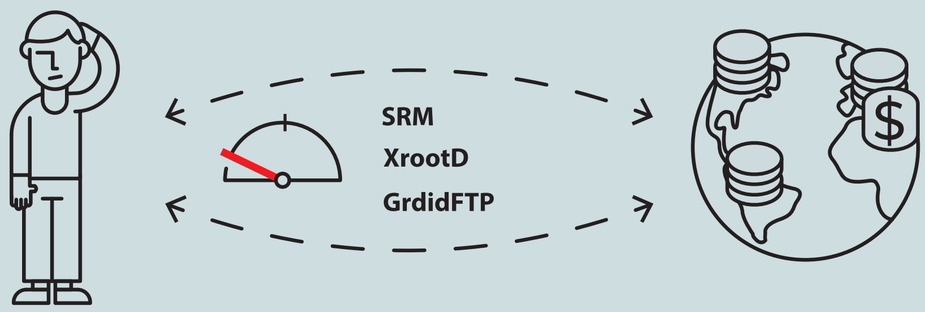
SCORES: A DYNAMIC CACHE SYSTEM for e–Science applications

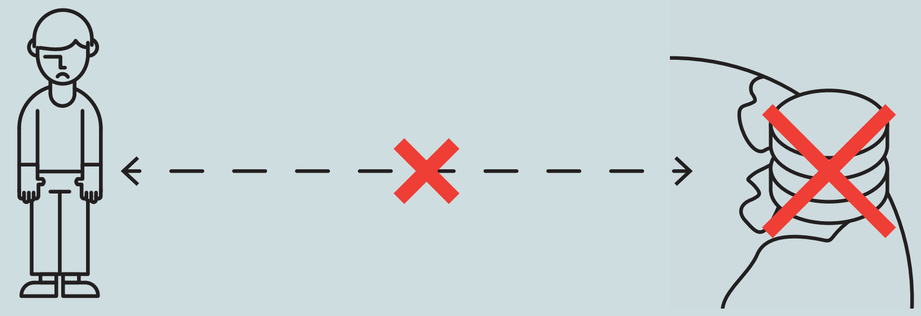
[a,b]Davide Michelino

[a] Consortium GARR – Via dei Tizii, 6 –00185 - Roma - Italy
[b] INFN-Napoli - Campus di M.S.Angelo Via Cinthia– 80126, Napoli, Italy
e-mail: davide.michelino@na.infn.it

WITHOUT CACHE

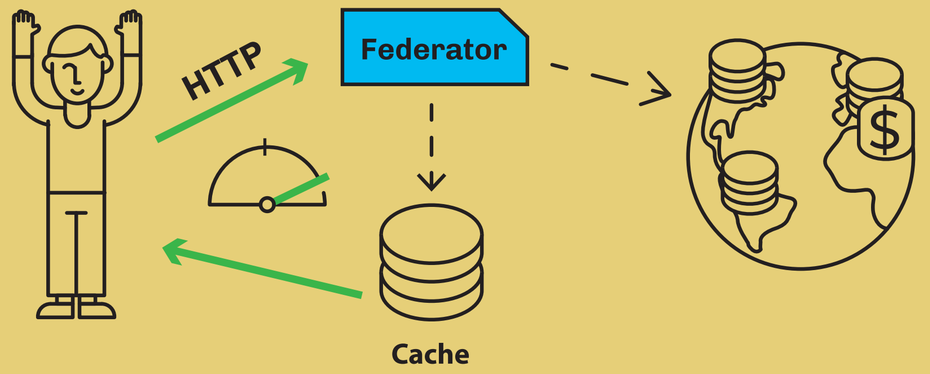


- Multiple access protocols
- Slow performances against high latency storage
- Expensive access to Pay-per-use storages

