Introducing *Saratoga*

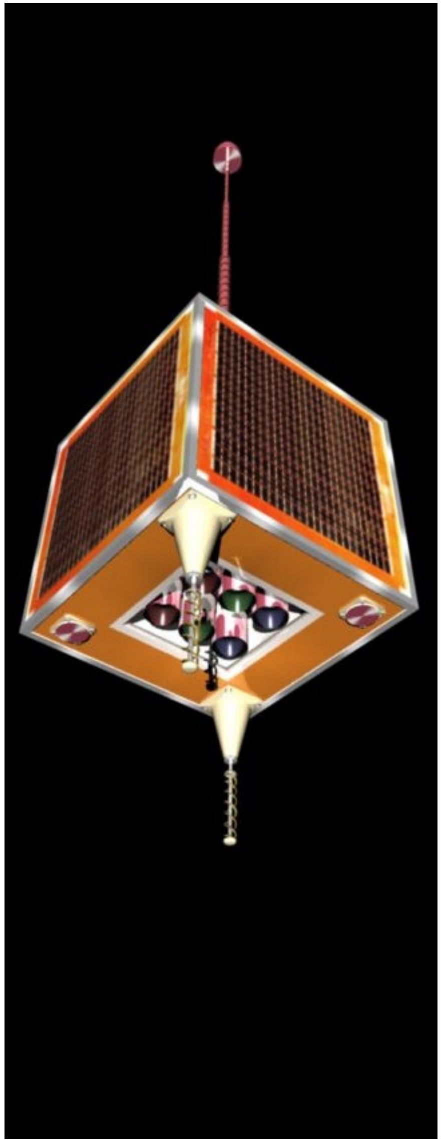
fast file transfer from space

Lloyd Wood
space initiatives manager
Global Government Solutions Group
Cisco Systems

Delay-Tolerant Networking workshop
Trinity College, Dublin, 22 May 2007

Short summary

- *Saratoga* is a simple file transfer protocol that can also be used to transfer DTN bundles.
- Developed by Surrey Satellite Technology Ltd (SSTL) to transfer remote-sensing imagery from its IP-based LEO satellites to ground.
- NASA Glenn has cleaned up the *Saratoga* design to create a new version of *Saratoga* for file or bundle transfers. We have sent an internet-draft describing this to the IETF.
- We already have multiple implementations (in Perl, Python, and C, on Linux and RTEMS).
- Using the testbed first used for the CLEO router in orbit, we are preparing to fly the RTEMS code on the UK-DMC satellite.



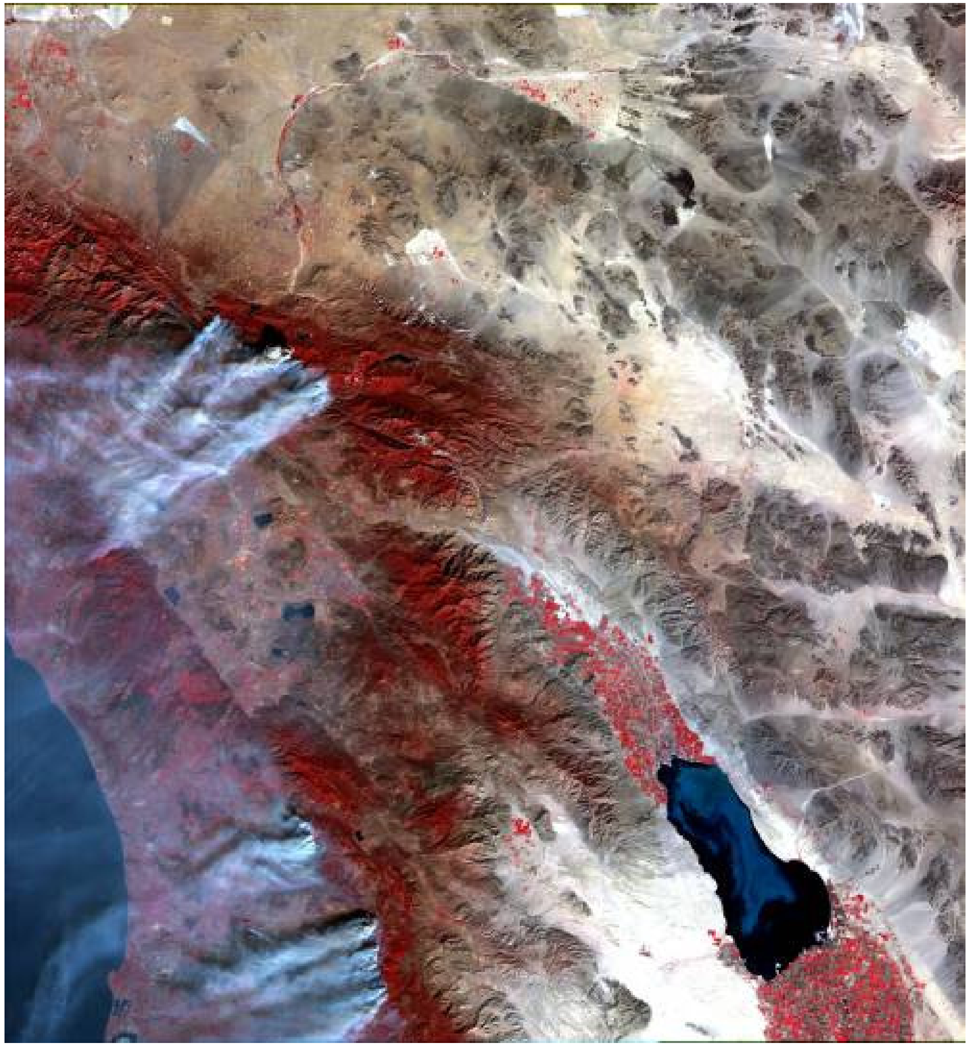
Disaster Monitoring Constellation (DMC)

