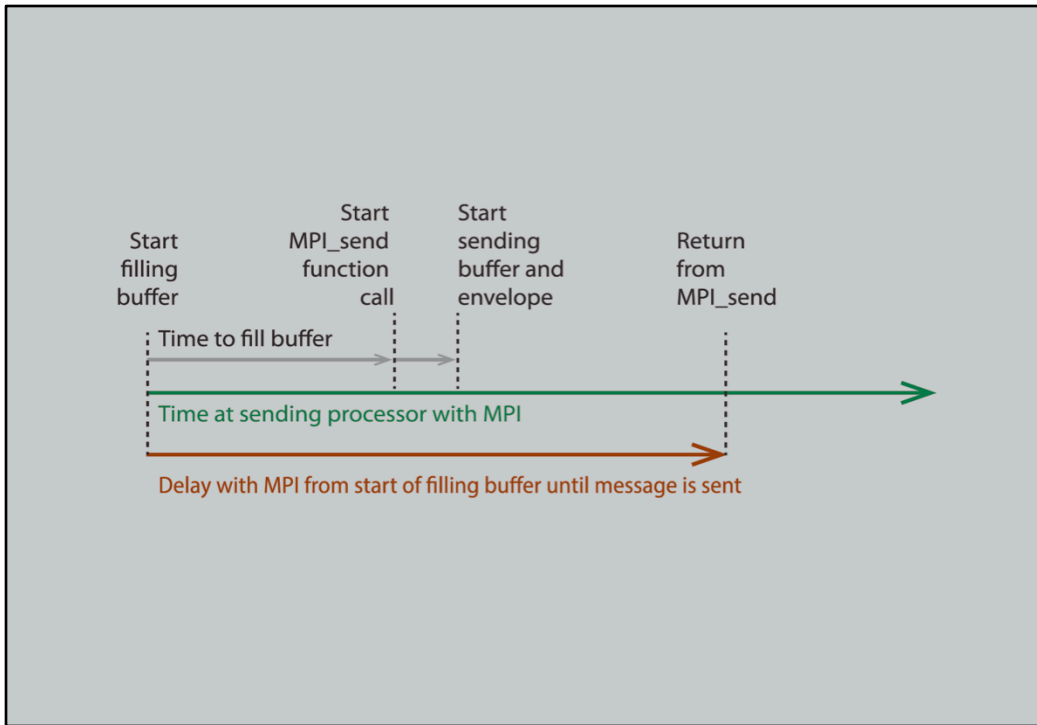
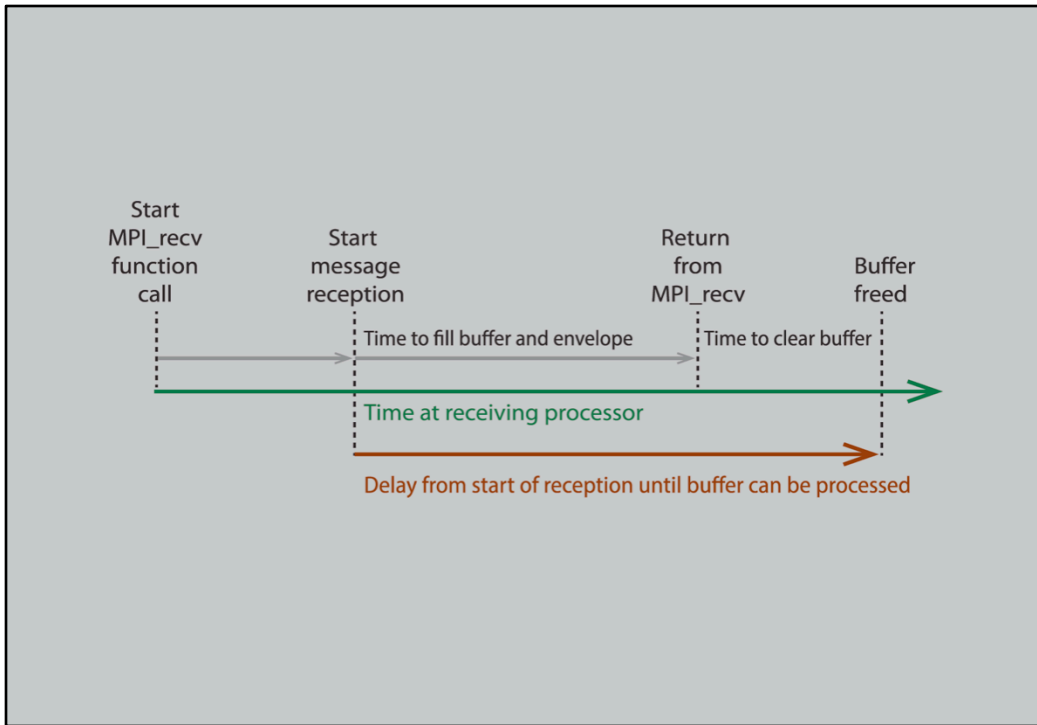


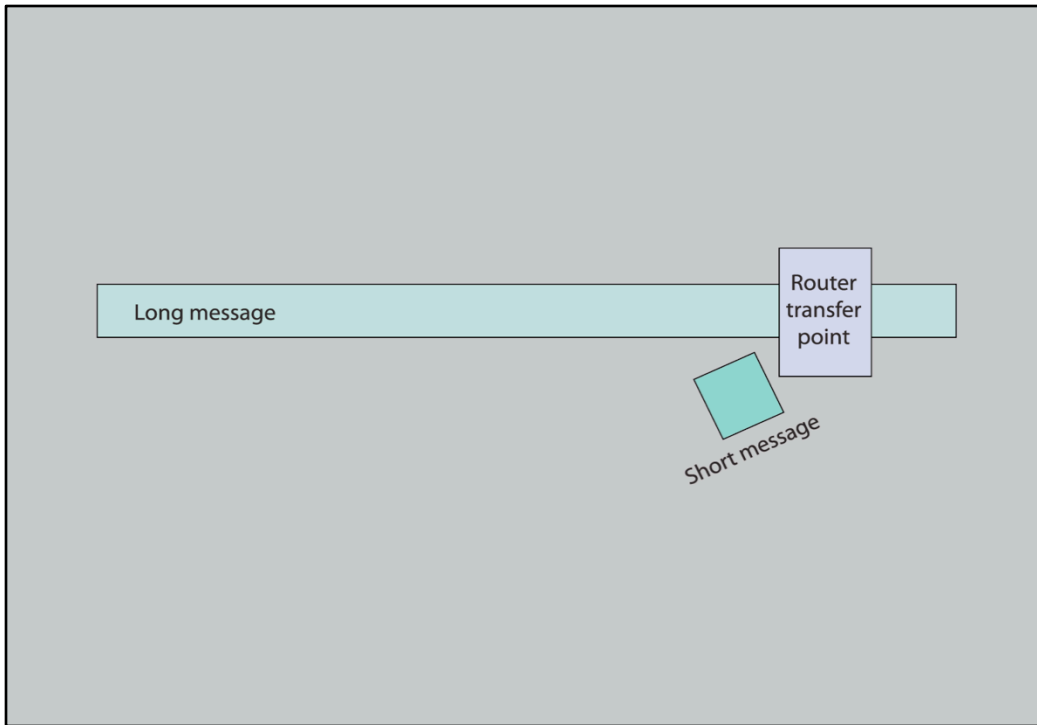
Good afternoon, I am Earle Jennings.
This presentation introduces the STAR Messaging Protocol.
First some problems with MPI are outlined,
and then this protocol is discussed,
which solves these problems and is optimized for Exascale and beyond systems.



Sending an MPI message locks up a buffer, until the message held in that buffer, completes transmission.



Receiving an MPI message, not only locks up a buffer to receive it, but also locks the buffer as long as it takes for the data to be moved elsewhere, or processed in place.



