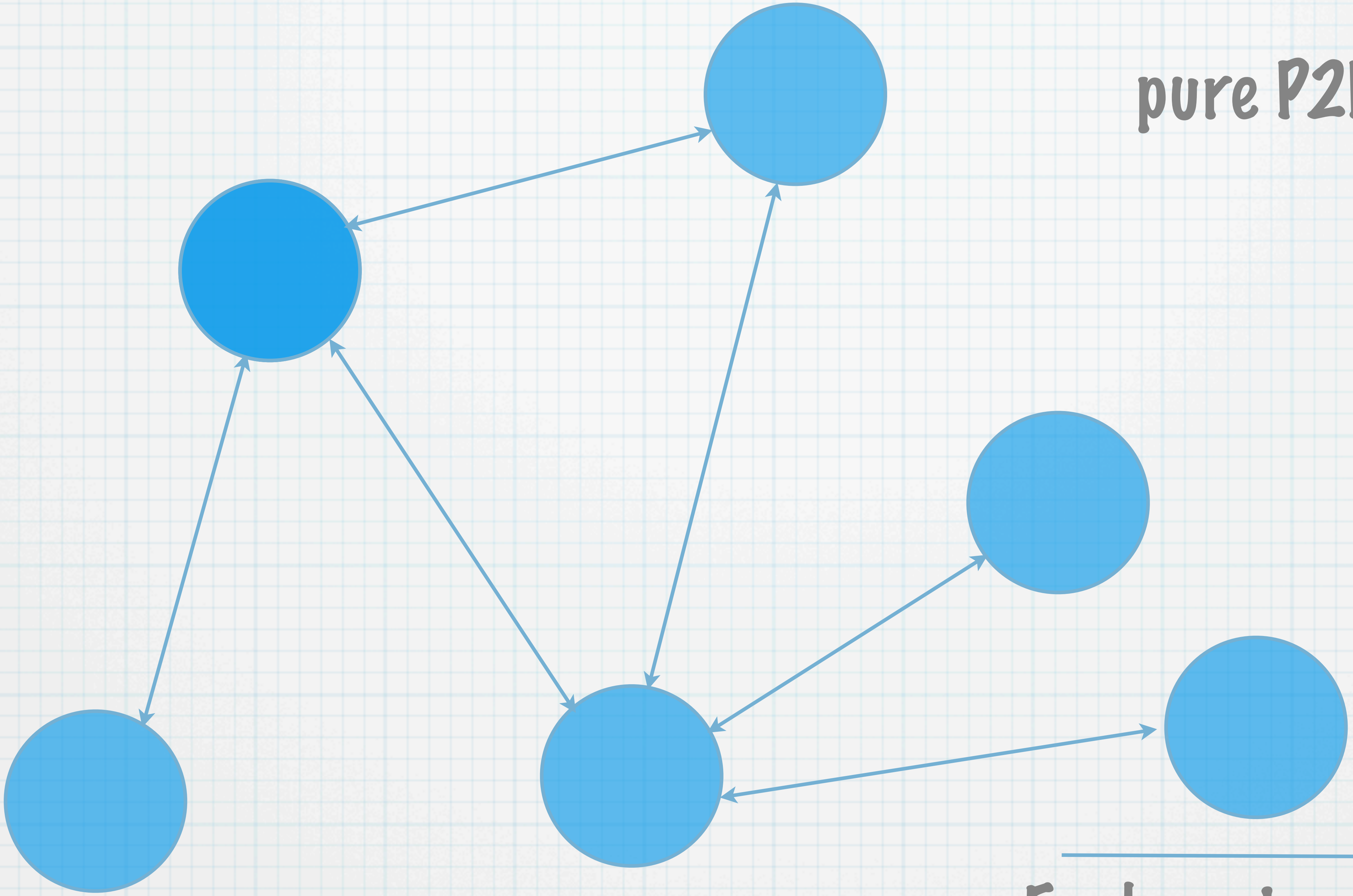
Jeff Templon, Nikhef

Thanks : the Nikhef under-30 DM club
(see Koeroo doc), Kors, & Ian

wikipedia

In P2P networks, clients provide resources, which may include bandwidth, storage space, and computing power. As nodes arrive and demand on the system increases, the total capacity of the system also increases. In contrast, in a typical client–server architecture, clients share only their demands with the system, but not their resources. In this case, as more clients join the system, less resources are available to serve each client.