Surrey Satellite Technology Ltd (SSTL) build and help operate an international constellation of small sensor satellites.

The satellites share a sun-synchronous orbital plane for rapid daily large-area imaging (640km swath width with 32m resolution). Can observe effects of natural disasters. Imaged the effects of Hurricane Katrina and the Indian Ocean Tsunami.

Government co-operation: Algeria, Nigeria, Turkey, United Kingdom, and China. Each government finances a ground station in its country and a satellite. Ground stations are networked together. Three more satellites have been announced and are being built.

www.dmcii.com



fires in California, 28 October 2003 (UK-DMC)

DMC can image anywhere on Earth

Palm Island Resort, Dubai, 14 Dec 2003 (UK-DMC)

Three Gorges Dam, China, July 2004 (UK-DMC)

www.dmcii.com

DMC in use: after Hurricane Katrina, 2005

In this false-color image, dry land is red. Flooded and damaged land is shown as brown.

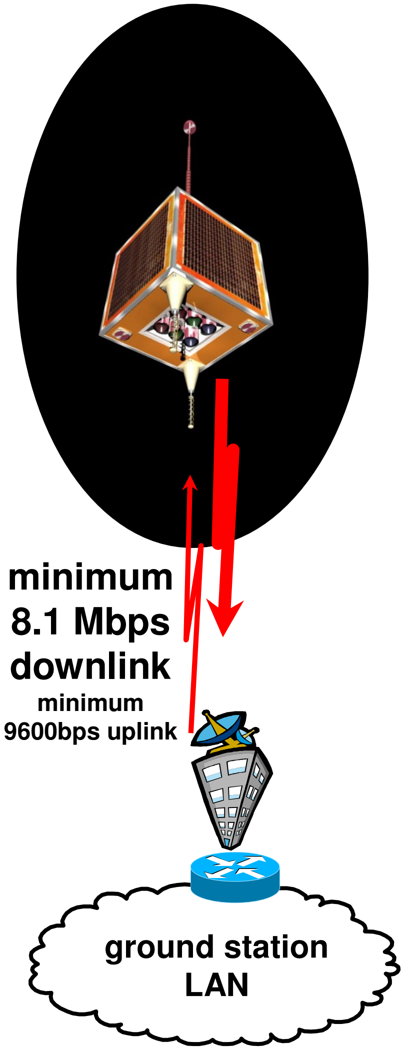
Small part of an image taken by the Nigerian DMC satellite on Friday 2 September, for the US Geological Survey.

DMC is working as part of the United Nations International Charter for Space and Major Disasters.

Imagery delivered by using Internet Protocol.

www.dmcii.com

How are all these images downloaded?



Satellite: each DMC satellite has multiple onboard computers. For housekeeping (the On Board Computer, OBC), for image capture and packetised transmission (the Solid State Data Recorders, SSDRs), for redundancy and survival. Interconnected by IP over 8.1Mbps serial links for data and slower CANbus for backup control. Each satellite is a custom-built local area network (LAN).

Newer satellites also have 20/40 Mbps X-band downlinks for added hi-res cameras; faster downlinks (100+ Mbps) are planned for future missions. Uplink is only 9600bps for command and control. Uplink speeds are also likely to increase… to 38400 bps. *Very* asymmetric; 850:1 or worse downlink/uplink ratio.

As much data as possible must be transferred during a pass over a ground station. Passes may be up to twelve minutes, depending on elevation. At 8Mbps, that's approximately 650MB of useful data (about a CD-ROM's worth) that can be transferred in a high pass – if you fill the downlink with back-to-back packets at line rate. Link utilization *really matters*.