A long MPI message can stall a short message at any and all router transfer points.
There is more bad news:
Exascale systems add fault resilience challenges to communications.
Also, data centers are plagued with malicious software attacks, often entering through their communication portals.
And all of these need to be addressed below the MPI layer.



The STAR message protocol, transmits and receives a STAR message, on every local clock cycle,

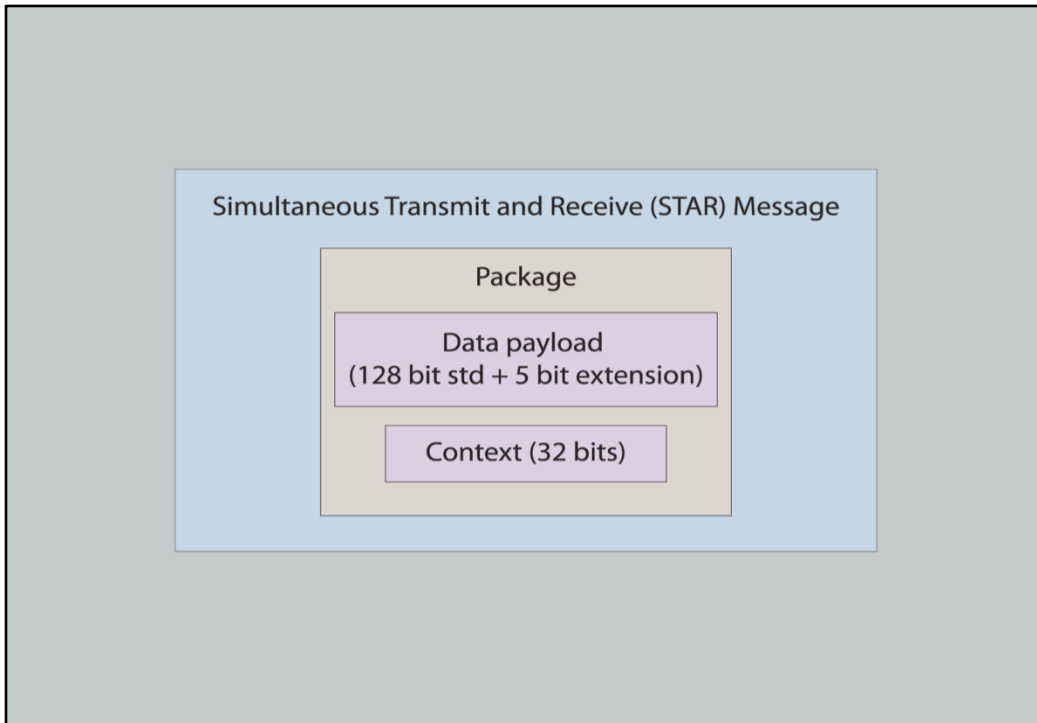
except when responding to an uncorrectable error on reception.

The response to such errors can involve

automatic, channel component replacement

within, at most, a microsecond.

Each STAR message traverses all communication pipes in one clock cycle, removing these 3 MPI problems from the HPC system.

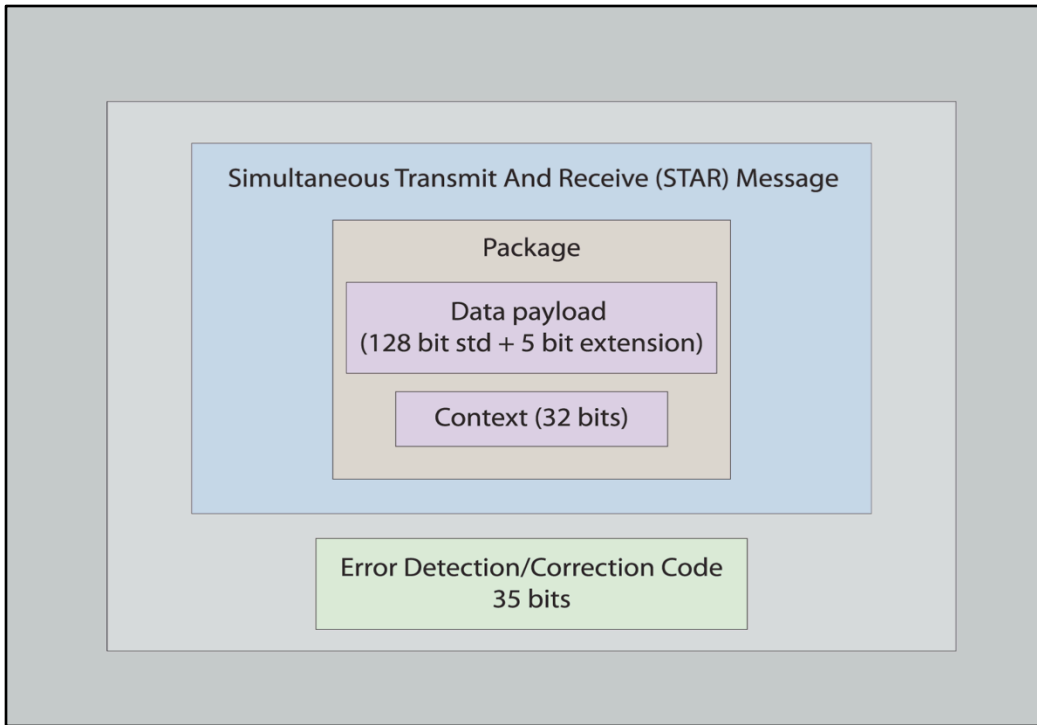


This shows the application layer of each STAR message

The context of the message

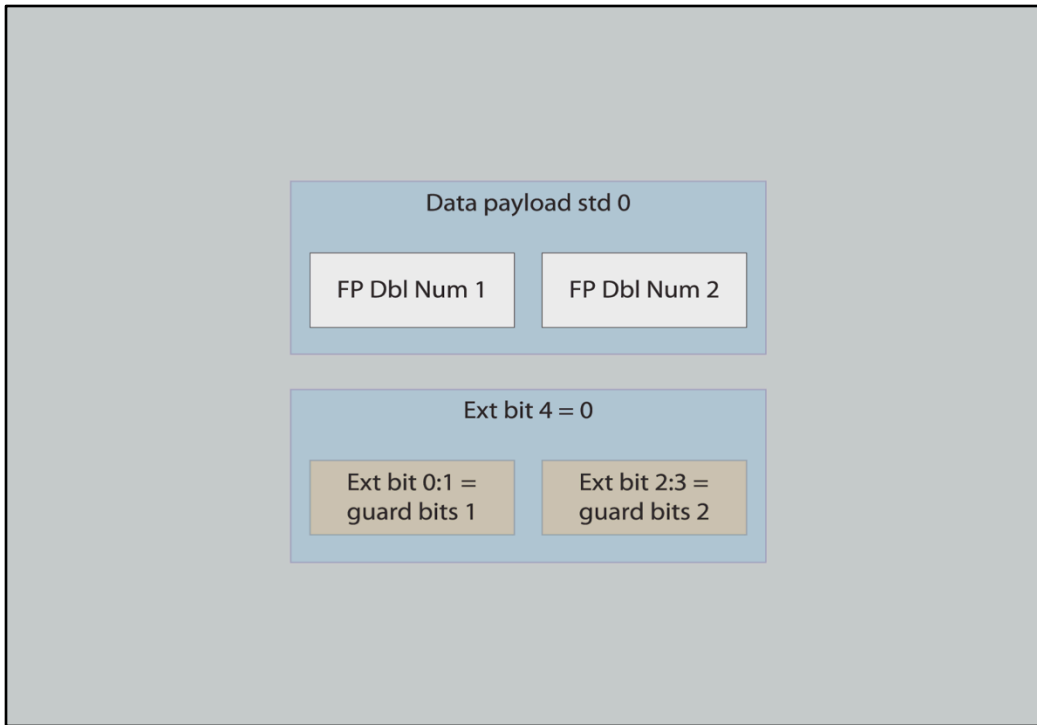
is interpreted at every STAR message core to determine its disposition and transfer.

