

Gridftp notes
25 June 2003

At one point, this wg was going to go away –
Gridftp 1.0 doc is in 60 day period –
Please point out problems there, though.

We were going to go away as a wg –

Fermilab folks – wanted to work on new issues,
So rechartered this group –to work on the next version of the protocol doc –

This is essentially where we are at right now –

Igor Mandrichenko, from fermilab, is cochair now.

Want to discuss the issues, and lock down a list of issues want to go after on the protocol
–

Every issue brought up will not make it to new protocol
We are focusing on protocol, not implementation details –

Have a started working list –
Igor will present –

Gridftp improvements –
Igor Mandrechenko, FNAL

Have several implementations of this protocol, including the ones distributed with globus
Discovered several issues with the protocol –

Background/history

- gridftp working group produced gridftp 1.0 protocol draft doc
- wg rechartered
- goal: improve

list of issues/improvements for v2.0 –
goal for today – finalize the list of issues (not including concrete solutions)
after ggf8 submit the list as ggf doc –

web site is at www-isd.fnal.gov/gridftp-wg

Issues: unidirectional transfers in E mode

- data must flow in the same direction as data socket connection initiation
- uploads must be passive, downloads must be active
 - will not work with firewalls, private networks, etc

- reason: possible race condition leading to lost data connections.

Issues: order of pasv/spas and stor/retr commands

- passive mode, server must provide data socket before it knows what to transfer
- inherited from old protocol
- port or pasv are important --

issue: possible disconnection of idle control and data sockets

- some firewalls drop idle tcp connections
- server with cache/staging functionality ...

issue: unreliable EOF communication in stream mode

- per rfc959: closing of data channel signals end of data
- server cannot distinguish between client termination and end of successful upload transfer
- doesn't exist in block or extended block mode –
- if do something with this – has to be a new mode, or this becomes a much larger problem to have stream mode changed

issues: control over server feedback

- gridftp server periodically sends 1xx responses during xfer (perf markers)
- can be used to protect control channel against being timed out by the firewall
- client needs to have control over frequency and contents of the feedback

issues: data protection

- tcp offers transmission error detection with checksums
- request from NASA and Sandia –once in every 4TB of data you will get an unreported error – -- undetected by tcp
- 99% of people will not use – but critical for those who need it --
- may not be sufficient for high volumes of data
- the idea is to introduce additional means of protection against transmission and storage errors
- have a prototypical solution that will share with mailing list –
 - flush whole file to disk, compare signatures on both sides
 - what about incrementally???
 - Can turn on transmission integrity with globusio – so don't need to worry about this as a protocol change
 -

issues: adoption of IETF draft for structured list

- there is IETF proposal for “ls” –like commands with structured output designed to be parsed by the client
 - <ftp://ftp.isi.edu/internet-drafts/draft-ietf-ftpext-mlst-16.txt>
- also the proposal includes the widely accepted and used SIZE and MDTM commands

- the proposal is to adopt the IETF draft as part of GridFTP
- (it is a proposed standard at ietf)

what are our next steps?

- ggf8: finalize the list
- after ggf8: submit the list as a ggf doc
- between ggf 8 and 9 –

there may be other issues – make sure we don't prematurely close the list –

note that with 1.0 – had very strong strawman – was basically pushed through, but not a lot of buy in –
can do it better now –

another issue –

a way to say on client side – to be able to do a transfer using XIO (and something like dynamo, or remote dma, or...

Issue: gridftp perf on lots of small files (packed transfers of large numbers of small files)

This we may have – with eretrieve or estore –

And then have FEAT command to find out what ERET or ESTO is installed –

ERET module = “opaque string”

Then lookup what this means –

- often it is more efficient to send many files as a single tar file
- this may have been provisioned by ESTO/ERET

another issue – common logging –

does this have anything to do with the protocol –

go to bugzilla and add a bug severity = enhancement –

chuck severence – wants to isolate data channel mechanisms –

will be adding 2nd control channel – which is soap –

because of new implementation – can rip out the control channel –

new implementation beta –

moving data within a week –

expect to go beta with basic functionality – august

and November – striped server beta

march/april – actual release –

new server speaking 1.0 or 2.0, depending on how this group is done

after this is released – turn attention to ogsa gridftp

year from now – new prototype –
of ogsa gridftp

another request –
when negotiating striping –
if cannot handle stripe – will go into parallel – before falling to single –

issue – “intelligent data channel provisioning”
if have a striped server, if can’t get all the stripes out—rather than falling back into
stream mode, then fall into e mode with parallel connections...

so – try calling this – Flexible striping

- a knob you can tweak for performance
 - but concept of failing down is one not thought about
- should be a way to dynamically negotiate and flexibly choose striping strategy
- may have an impact on the unidirectional xfer in Emode

so have the list –
need to turn into a document –