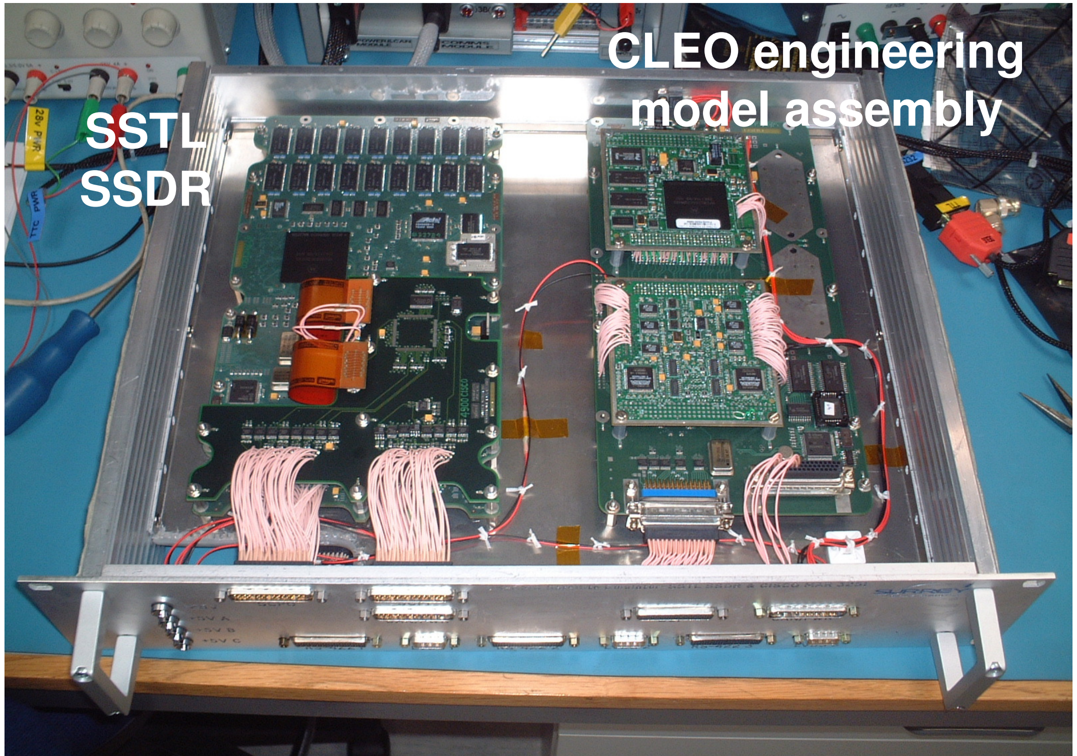
Ground-based testbed for development

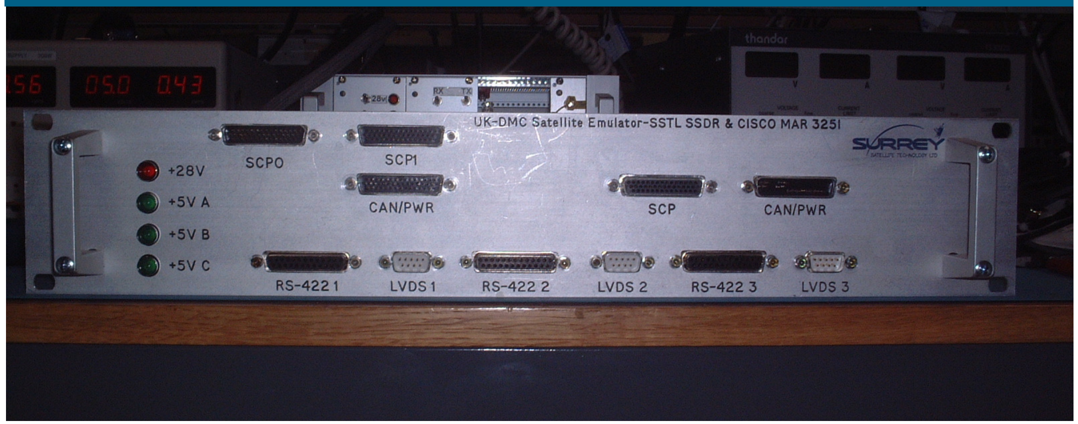
NASA Glenn needed to gain familiarity with operating and configuring CLEO router with SSTL's onboard computers.

Ground-based testbed allowed configuration changes to be tested on the ground at leisure before being made to CLEO router in orbit during a ten-minute pass over a ground station.

Built rack-mounted ground-based testbed ('flatsat') using SSDR and engineering model of mobile router, and networked it from NASA Glenn in Ohio, so NASA could get familiar with SSDR design and use.