The context, and its interpretation, **is under complete control of the program.**

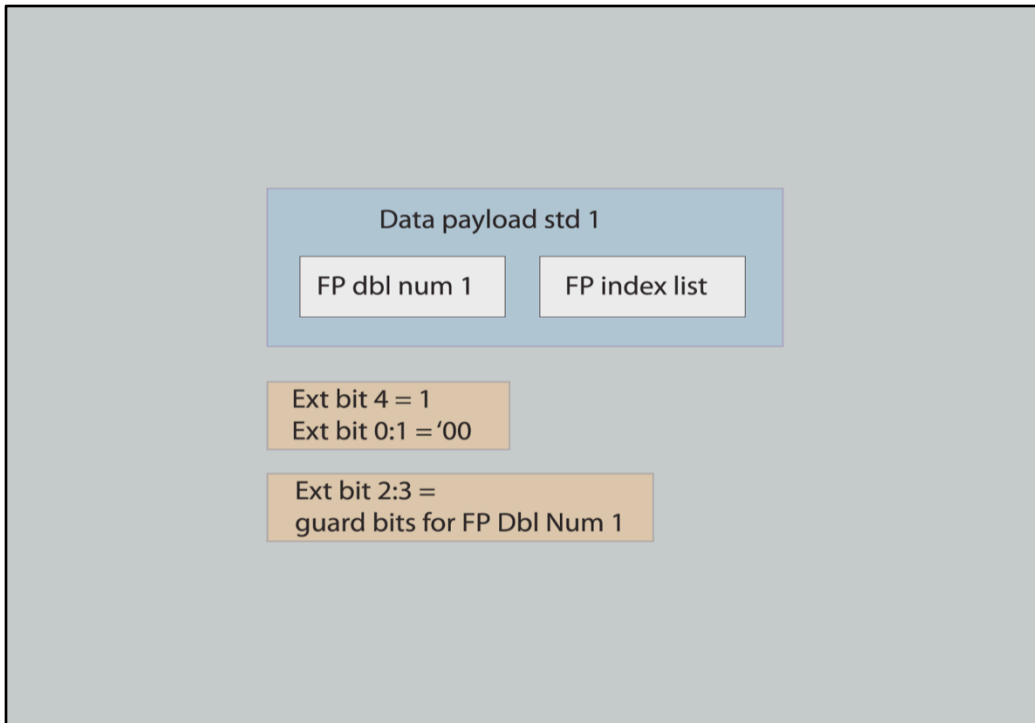


This shows the transport level, of each message, particularly between chips. The EDC code can support 1 bit corrections and 2 bit error detection on each 33 bits of the package.

Assuming a 1 ns local clock at each communicating node, 200 Gbits/second can be delivered, and sent, on each STAR channel.



The data payload may be configured by the 5 bit extension code in many ways. Consider a first payload of two double precision numbers, each with two guard bits. This supports complex number arithmetic, as well as bulk downloading and uploading from the Data Processor Chips, to save and restore the state of its cores.

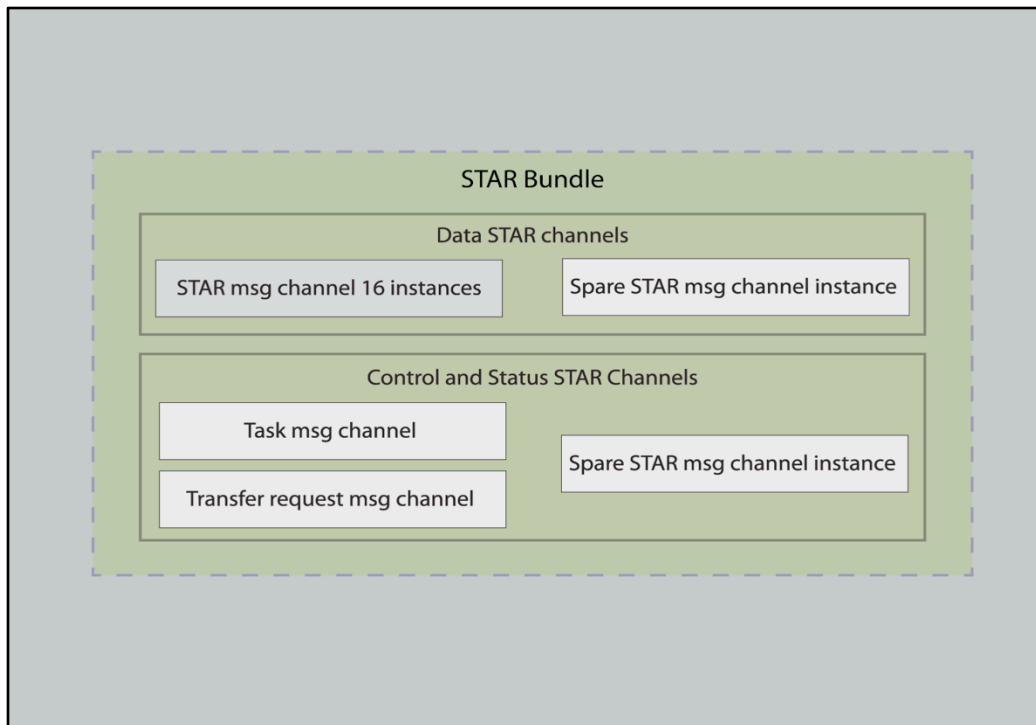


Here is a second payload, configured as a double precision number and an index list. The numeric data, of up to 66 bits, can be arranged to support compressed, vector transfers. . . .

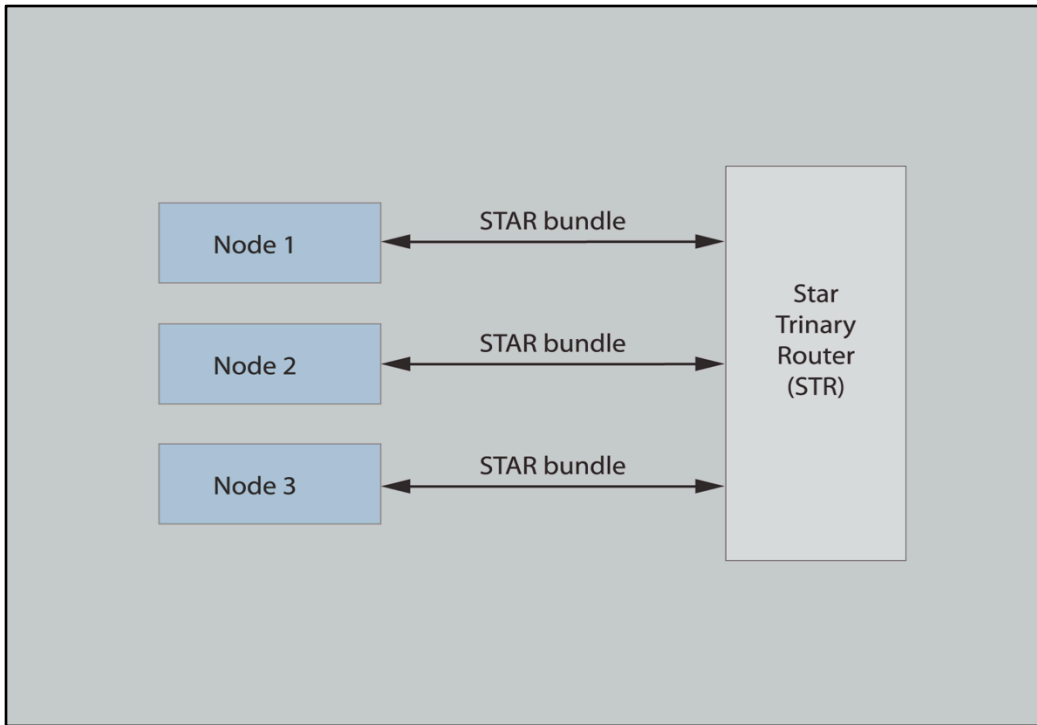
These messages, can each support from 1 to 6 numeric components, using Gustafson's posit notation as well as Floating Point numbers.