pure P2P



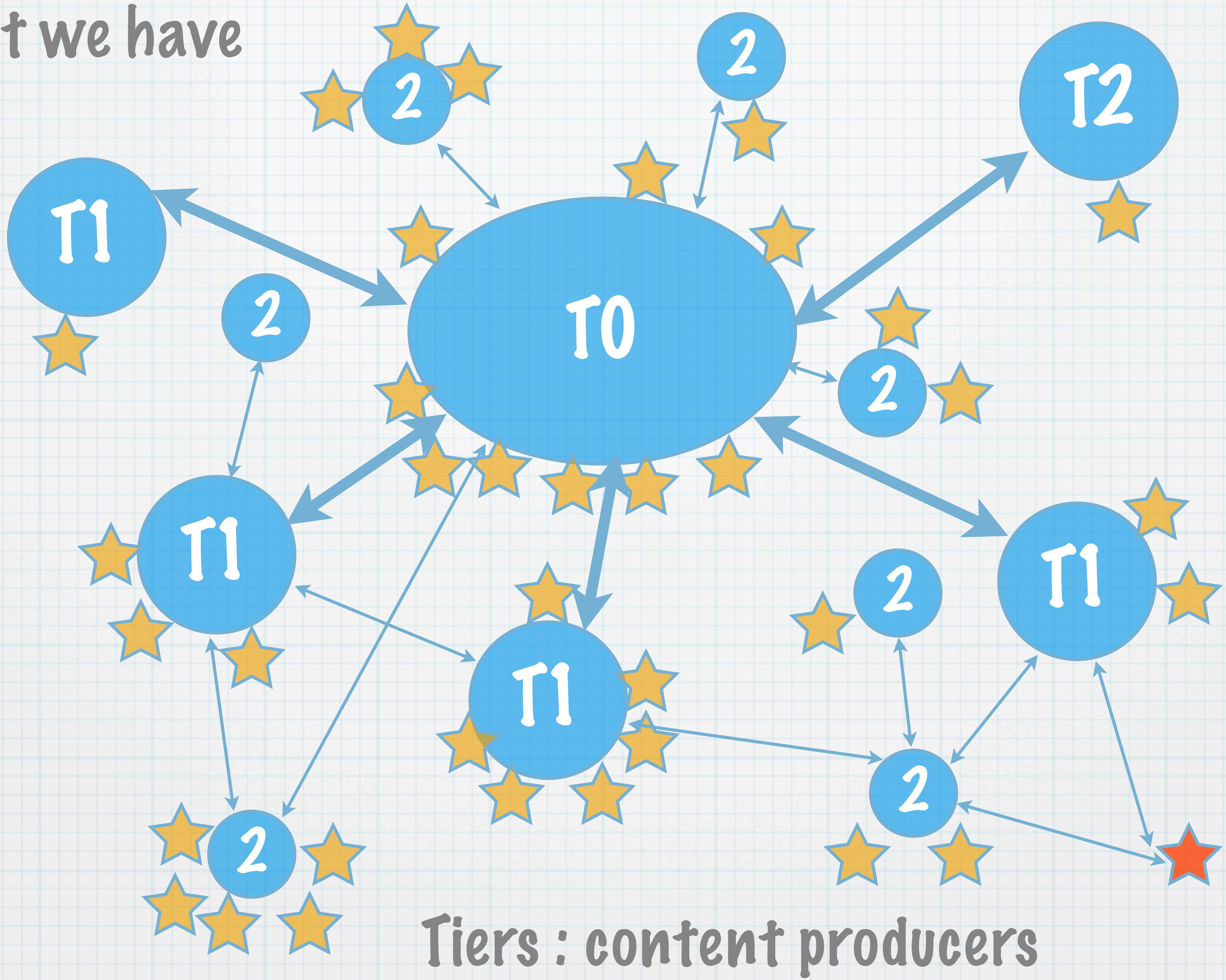
Each node:
wants files;
provides files;
data on node is "local"

wikipedia

In P2P networks, clients provide resources, which may include bandwidth, storage space, and computing power. As nodes arrive and demand on the system increases, the total capacity of the system also increases. In contrast, in a typical client–server architecture, clients share only their demands with the system, but not their resources. In this case, as more clients join the system, less resources are available to serve each client.

Thought : include WNs in P2P layer, then it works ...
BitTorrent DNA?

