A Classification and Evaluation of Data Movement Technologies for the Delivery of Highly Voluminous Scientific Data Products

Chris A. Mattmann, Sean Kelly, Dan Crichton, Steve Hughes, Sean Hardman, Paul Ramirez, Ron Joyner

Jet Propulsion Laboratory

NASA/IEEE Conference on Mass Storage Systems and Technologies

What we were trying to accomplish

- Understand the tradeoffs amongst different large-scale, data movement technologies
  - Develop a classification framework
- Help tackle a real world problem
  - MRO mission and upcoming LRO missions
- Planetary Data System
  - NASA's science data archive for all solar system missions
  - Nodes distributed across 9 different centers (called "nodes")

Our approach

- Select technologies to evaluate
  - Commercial and open source UDP bursting technologies, GridFTP, bbFTP, hard-media, FTP, SCP
- Identify dimensions for classification
  - Cost to operate/implement, scalability, reliability, ease of use, transfer rate, industry adoption
- Classify data movement technologies along dimensions

Classification

Technology Approach	Scalability	Reliability (Dependability)	Ease of use	Efficiency	Transfer Rate	Cost to Operate	Cost to Implement	Industry Adoption	Pro/Con
CD	Constant time to transfer small data volumes (gigabytes) – 600 MB/CD	Highly reliable	Easy to use and deploy (standard hardware component on most machines)	Low efficiency, only 600 MB-700 MB per CD-R	Constant – around 600 MB/disc sent	..25 FTE	1 FTE procurement alone + .25 FTE to maintain	1 (pervasive)	+ pervasive and easy to use - data volume too low
Data Brick	Constant time (48 hours estimated) to transfer small to large (petabyte) volumes	Highly reliable	Unclear how easy to use, special hardware required	High efficiency, can store terabytes of information on a single brick	Constant – anywhere from GB to TB/brick	~1 FTE, (special hardware and support)	.5 FTE procurement + .5 FTE to maintain	2 (starting to gain attention)	- data volume increase very high - may need special hardware/training
Blue Ray Disc	Constant time to transfer small to large (petabyte) volumes – 25 GB/disc	Highly reliable	Easy to use and deploy (will require special hardware)	Medium efficiency, 25 GB/disc	Constant – 25 GB/disc	.5 FTE (special hardware)	.5 FTE procurement + .5FTE to maintain	2 (starting to gain attention)	+ data volume increase from DVD - still volume too low
HD DVD	Constant time to transfer varying sizes of data (from megabyte-petabyte) – 15 GB/disc	Highly reliable	Easy to use and deploy (will require special hardware)	Medium efficiency, 15 GB/disc	Constant – 15 GB/disc	.5 FTE (special hardware)	.5 FTE procurement + .5 FTE to maintain	2 (starting to gain attention)	+ data volume increase from DVD - still volume too low
DVD	Constant time to transfer small to medium (terabyte) volumes – 4.7 GB/DVD	Highly reliable	Easy to use and deploy (standard hardware component on most machines)	Low-Medium efficiency, 4.7 GB/DVD	Constant – 4.7/DVD	.25 FTE	1 FTE procurement alone + .25 FTE to maintain	1 (pervasive)	+ pervasive and easy to use - data volume too low
FTP	Linear scalability based on dataset sizes from 40 MB to 1 GB.	Fault rate dependent on underlying TCP/IP protocol, but 0 faults / 20 hours of testing and 10s of GBs of data.	Easy to deploy (standard component on Linux/UNIX/Mac, and some Windows solutions). Well documented command line params.	Outperforms UFTP on datasets > 200 MB. Outperforms Aspera on all datasets tested.	6.272 Mbps on JPL LAN on average for datasets from 40 MB to 1 GB.	Approx .25 FTE (sound documentation, installation and operation procedures)	Approx .05 FTE for any tailoring necessary, otherwise freely distributable (under GPL and open source).	1 (pervasive)	+ pervasive technology + great documentation + easy to use - limited by underlying TCP/IP
bbFTP	Directly proportional to dataset size	High reliability	Easy to deploy on Unix based systems with /etc/passwd security. Can also use Globus GSI security.	Outperforms FTP by only fair amounts on JPL LAN	Configurable parallelism; max on JPL LAN = 11.060 Mbps	Approx 0.25 FTE (similar usage scenarios to ftp)	Approx 0.25 FTE (simple to build/deploy to Unix-style systems)	4 (Low penetration)	+ easy to use + easy to deploy – Unix only? – Better on WANs?
GridFTP	Directly proportional to dataset size	High reliability; protocol supports retransmit and reassembly	Difficult to deploy; relies on Grid Security Infrastructure and certificate management for hosts, users, services	Outperforms FTP by only fair amounts on JPL LAN	Configurable parallelism and block size; max on JPL LAN = 11.639 Mbps	Approximately 2 FTE (software manager + certificate manager)	Approximately 1 FTE	2 (everyone's jumping onto grids)	+ highly secure + open source + industry backing – Requires a PKI and staff – Better on WANs?
UFTP	Very good for 40MB-200MB datasets, after that, very poor.	Fault rate dependent on maximum TX rate: difficult to find correct rate.	Difficult to discern command parameters.	Faster than FTP and Aspera for volumes between 40 MB-200MB. Afterwards, slower than both.	3.49713 Mbps on JPL LAN on average for datasets from 40 MB to 1GB.	approx 1 FTE + software support agreement	approx .5 FTE for implementation tailoring open source product, so $0 procurement	5 (bleeding edge technology)	+Effective for bursts of data to multiple clients from one server (multicast) - Difficult to implement and operate.
Aspera	Near linear scalability based on dataset sizes from 40 MB to 1 GB.	High reliability: no faults over any benchmarks so far: 0 faults/20 hours of testing and 10s of GBs of data	Easy to deploy (Linux RPM, Mac/Windows Installers), but undocumented command line params.	Poor performance on JPL LAN (outperformed by FTP and UFTP)	1.46 Mbps on average for JPL LAN for datasets from 40 MB to 1 GB.	approx 1 FTE + software support agreement from Aspera	Approx .25 FTE for tailoring + licensing cost (per set software license agreement based on bandwidth)	3 (somewhat widespread use, but also somewhat bleeding edge)	+ easy to deploy + easy to execute - difficult to understand parameters - only good on WANs seemingly