Initially, 64 bits will be used to support three modes, 16, 32 and 64 bits, for vector dot products of 4, 2 and 1 components, respectively.

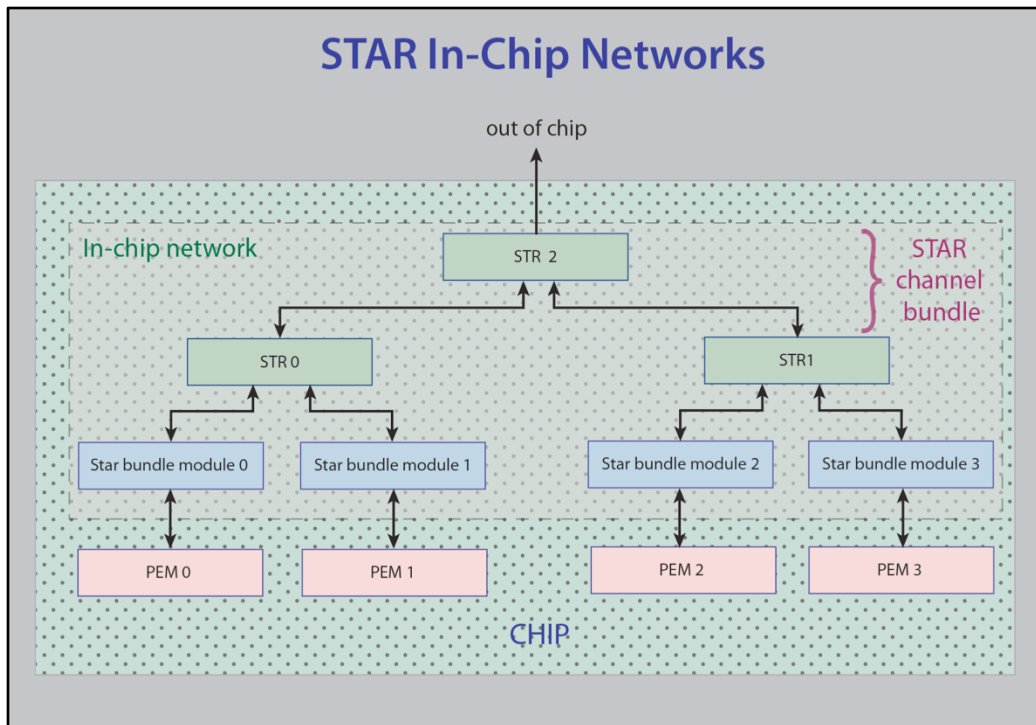
As an application progresses, it can use fewer parameters per payload, for greater precision.



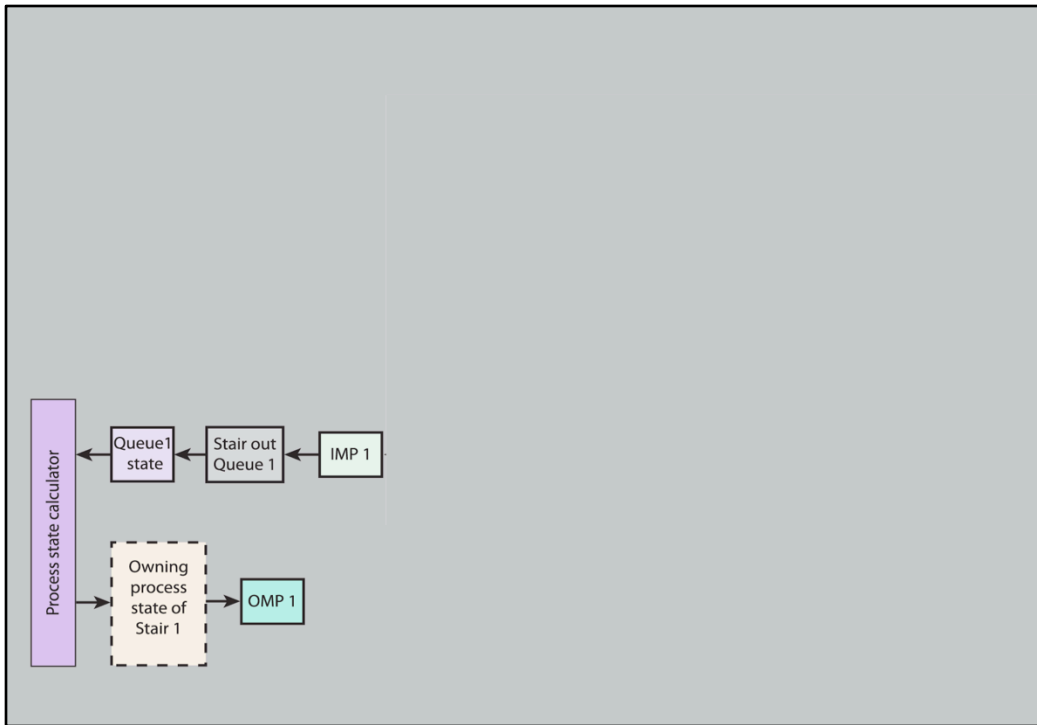
A STAR bundle, includes data, as well as, control and status channels. The data channels include 16 instances for data transfers, and a spare instance, used for fault resilience among these data channels. The control and status channels include a task message, a transfer request, and a spare fault resilience instance. The transfer request allows the DPCs to request DRAM accesses without calculating each address themselves. This will save at least 15% of the compute energy in the DPCs.



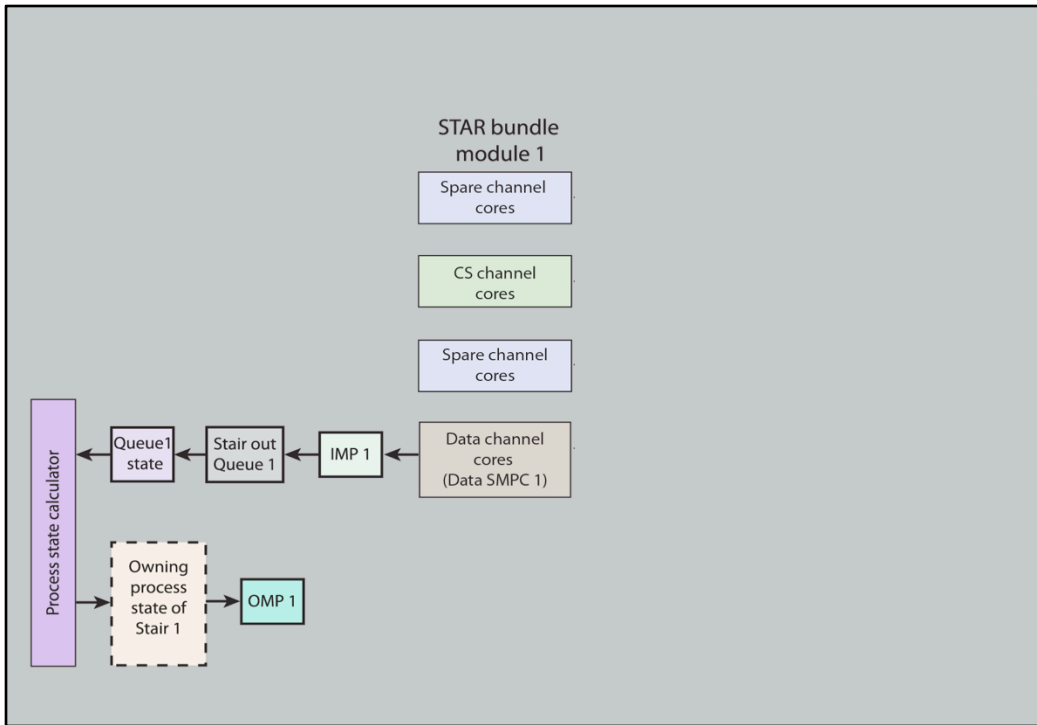
A STAR communications network is a point-to-point network of nodes, acting as sources and destinations, and routing nodes, called STAR Trinary Routers. Each STR has three bidirectional STAR bundle links, which may be to nodes or other STR instances. Each node can be a module within a chip, or a chip. The STRs, inside a chip, are laid out as a communication module, paired with a module of cores, as a unit.