Now using testbed in development role for flying *Saratoga* and DTN code on UK-DMC satellite.





Networking constraints

- Packet loss is caused by channel-induced errors, most often at the start/end of a pass.
- There is no congestion due to coarse-grained scheduling and limited multiplexing; file transfers are sequential, not parallel. Transfers go one hop.
- High asymmetry and large files preclude use of TCP. TCP ack congestion of backchannel is the main problem, even before we get into the inability of TCP to fill a pipe or handle loss well.
- *Saratoga* was designed in 2003 (before LTP) to replace an SSTL-written implementation of CFDP that was first flown on AlSat-1 in November 2002. That CFDP couldn't fill the pipe; *Saratoga* can.

Basic *Saratoga* design

- Flood data packets out as fast as you can. No specified congestion control, since you're only going one hop. Any multiplexing of flows is done by the *Saratoga* peer.
- Every so often, ask for an acknowledgement from the file receiver. Receiver can also send acks if it thinks it needs to, or to start/restart/finish transfer.
- Acks are Selective Negative Acknowledgements (SNACKs) indicating left edge and any gaps to fill with resent data (and with enough information so that sender congestion control could be added).
- That's it. But just how big is a file/bundle?

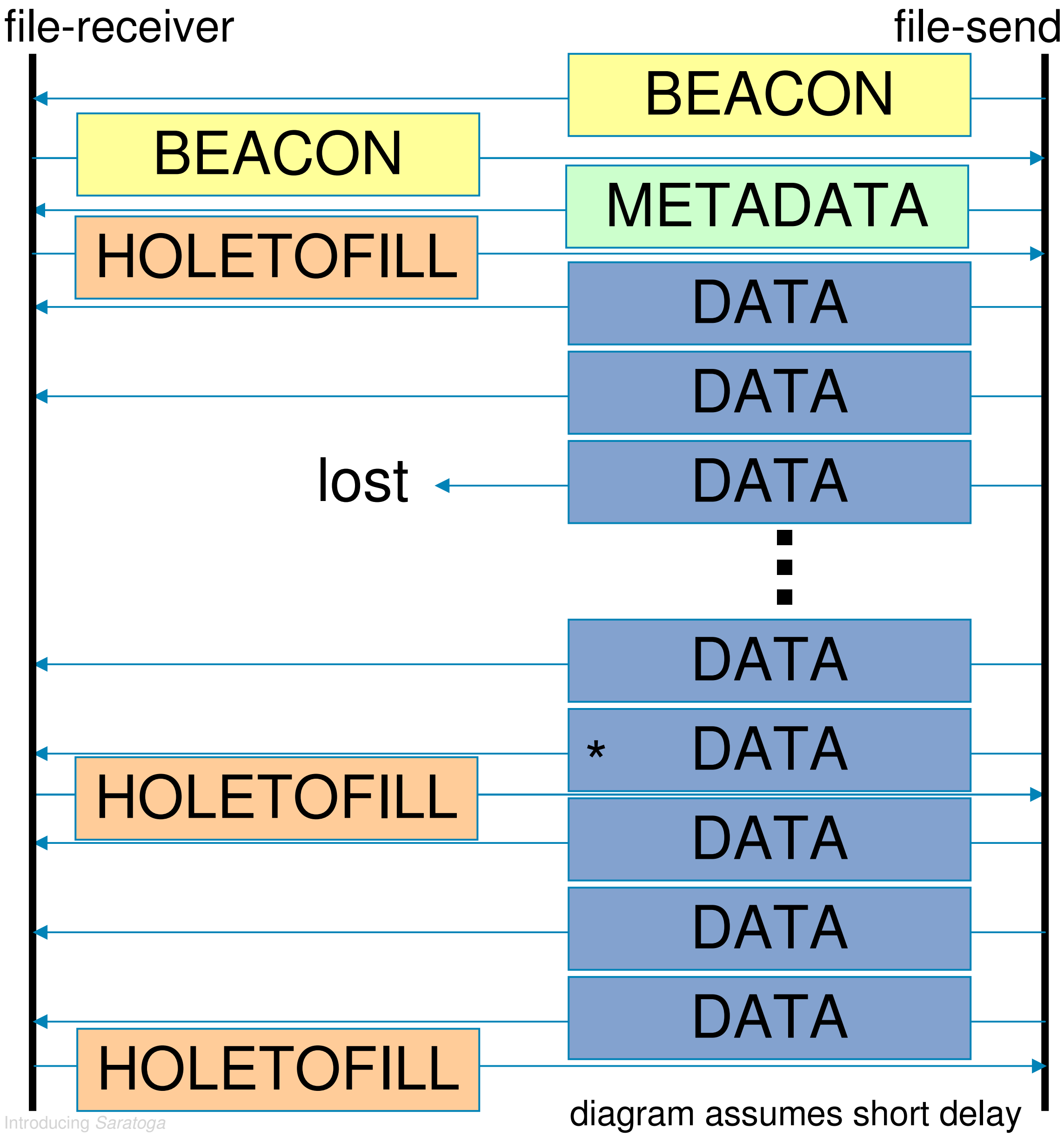
Filesizes

- For the DMC, imaging files are large – typically up to a few gigabytes at 32m resolution; larger for newer cameras. So we think bundles will also be large.
- But ad-hoc/sensor nets also need to transfer small files/bundles; guessing a range limits use.
- So we allow a range of file-descriptor pointers to be advertised: 16/32/64/128-bit file descriptors.
- If file is less than 64KiB, use 16-bit offsets. If file is larger but less than 4GiB, use 32-bit offsets…
- 16-bit is *always* supported. Others are optional.

Saratoga packets

BEACON	Sent periodically. Describes the *Saratoga* peer: Identity (e.g. EID) capability/desire to send/receive packets. max. file descriptor handled (16/32/64/128-bit).
REQUEST	Asks for a file via 'get', directory listings, deletes.
METADATA	Sent at start of transaction. Describes the file/bundle: identity for transaction file name/details, including size. descriptor size to be used for this file (one of 16/32/64/128-bit pointer sizes.)
DATA	Uses descriptor of chosen size to indicate offset for data segment. May request an ack.
HOLETOFILL	Ack. Can use the descriptor size to indicate offsets for missing 'holes' in data.

Saratoga transactions: 'put'



Beacons (optional); show filesize capabilities.

Describes chosen file.

Ack indicates reception.

Lost data creates hole in copy at receiver.

Ack requested and sent describing hole to be filled.

Empty ack indicates transaction is complete.

Saratoga transactions: 'get'