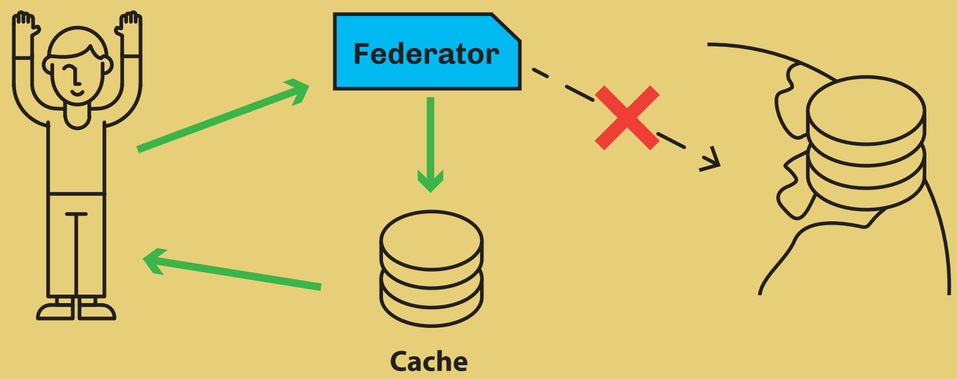


- Limited dynamicity
- Storage failures can produce data unavailability
- Network failures can produce data unavailability

WITH CACHE + FEDERATOR



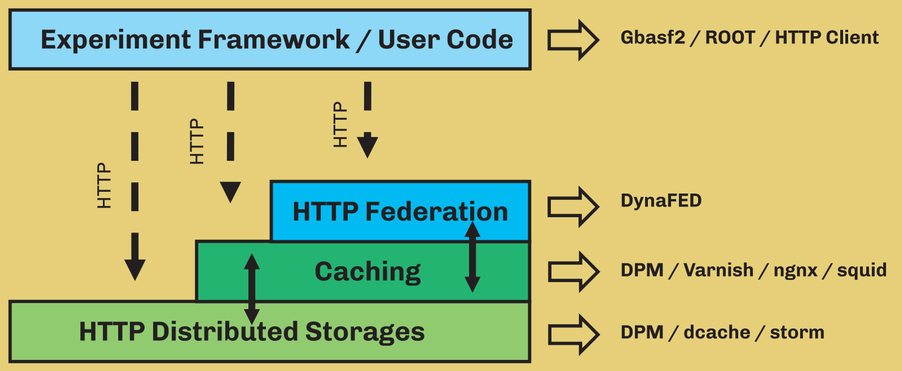
- Single standard protocol
- Single namespace view to access all data
- High performances thanks to caching
- Reduce accesses to Pay-per-use storages



- High dynamicity
- At least 2 replicas per file (remote storage(s)+cache)
- More resilient to storage unavailability

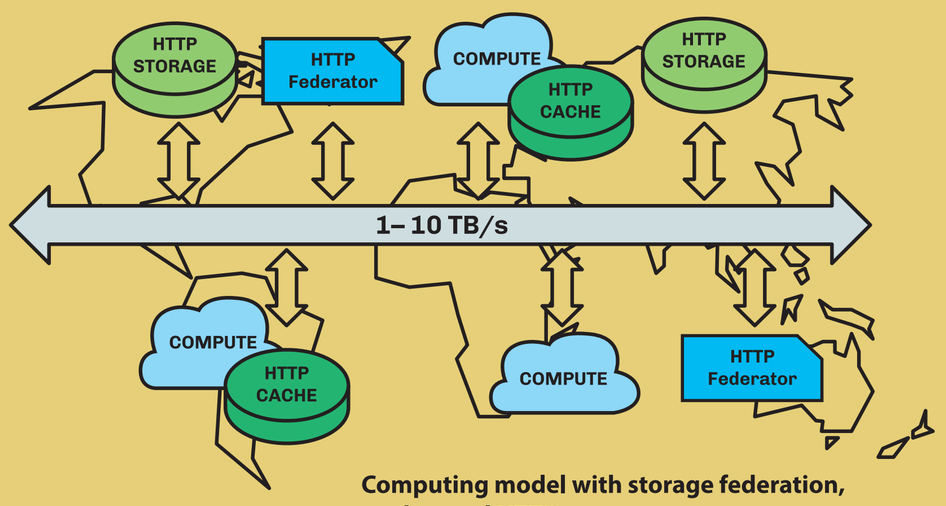
STACK IMPLEMENTATION

- Enable the usage of standard tools like browser
- The cache level can be introduced or removed transparently
- The cache level can be implemented with different technologies
- The storage level is technology agnostic



OVERALL ARCHITECTURE

- Data caches can be placed in different sites
- Each site can add its own federator
- Exploit new computing model paradigsm
  - Site without pledged storage
  - Use opportunistic resources
- Cache can serve a local site or run as geographic service



Computing model with storage federation, caches and HTTP access