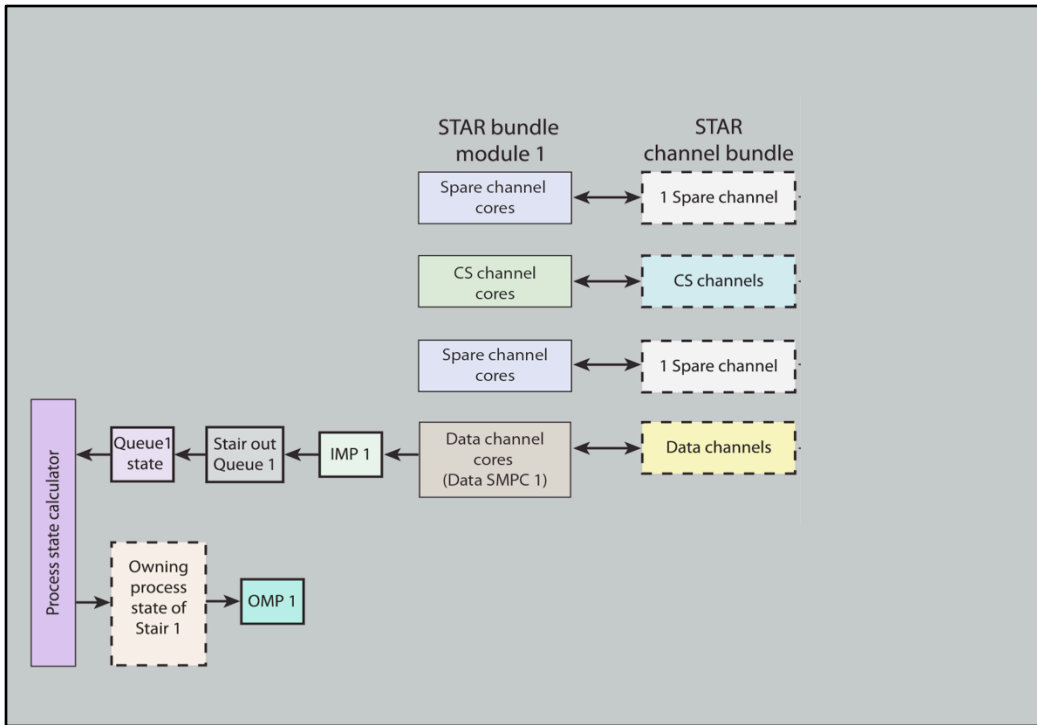
This shows a STAR network including a binary graph of channel bundles, whose nodes are STRs, interfacing through bundle modules to cores, or core modules, in particular, the Programmable Execution Modules (PEMs) 0:3, in a chip.



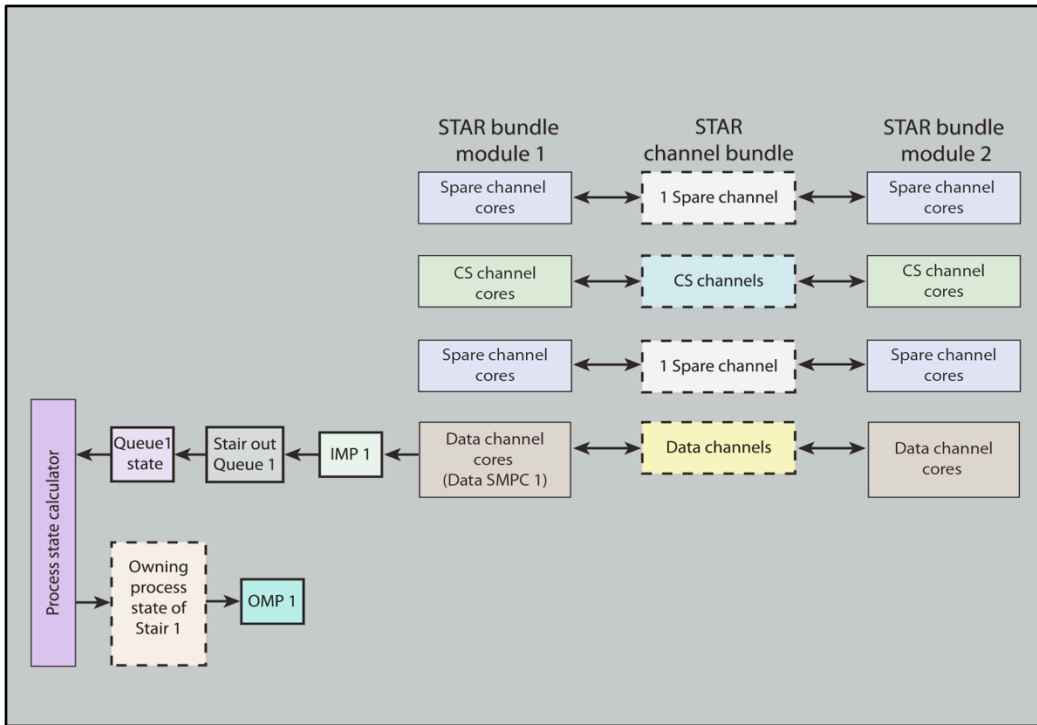
This shows the STAR network interface in a first PEM or STR



As part of a STAR bundle module



communicating across a STAR channel bundle



to a second STAR, bundle module.

Exascale Communication Reliability Constraints

- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime

Exascale Communication Reliability Constraints

- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate



Exascale Communication Reliability Constraints

- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate
 - The number of data channels/bundle communicating with each of the chips

Exascale Communication Reliability Constraints

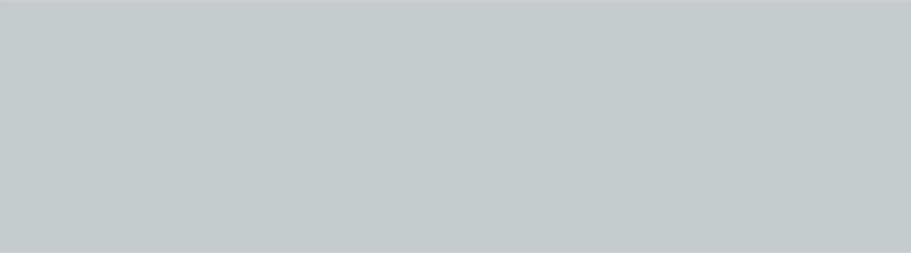
- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate
 - The number of data channels/bundle communicating with each of the chips
 - The number of optical fibers used in each data channel,

Exascale Communication Reliability Constraints

- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate
 - The number of data channels/bundle communicating with each of the chips
 - The number of optical fibers used in each data channel,
 - The number of Data Processing Chips (DPCs) in the exascale system and

Exascale Communication Reliability Constraints

- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate
 - The number of data channels/bundle communicating with each of the chips
 - The number of optical fibers used in each data channel,
 - The number of Data Processing Chips (DPCs) in the exascale system and
 - The number of routers needed to avoid deadlocking to these chips.