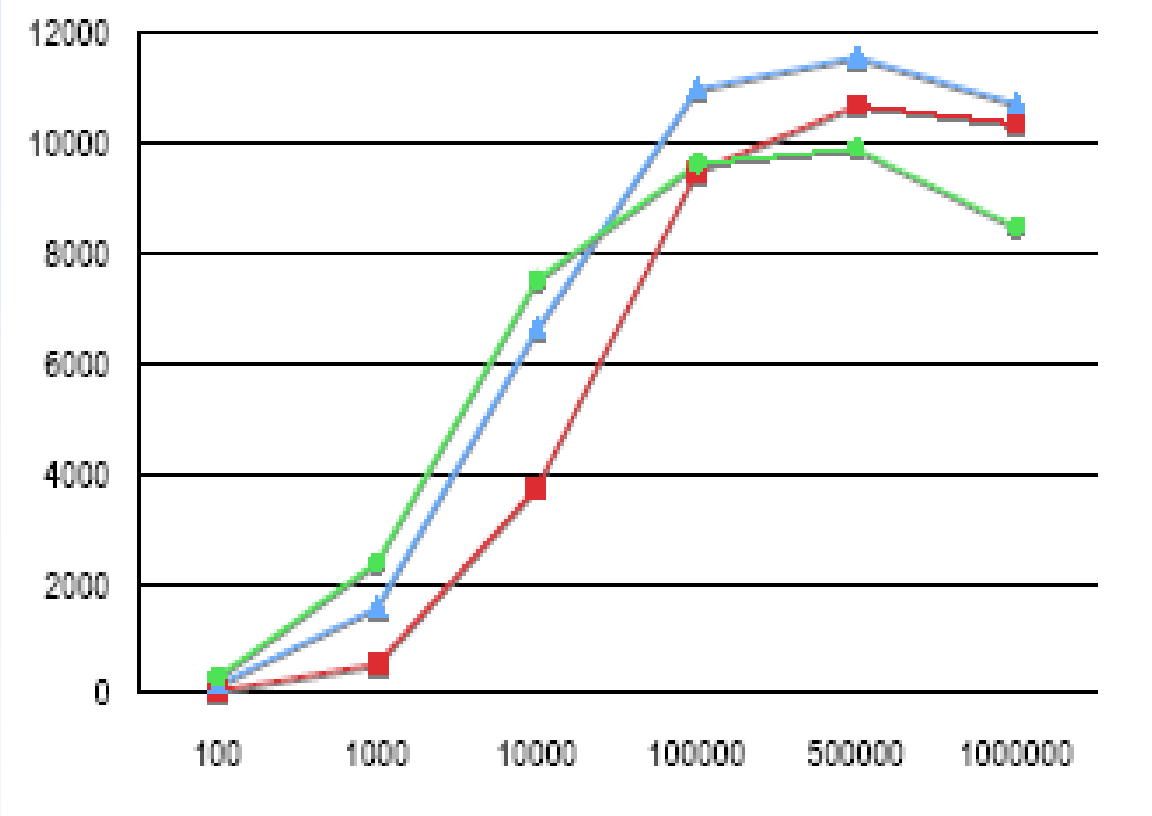
* Full matrix is available at:
http://www-scf.usc.edu/~mattmann/DM-Matrix-090105.doc