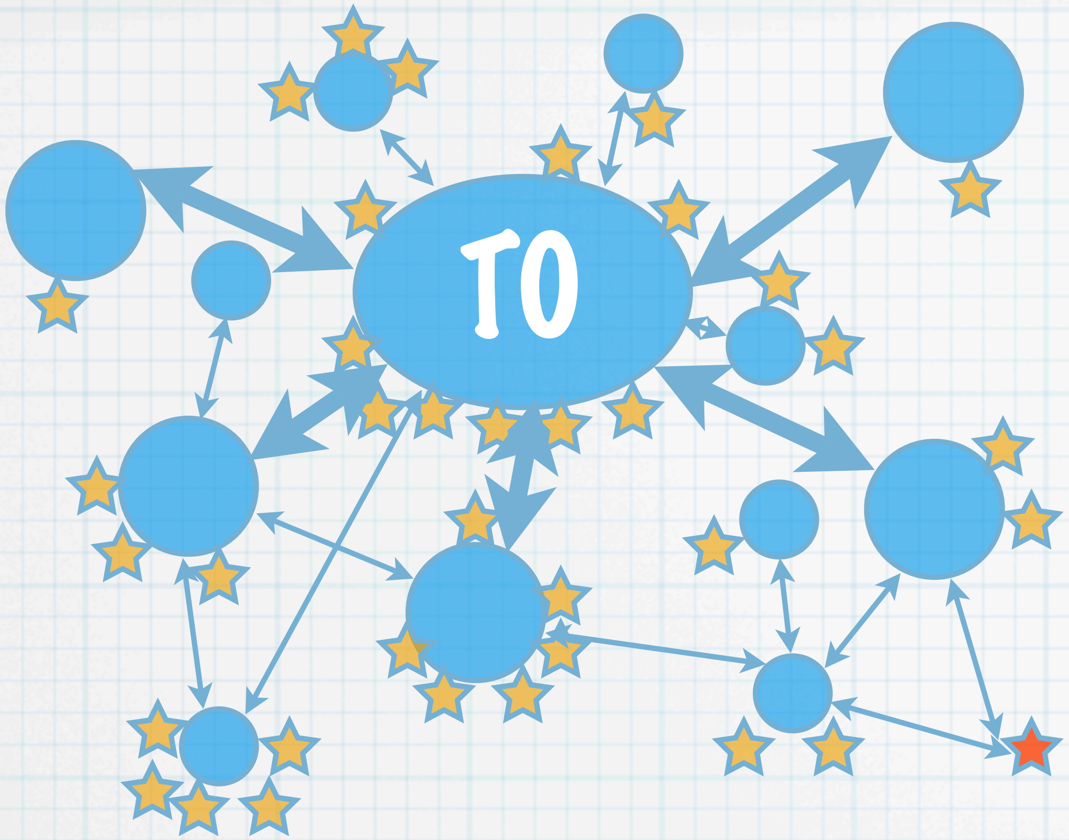
what we have



Tiers : content producers

also providers for users ★

not all connections or sites are equal



Looks sort of like a Content Distribution Network

CDNs can dynamically distribute assets to strategically placed redundant core, fallback and edge servers. CDNs can have automatic server availability sensing with instant user redirection. A CDN can offer 100% availability, even with large power, network or hardware outages.

CDN technologies give more control of asset delivery and network load. They can optimize capacity per customer, provide views of real-time load and statistics, reveal which assets are popular, show active regions and report exact viewing details to the customers. These usage details are an important feature that a CDN provider must provide, since the usage logs are no longer available at the content source server after it has been plugged into the CDN, because the connections of end-users are now served by the CDN edges instead of the content source.

CDN nodes cooperate with each other to satisfy requests for content by end users, transparently moving content to optimize the delivery process.

CDN terms

- **origin server** - 'home' for a given file. permanent location. delete = gone forever. **archive management (SRM?)**

- **proxy / cache** - clients access file from these. Files may disappear or even proxy itself ... doesn't matter. **performant data delivery**

CDN + P2P

- Some CDNs are overlay on P2P
  - collaboration between proxies
  - Distributed Hash Tables for quick, reliable file resolution within proxy network (Pete's catalogue yesterday?)