Exascale Communication Reliability Constraints

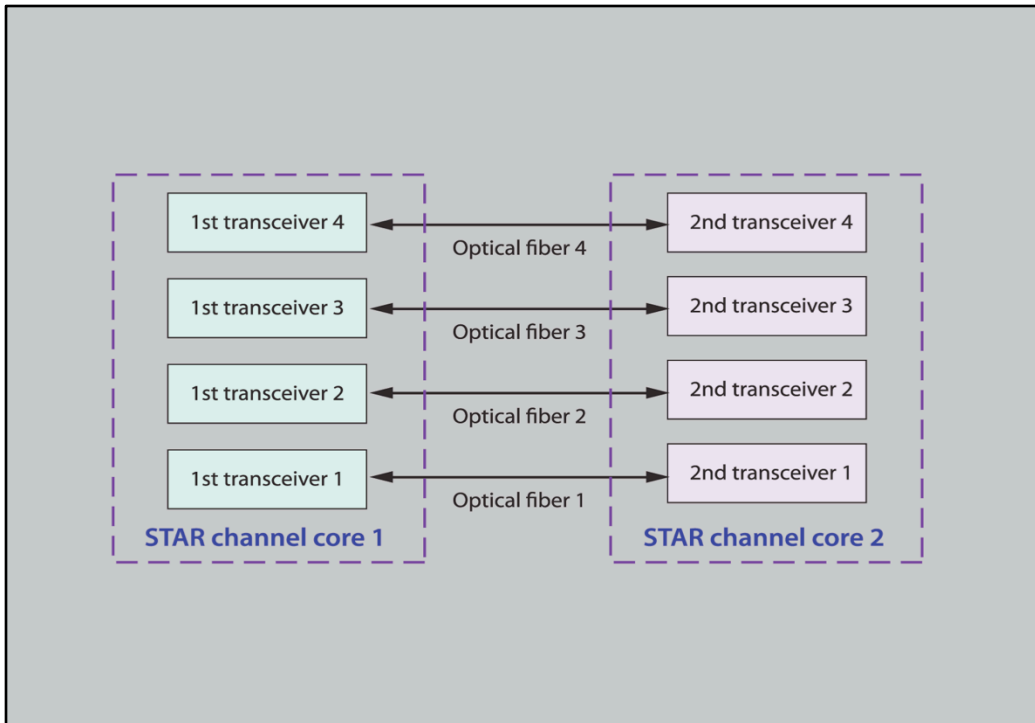
- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate
 - The number of data channels/bundle communicating with each of the chips
 - The number of optical fibers used in each data channel,
 - The number of Data Processing Chips (DPCs) in the exascale system and
 - The number of routers needed to avoid deadlocking to these chips.
- This leads to a Mean Time Between Failure (MTBF) for the supercomputer being no more than minutes.

Exascale Communication Reliability Constraints

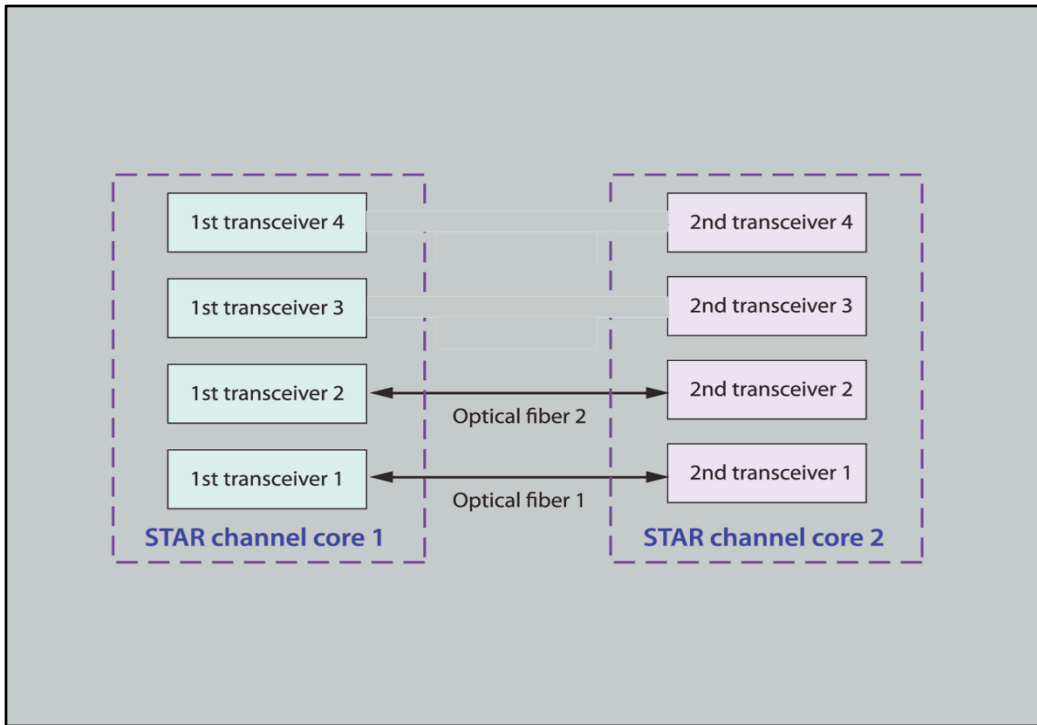
- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate
 - The number of data channels/bundle communicating with each of the chips
 - The number of optical fibers used in each data channel,
 - The number of Data Processing Chips (DPCs) in the exascale system and
 - The number of routers needed to avoid deadlocking to these chips.
- This leads to a Mean Time Between Failure (MTBF) for the supercomputer being no more than minutes.
- When communications fails, the system is likely to fail.
 - There is no time to save these messages elsewhere, or call an operating system.

Exascale Communication Reliability Constraints

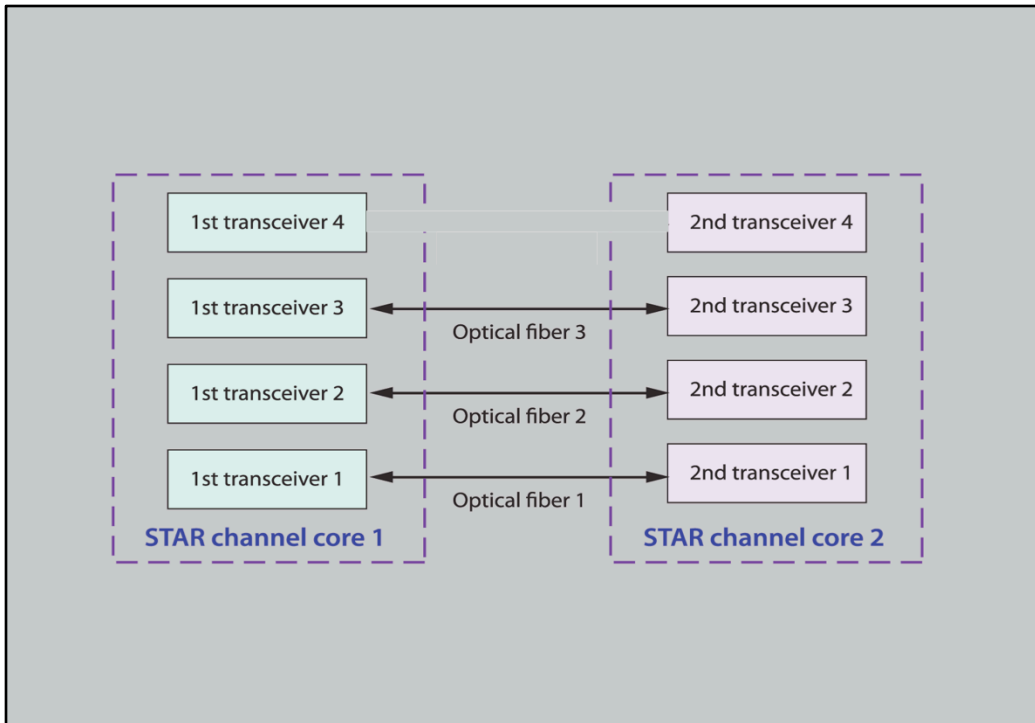
- No single optical fiber can deliver all the bandwidth needed for 500+ cores in a chip to save their state in 1-2% of runtime
- The failure rate for one optical fiber/transceivers, is “statistically multiplied” by the following for communication system failure rate
 - The number of data channels/bundle communicating with each of the chips
 - The number of optical fibers used in each data channel,
 - The number of Data Processing Chips (DPCs) in the exascale system and
 - The number of routers needed to avoid deadlocking to these chips.
- This leads to a Mean Time Between Failure (MTBF) for the supercomputer being no more than minutes.
- When communications fails, the system is likely to fail.
 - There is no time to save these messages elsewhere, or call an operating system.
- This implies that a multi-degree of freedom, fault resilience strategy must be implemented, in local communications hardware.



