

SUMMARY of GFS F2F at IBM Almaden (06/AUG/2004)

Objective: Gather vendor expectations/input, present status and discuss issues.

Present:

- Osamu Tatebe, AIST, Japan
- Jane Xu, IBM Systems and Technology Group
- Arun swaran Jagatheesan, San Diego Supercomputer Center (SDSC)
- Leo Luan, IBM Almaden Research Center
- Manuel Pereira, IBM Almaden Research Center
- Cameron Bahar, Storage Machines
- Naveen Nalam, Storage Machines
- Alan Yoder, Network Appliance

Vendor Expectations and Suggestions

1. Speaker: Cameron Bahar, Storage Machines

1.1. Expectations:

- 1.1.1. Plug-n-play utility model of storage and data would be useful
- 1.1.2. Use existing protocols for access
- 1.1.3. Clearly define management and security interface for plug-n-play of storage resources

2. Speaker: Alan Yoder, Network Appliance

2.1. Expectations

- 2.1.1. No new clients
- 2.1.2. *How much of "global" is global?*
 - 2.1.2.1. Leo/Jane: Global as in inter-organizational and wide-spread
- 2.1.3. WAN-file system: has some complexities
 - 2.1.3.1. Jane: multiple vendor participation
- 2.1.4. Usage from a Grid data centers perspective
- 2.1.5. Move data around transparently on demand
- 2.1.6. Standards-based
 - 2.1.6.1. Don't introduce new clients. Changes in wide-spread client-side unlikely (introduces more complexities for administration)
 - 2.1.6.2. CIFS, NFS as main targets.
 - 2.1.6.3. Leo: DAFS? – its an exception
- 2.1.7. Namespace management only
 - 2.1.7.1. Looks like local to the users
 - 2.1.7.2. (e.g.) data from archive to online
 - 2.1.7.3. ILM on demand
- 2.1.8. EGA: Enterprise Grid Alliance
- 2.2. What can be learnt from similar experiences with NFS and AFS
 - 2.2.1. Why didn't we do this before?
 - 2.2.2. Cam: We have wide-spread infrastructure and protocols more mature that were not available before
- 2.3. Having real engineers do the real work

- 2.3.1. Focus on getting real products
- 2.3.2. Engineering management supported interest
- 3. Speaker: Jane Xu, IBM (*Request to upload slides to Gridforge*)
 - 3.1.1. GFS requirements
 - 3.1.1.1. Geographically distributed
 - 3.1.1.2. Good performance
 - 3.1.1.3. Simple management
 - 3.1.1.4. Support for different physical file systems
 - 3.1.1.5. Different data ownership models
 - 3.1.1.6. Security
 - 3.1.2. Global Namespace – federation
 - 3.1.2.1. Unifies data sources from heterogeneous data
 - 3.1.2.2. Background data management autonomously
 - 3.1.2.3. Inter/Intra enterprise
 - 3.1.3. Caching/Replication
 - 3.1.3.1. WAN latency
 - 3.1.3.2. Caching on demand (autonomous)
 - 3.1.4. Standardization
 - 3.1.4.1. Global namespace – Call it “GFDS” or “VFDS”
 - 3.1.4.2. Distributed caching and related standards
 - 3.1.4.2.1. NFSv4 extension
 - 3.1.4.2.2. IETF NFSv4 WG
 - 3.1.4.3. Wide Area Replication
 - 3.1.4.3.1. GGF Replication Service WG (OGSA-REP)
 - 3.1.4.4. We need standards to enable single logical rendering of heterogeneous file systems for Grid Computing
 - 3.1.4.4.1. API or protocol or web service (java, c++, usage)
 - 3.1.4.4.2. GGF GFS WG
 - 3.1.4.4.3. SNIA FS WG (more on storage management)
 - 3.1.5. Discussions for Architecture
 - 3.1.5.1. Global Namespace service
 - 3.1.5.1.1. Allow NFS or CIFS server rendering
 - 3.1.5.1.2. Vendors can choose any uniform namespace
 - 3.1.5.2. WAN File System Management API
 - 3.1.5.2.1. Replication
 - 3.1.5.2.2. Caching
- 3.2. Speaker: Leo Luan, IBM Research
 - 3.2.1. Based on needs in the industry/IT products
 - 3.2.2. Does not necessarily need to be a new service
 - 3.2.2.1. Schema, syntax,
 - 3.2.2.2. Leverage existing service
 - 3.2.2.3. Store namespace – object
 - 3.2.3. No new clients
 - 3.2.4. Standards
 - 3.2.4.1. Web-Service based mechanism
 - 3.2.4.2. Existing protocols first

- 3.2.4.3. Generalize after that
- 3.2.5. Other issues
 - 3.2.5.1. Security mechanisms
 - 3.2.5.2. Federated view of heterogeneous mechanisms

Status of Grid File System Directory Services

4. Discussion of Grid File System Directory Services document (*Please refer the document from GridForge*)

- 4.1. Virtual Directories
- 4.2. Junctions
- 4.3. CIFS
 - 4.3.1. CID – too big for I-node
 - 4.3.2. DFS referral
 - 4.3.3. ACTION: Standardize extended schema
- 4.4. Changes needed
 - 4.4.1. Create
 - 4.4.1.1. Update relative path
 - 4.4.1.2. Changes in Entry
 - 4.4.2. ReSync
 - 4.4.2.1. Check parent ID
 - 4.4.2.2. Check client ID

Discussion on GFS Architecture to allow plug-n-play of GFS from multiple vendors

5. It was concluded that more discussion needs to be made on the architecture

- 5.1. Issues that need to be clarified
 - 5.1.1. Is it possible to have a CIFS client -> GFS client interface mapping?
 - 5.1.1.1. All of rather agreed on this the feasibility
 - 5.1.1.2. Would it be scalable without performance penalty
 - 5.1.2. To provide plug-n-play of resources and resource namespaces, each vendor/implementer must allow GFS-2-GFS communication with a standard interface. Do we use a Web/Grid Service for inter-GFS communication? Do we use XML on the wire?
 - 5.1.3. Do we base our work on other standards within GGF that might not have any implementations?
 - 5.1.4. Can inter-organizational sharing of data or storage be achieved without cross-registering users (just by trusting the other CA or other organization)?
 - 5.1.5. Arun agreed on writing up more on the architecture document and presenting it to the list. This would enable further discussions with more vendor participation to know the feasibility and value of the architecture.