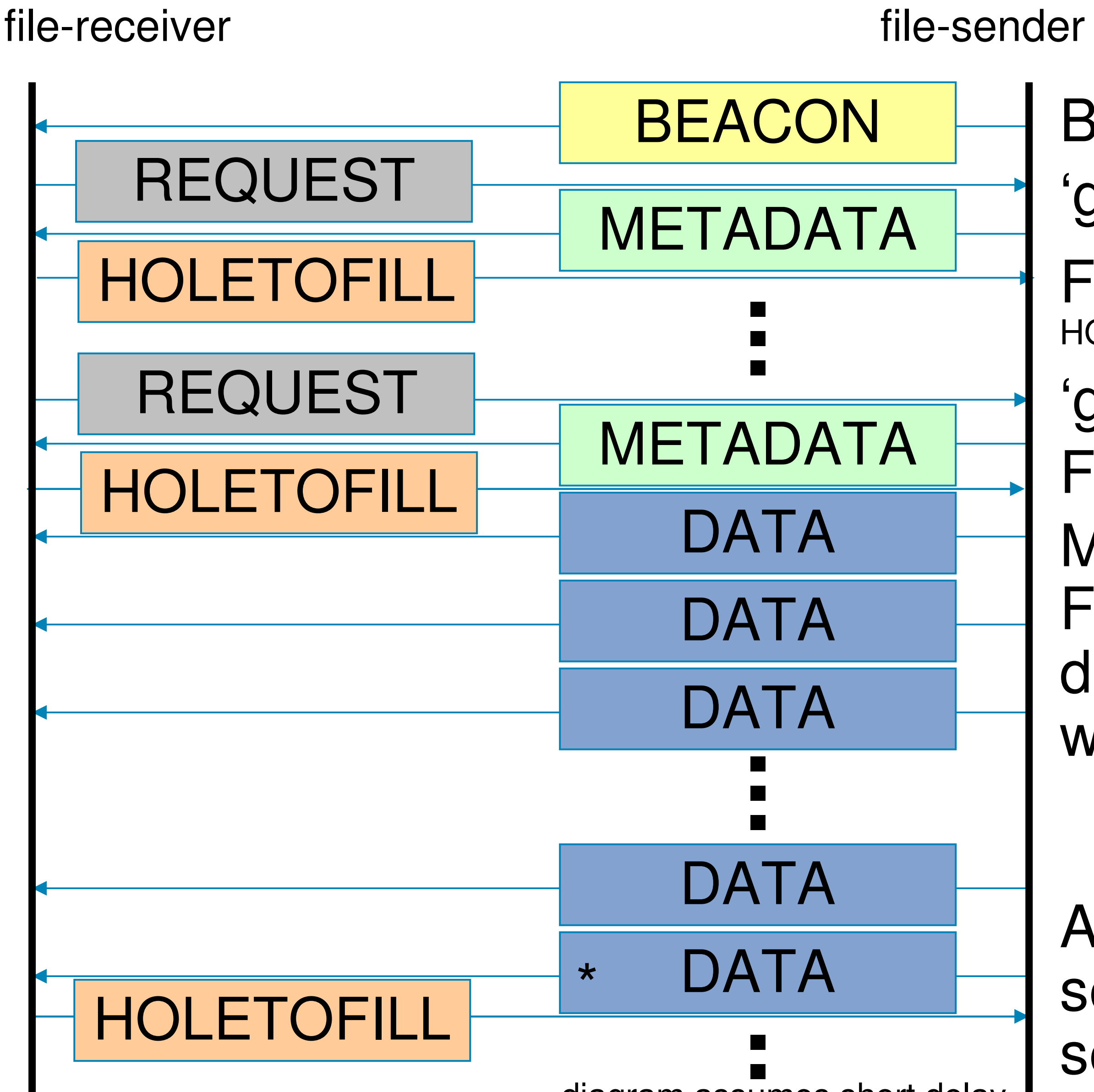


diagram assumes short delay

Beacon heard (optional).
'getdir' can request file list.

File list sent as file…
HOLETOFILL/DATA transaction omitted

'get' requests a file.

File is described.

METADATA is acked.
File data is streamed out directly after METADATA, without waiting for ack.

Ack requested and sent. Sender continues to send DATA.

Why *Saratoga* instead of TCP?

- Link utilization and throughput.
- Assumptions about loss/congestion.
- Coping with link asymmetry.
- Simplicity. TCP is really for a conversation between two hosts; needs a lot of code on top to make it transfer files. We're just interested in moving files; makes e.g. sequence nos. simpler.
- Long delay use – eventually TCP will fail to open a connection because its SYN/ACK exchange won't complete. TCP has many unwanted timers.

Why *Saratoga* instead of LTP?

- Similar design constraints, e.g. using UDP rather than direct over IP for portability, ease of UDP port number *vs.* getting an IP protocol number.
- *Saratoga* was developed before LTP was specified, but advantages in using *Saratoga* are pretty much as they are when compared to CFDP:

  - simpler (draft is half the size, few options).

  - faster; intended for line rate across a single hop.

  - easy to implement; headers fit into C structs, no SDNV.
- *Saratoga* may also be more generally useful for p2p file delivery. Broadcast? Multicast? Possible.

Licklider LTP vs *Saratoga*

Feature	*Licklider*	*Saratoga*
large object transfers	yes (SDNV)	yes (descriptors)
works under high latency	yes	yes
handles asymmetry	yes	**yes, very well**
supports ‘push’ forwarding transfers	yes	yes
directory listings for selection/pull	no	**yes**
includes object metadata	no	**yes** (optional)
supports delivery of corrupted data	sort-of (why?)	**in -01** (UDP-Lite)
object integrity checksums	no	**yes** (optional MD5)
beacons for discovery and automated transfers	no	**yes** (optional)
multicast/broadcast delivery	no	**for -01?**
name requires explanation	yes	yes

Why *Saratoga* as a name?

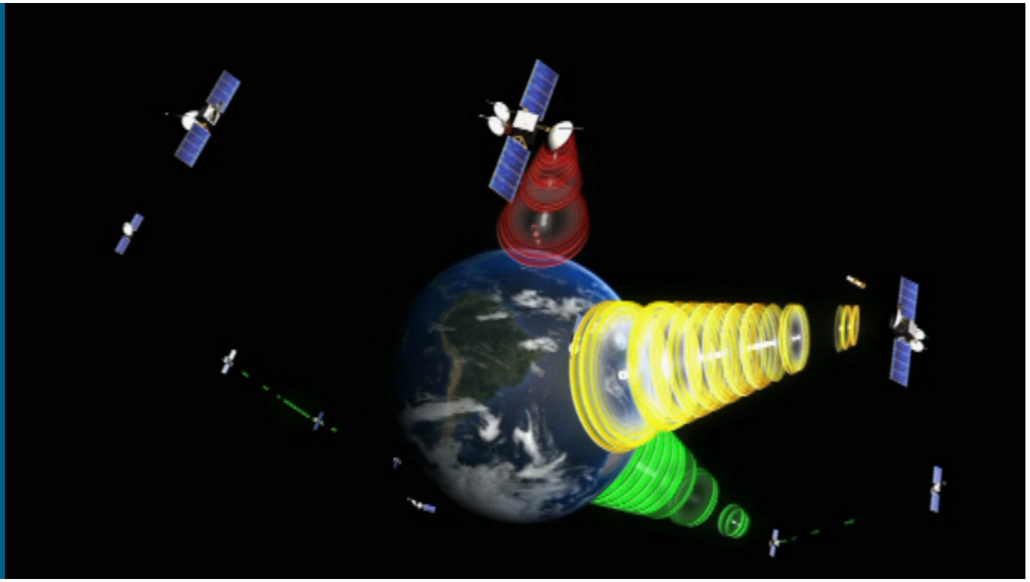
Photo # 19-N-84312 USS Saratoga underway in Puget Sound, 15 May 1945

USS Saratoga **(CV-3) is sunk off Bikini atoll.**
Chris Jackson of SSTL dives there.

