

# Summary of issues and questions raised



## Summary of use

- Generally positive response on current state!
- Now the issues, questions and missing features...

# CMS



- Which network model ?
  - Store-and-forward or direct transfer
- Single most noteworthy lacking features:
  - checksum (or at list file size) check after transfer
    - (file size already checked, but no check-summing yet)
- Other desiderata:
  - FTS stays in "waiting" even if file already exists at destination and SRM.put fails
    - DISCUSS retry policy
  - Would like to have full timing info for each transfer: request, start, end



# Alice

- **Transparent use**
  - ALICE requires the automatic discovery of the FTS Endpoints and the names of the FTS proxies servers through the information system
  - An upper layer able to hide the transfers among the different SRM
- **Tool for traffic monitoring**
  - Possibly R-GMA?
- **Closer coordination of FTS and LCG2 releases**
  - Updating the clients on VO-Box by hand is a pain
- **Things not needed soon (or at all?):**
  - Catalog update (handled externally)
  - LFN resolution (handled externally)
  - Tx-Ty predefined channels not needed [ Transparent use? ]
    - Deal in terms of site name or SRM names

# LHCb



- Pre-staging needed: timeout period of FTS Transfer Agent
  - 'Failed on SRM get: SRM getRequestStatus timed out on get'
  - Currently LHCb pre-stage externally.
- Different behaviour of dCache/Castor
  - On file overwriting? Can we do better? Overwrite flag? Wait for SRMv2?
  - Related problem in SRM v1: No single method to perform physical removal
- advisoryDelete problem
- Better retry
  - Don't retry 'lost-cause' files e.g. source isn't there

# LHCb



- Plugins
  - Interested in LFC plug-in (can share some experience on this!)
- T1 to T1 channel use-case
  - Ideally full connected mesh over ~6 T1 sites
  - T2 to T0 mentioned as well
    - DISCUSS model
- Transparency:
  - Central service to submit
  - Road-map suggested: what can we do *now*

# ATLAS



- ReplicaVerifier
  - Done already by FTS(?)
- Staging
  - Find out stage status of file
- Plug-ins:
  - Catalog interactions - not just grid catalog - multiple catalog updates
  - Zipping file plug-ins
  - Call-backs to avoid polling
- Retry policy and Hold states
  - Can FTS retry more! (except 'permanent-error' jobs)
- Priorities
  - Already done: maybe allow high-pri submit for VO manager as well
- TierX to TierY transfers handled by the network fabric, so channels between all sites should exist
  - Routing... data planner.
- Bandwidth monitoring
- Error reporting issues (who do we call / mail?)



## Other things

- All experiments poll
  - Can we use the statistics from FTS to help with this - reduce the polling time?
  - Call backs?
- Retry logic and Hold state
  - We should discuss the model here
- VO manager role
  - Who gets to modify what?
- Interface stability
  - How often is it OK to make changes that may break things





## Summary of issues / features

- Work our retry logic and Hold states
- Pre-staging
  - Configurable to be on or off
- "Central service" or equivalent: (data planner and scheduler)
  - LHCb suggested road-map: client tool first
- VO customisation (plug-ins)
  - LFC interaction
  - Other catalog interactions
  - Pre/post steps
  - Agree on convention for this
- How to handle T1 to T1
  - Also T2 to T2 and T2 to T0?
- Bandwidth monitoring
  - Better channel monitoring
- What is the model
  - Store and forward



## Summary of issues / features

- Fix advisoryDelete problem
  - Deletion
- Full transfer history
- Timing information + number of retries summary
- Check-summing and validation
  - Simple file size check.. Check-summing harder via SRM
- Interface stability
  - Frequency of update?
- Process:
  - Just who do we email when it goes wrong?