Experiments

- Identified that for PDS, the key dimension to currently evaluate was *transfer rate*
- Performed quantifiable measurements of transfer rate
  - LAN/WAN
    - Parallel TCP/IP v. UDP bursting technologies
    - Use FTP/SCP as a baseline

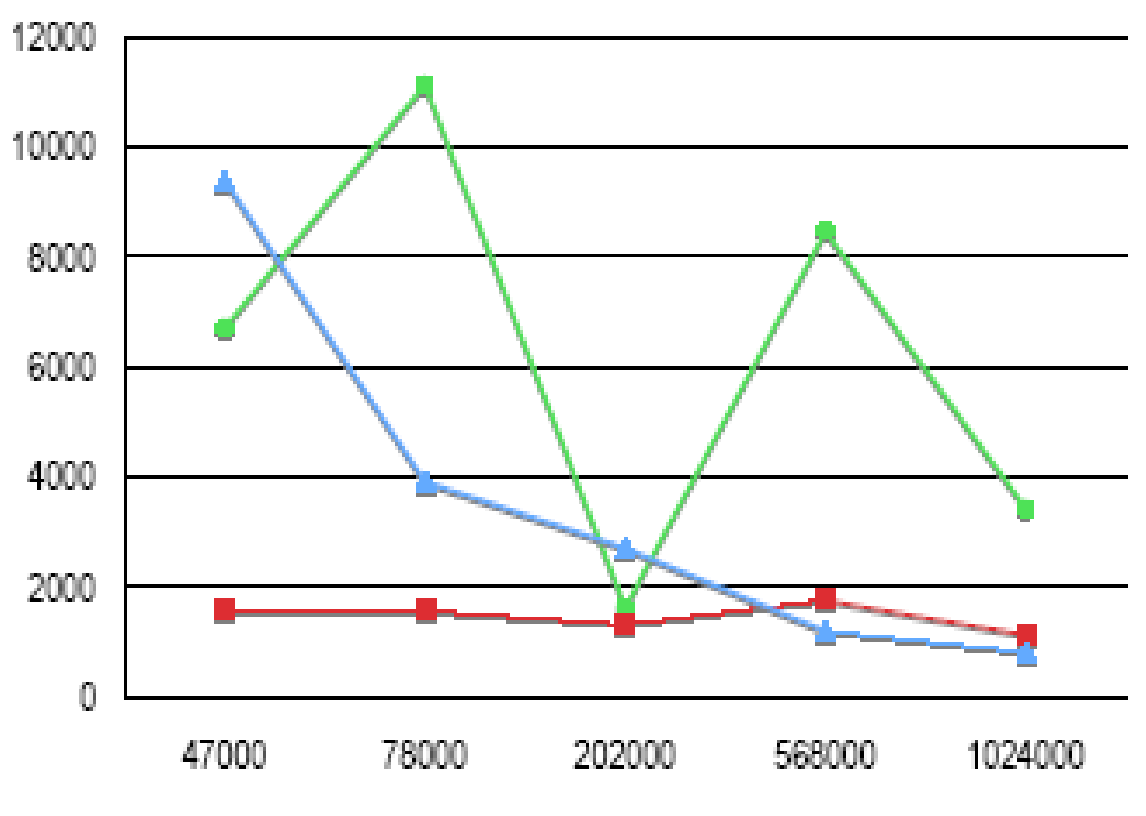
TCP/IP LAN

Transfer rate (Y axis) versus file size (X axis)
GridFTP: blue, bbFTP: red, FTP: green