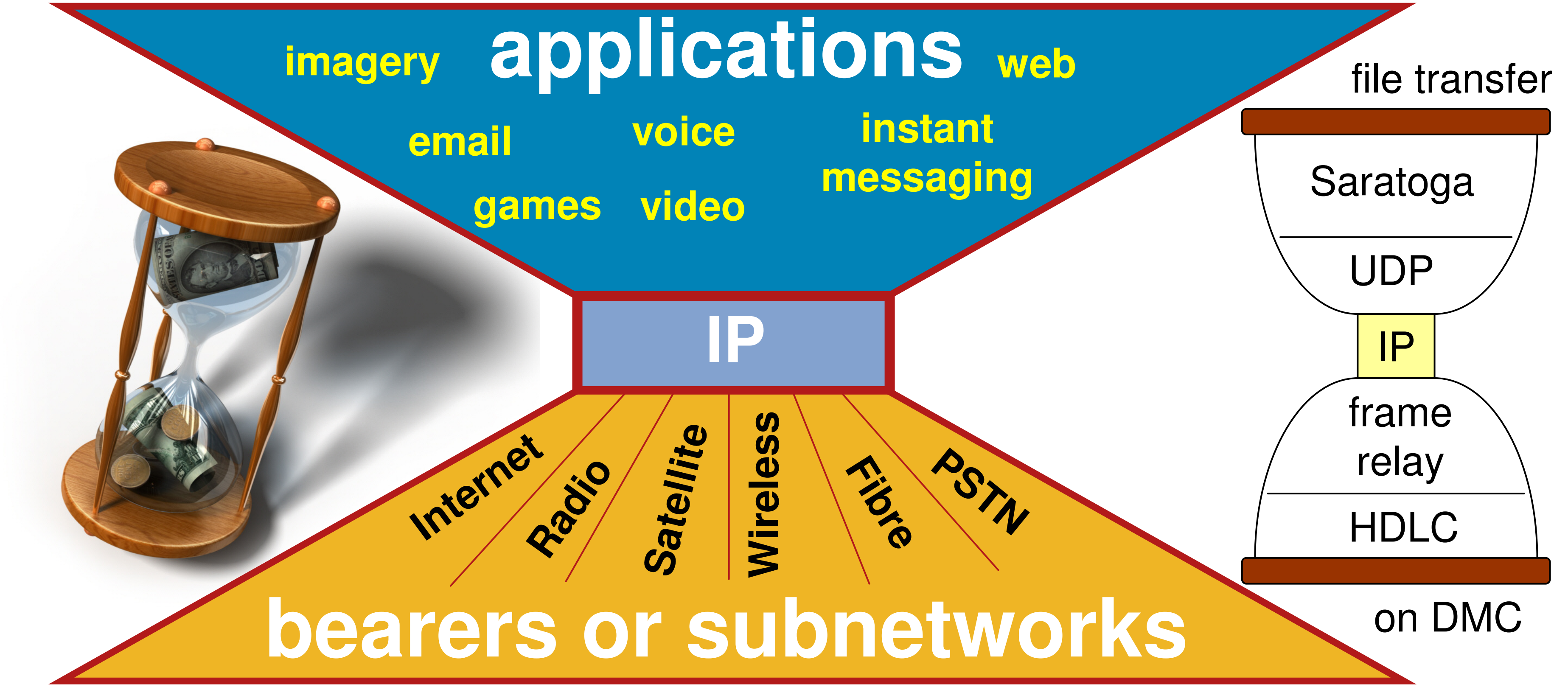
Why are we pursuing DTN with DMC?

- Because the DMC is an example of using IP both on the ground and in space, with the ground station acting as a gateway between types of use.
- Assumptions governing IP use (link use, shared contention *vs* dedicated scheduling models) differ between ground/space, but the protocol used remains the same.
- DMC can be seen as a prototypical DTN scenario, with long delays between passes over ground stations.
- It's a starting point for ground ad-hoc sensor networks, too.