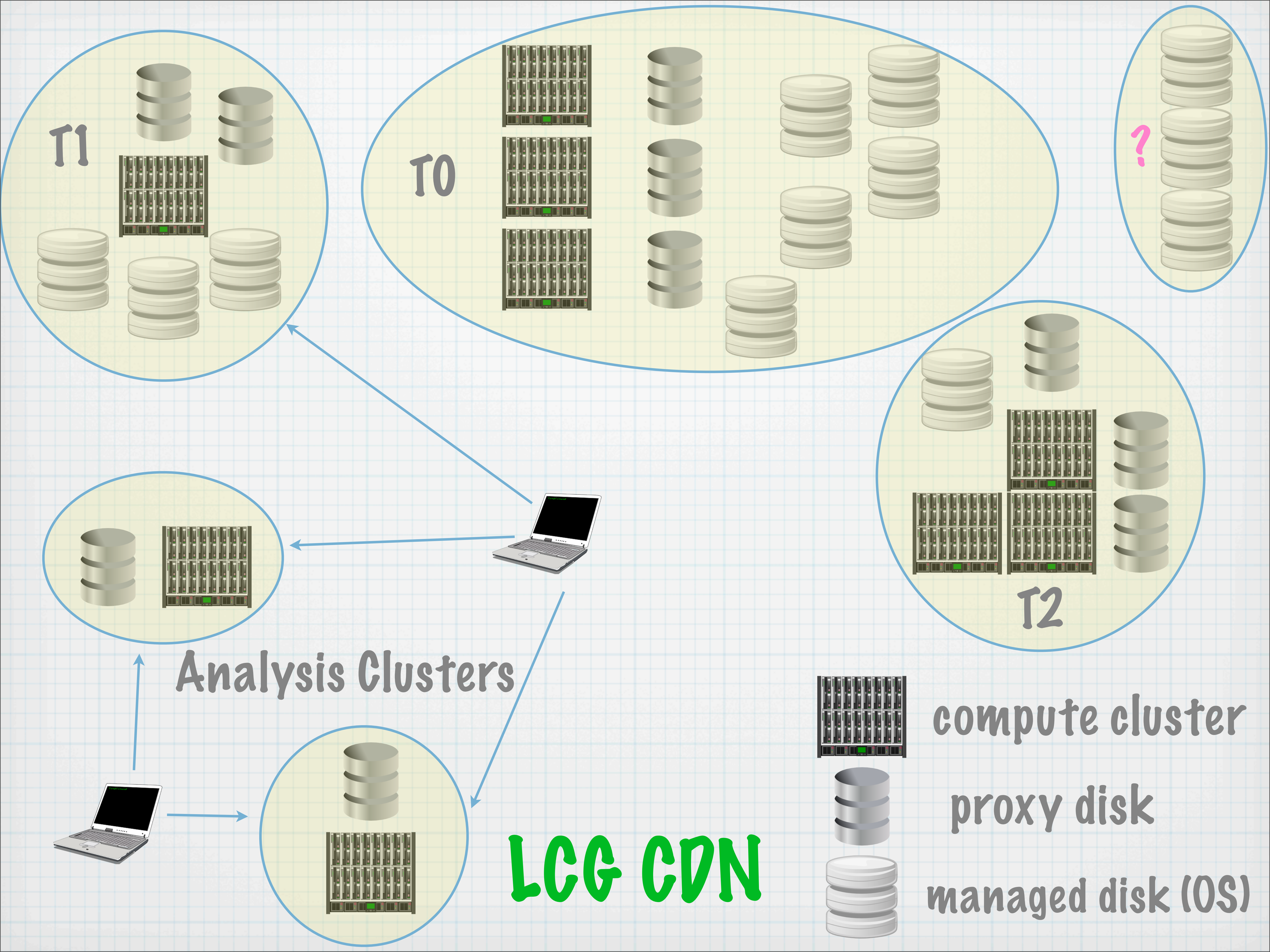
“get file” in CDN

- user : “get” on URL. DM system all in same TLD (all urls like cdn.cern.ch/somefile).
- DNS resolves to a close proxy (or anycast)
- if chosen proxy has file : return it.
- Otherwise, chosen proxy looks for a close proxy that has it, gets it, and return it.
- if no close proxy has it, chosen proxy gets it from origin server and returns it

consequences

- Clients never access “origin servers”; SRM then has fewer problems to solve (management of large disk, tape)
- T1 storage down? Who cares?
- Popular files rapidly everywhere, boring files only on origin servers
- caches are read-only from user perspective : simpler. focus on fast

Less entanglement!!!



LCG CDN

T0

T1

T2

?

Analysis Clusters

compute cluster

proxy disk

managed disk (OS)

What is available

- many commercial CDNs.
- CoralCDN (http://www.coralcdn.org/)
- CoDeen web cache network (same group)
- I asked Coral group leader some questions, answer was, they want to work with us!

what is missing

- protocols (only HTTP now) ... or do we want a virtual file system interface?
- security (most CDNs assume all is public)
- still need to get data into system : reliable file transfer daemon @ site (cf Claudio's paper)

CDN Strawman

- O(1) origin server for each data
- each computing site has proxy, with associated cache disk
- proxies and origin servers : separate machines
- use CoralCDN to organize proxies; add
  - security
  - VFS interface.
  - storage/bandwidth quotas?