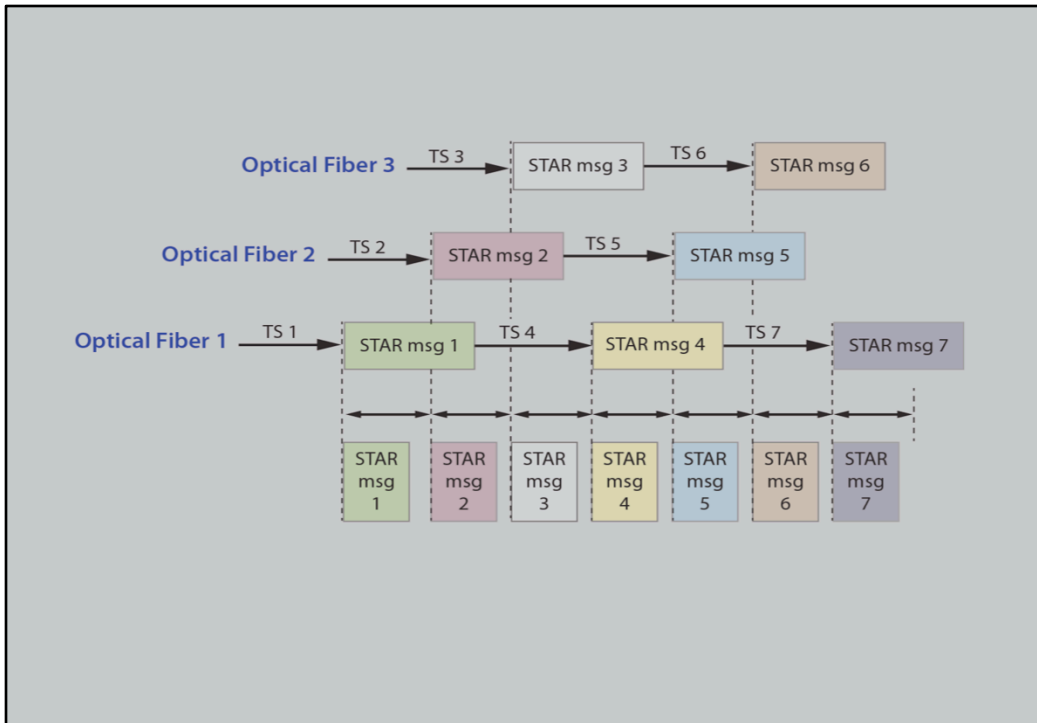
Communication between chips requires optical communications.
 Consider the individual messages, communicated on a STAR message channel.
 This is shown as an implementation consistent with today's 100 Gbit Ethernet opto-transceivers
 using four optical fibers, and associated transceivers, between two STAR channel cores.
 The STAR channel cores, interfaced to a single STAR message channel, support four degrees of freedom
 in automated, local, fault resilience in response to an uncorrectable, message fault



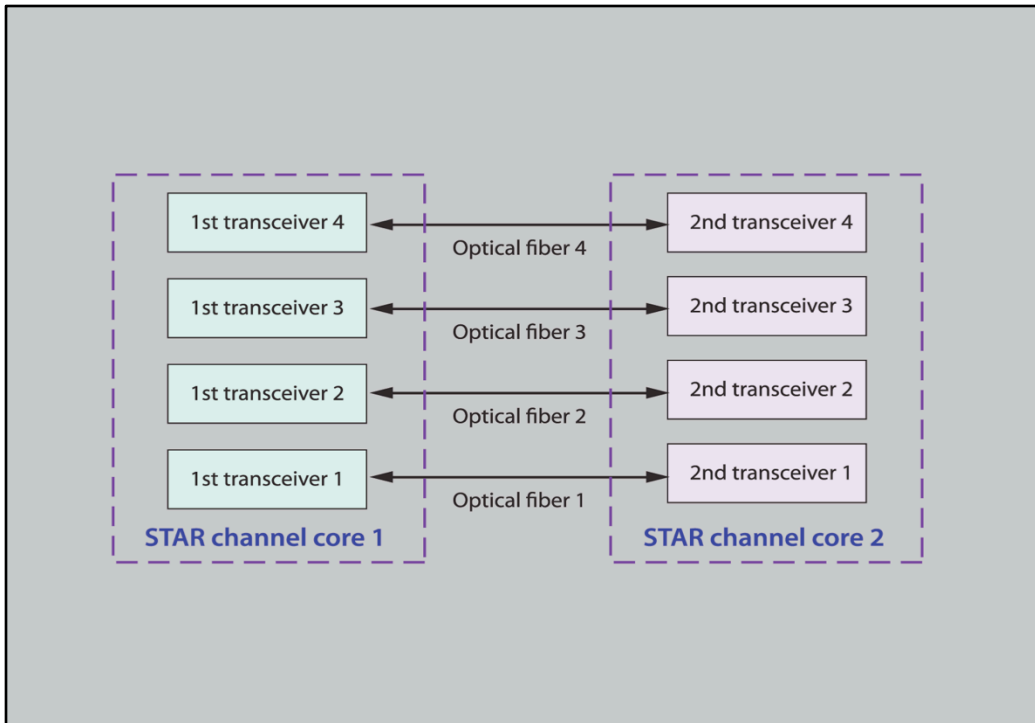
In the best of situations
 2 transceivers pairs can be operated locally at 130-140GHz,
 for 100 Gbit/sec transfers across two optical fibers
 delivering a STAR message every ns.



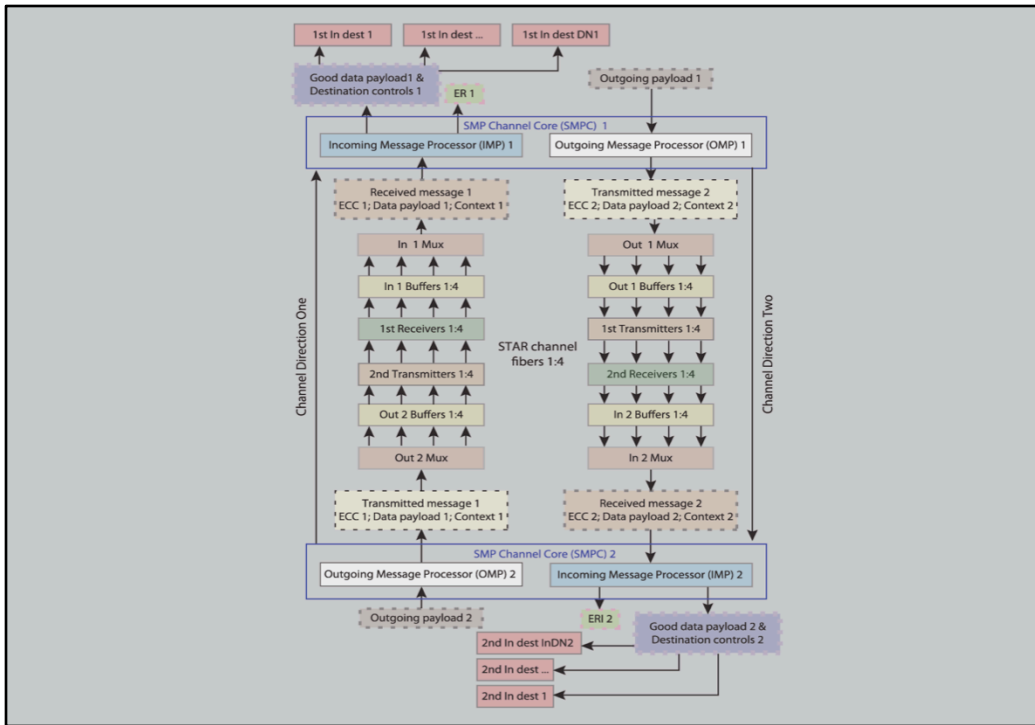
In other situations
3 transceiver pairs
are operated locally at 100 GHz
for 66-70 Gbit/sec transfers across three optical fibers
delivering a STAR message every ns