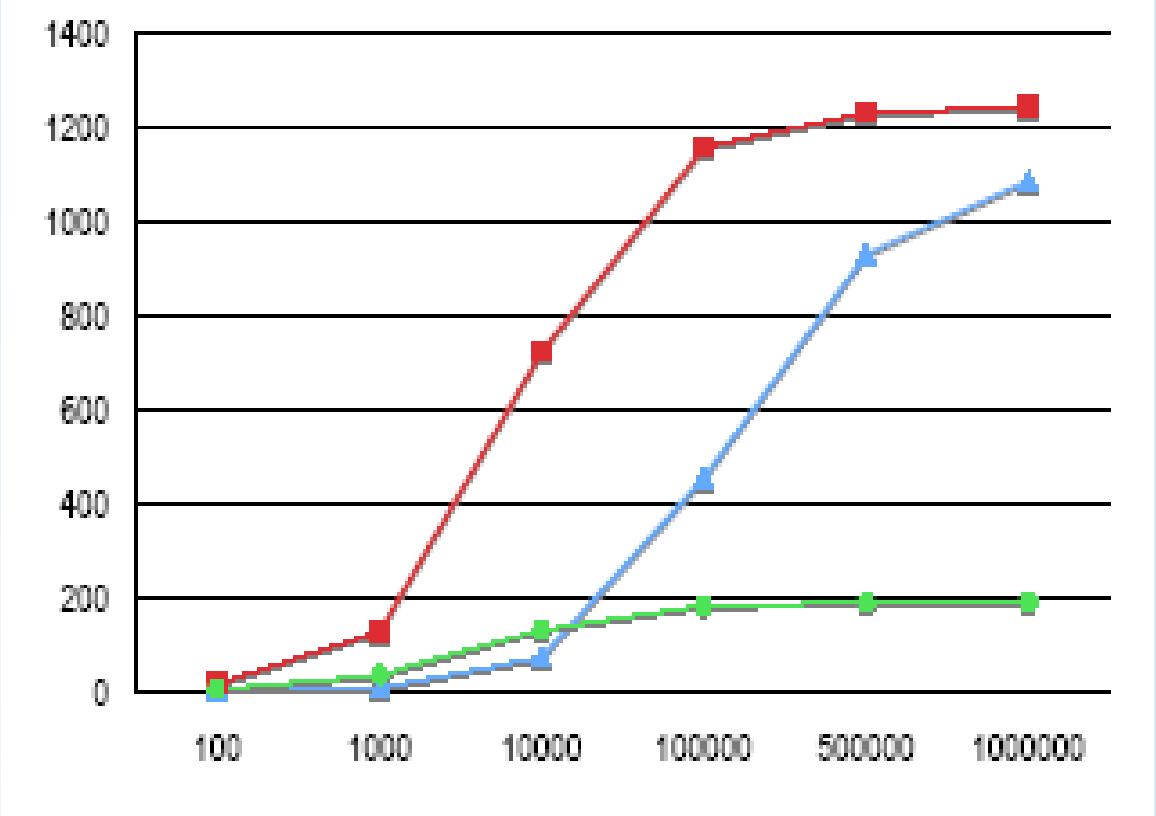
UDP LAN

Transfer rate (Y axis) versus file size (X axis)
UFTP: blue, FTP: green, CUDP: red



TCP/IP WAN

Transfer rate (Y axis) versus file size (X axis)
GridFTP: blue, bbFTP: red, FTP: green



UDP WAN

- Unable to test UDP on the WAN
  - Network configuration and firewall rules difficult to configure
  - Easier for SA's to open up TCP ports than UDP ports
  - JPL not connected to MBONE network (multicast)

Conclusions

- TCP/IP streaming technologies testable on WAN (the real use case for PDS)
  - Easier to configure firewall rules
  - Transfer rates higher on LAN
  - Order of magnitude (12x) improvement over that of FTP
- UDP technologies
  - Promising on LAN, but failed to outperform even basic FTP
  - Not testable on WAN because of difficult firewall rules

Questions?