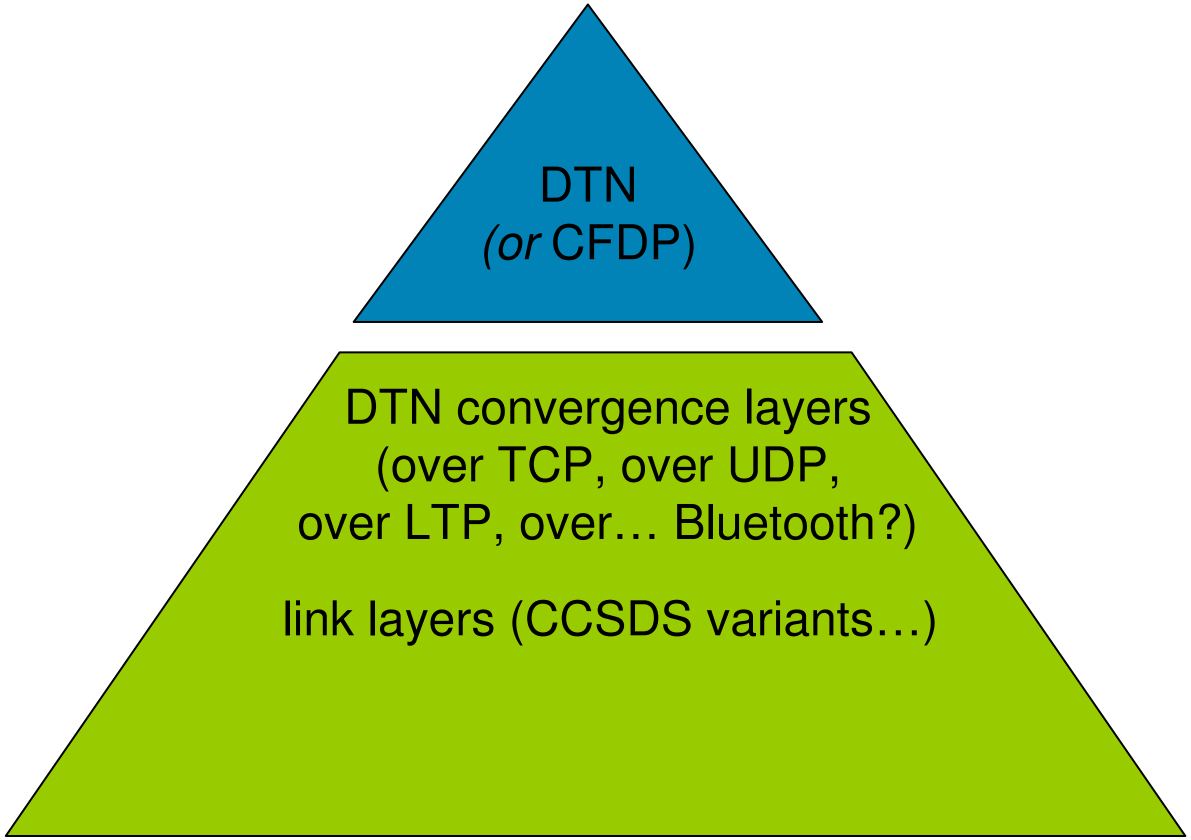
Wider thoughts



IP – ‘the waist in the hourglass’



DTN – ‘the eye of the pyramid’?



DTN and CFDP are adapted to run over many things.That makes them complex.

…or is this just the bottom half of a really big hourglass? what applications use CFDP/DTN?

DTN – ‘the eye of the pyramid’?

…or is this just the bottom
half of a really big hourglass?
what applications use CFDP/DTN?

NASA's networking use – two types

- Manned missions – 'the human network'.

  Humans are unpredictable traffic sources. Many communications happen at once to support them; arbitration/fairness is required between traffic flows competing to share the link. That's what routers do, and why there are routers on the International Space Station.

  Mix of TCP and UDP traffic. (Email/DNS/BGP need TCP). DTN/file transfers become background traffic.

- Unmanned missions – the deep space network.

  Payloads can be scheduled – 'offload your data, shut up and shut down', before switching to the next payload. Coarse-grained scheduling of links, dedicated to one payload at a time without competition, avoids the need for a router to arbitrate competition for the link.

  UDP traffic. TCP and its assumptions don't fit deep space.

Just how far can IP go?

- IP isn't limited to geostationary orbit (although some protocols using IP suffer badly with distance, most famously TCP).
- IP itself is addressing only. No timers, no delay assumptions (though TTL imposes limit on hops).
- Protocols with timers are based on assumptions about their use. Are the assumptions valid for the long delays of deep space?
- UDP via a static route from Pluto? Why not? With the right simple rate-based protocol over UDP, it *will* work. (TCP won't.)
- DTN, like Internet, can be IP-based; it's just how IP is *used* that is altered.

with thanks to
Will Ivancic, Wes Eddy, Jim McKim
and Chris Jackson

background reading:

http://www.cisco.com/go/space

ftp://ftp-eng.cisco.com/lwood/cleo/README.html

Questions?

thankyou