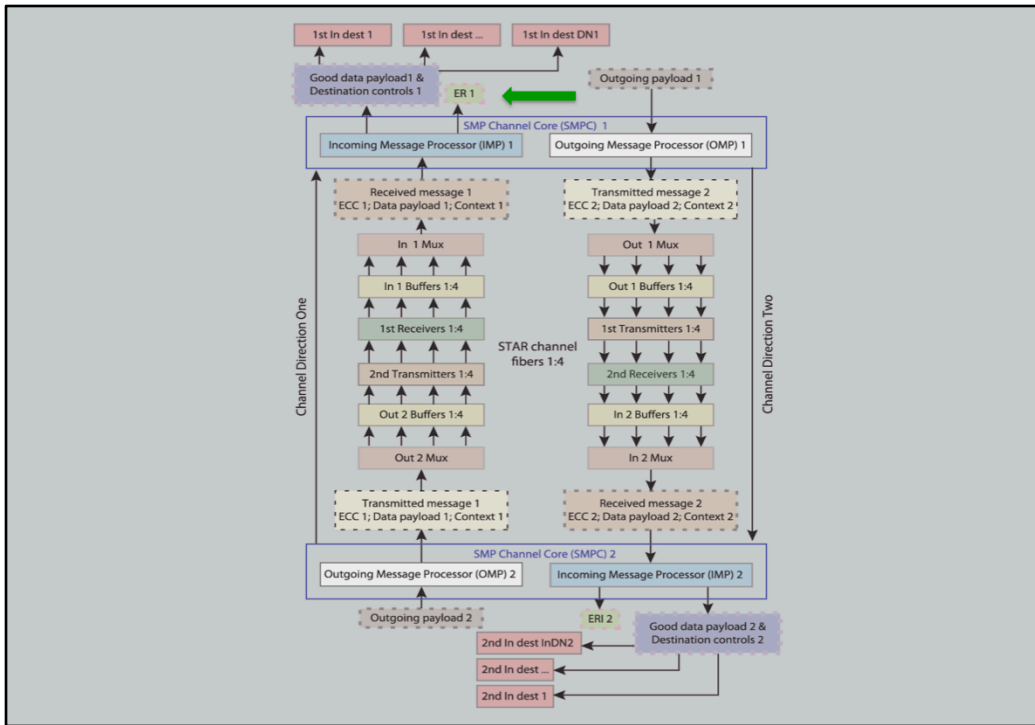
This shows transmitting and receiving STAR messages using three optical fibers. TS k stands for Training Sequence for STAR message k for k=1:7



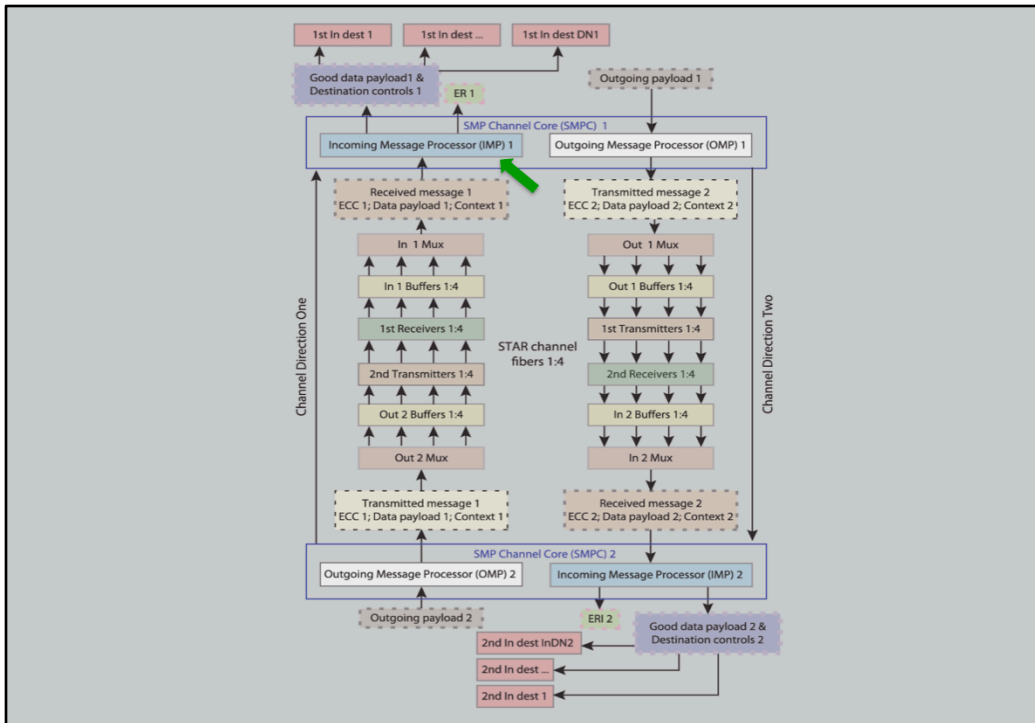
Fault resilience can use all four optical fibers.
Consider the automated response when operating all 4 fibers fails.
The faulty link, from transmitters to receivers in one direction,
is automatically replaced by the spare optical fibers and circuitry in that direction.



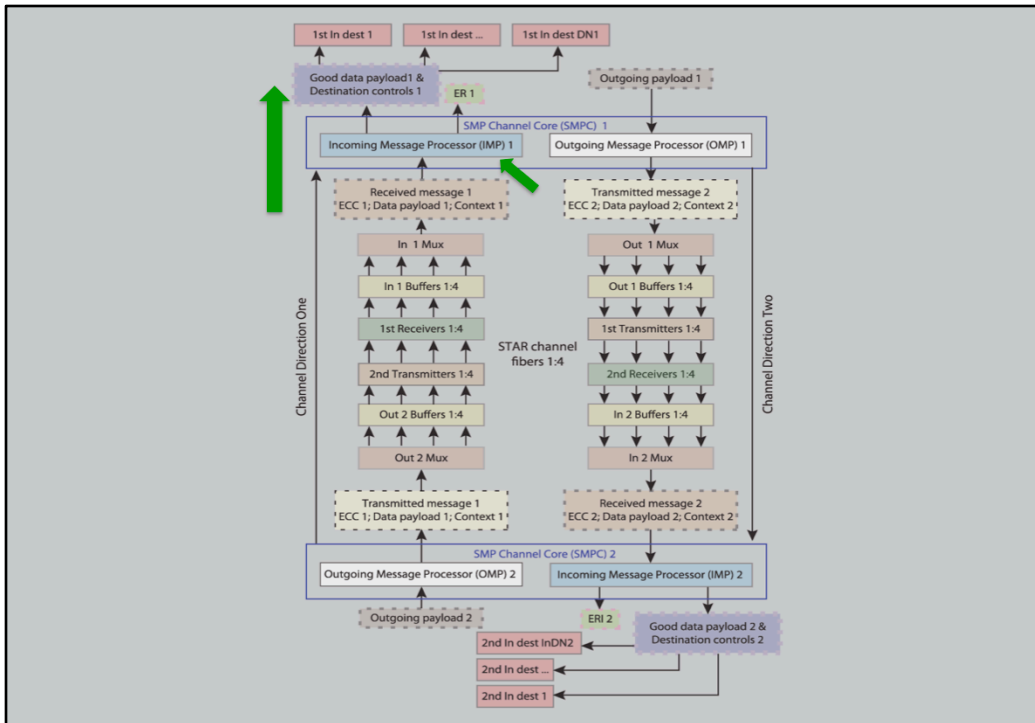
This shows some of the details of the STAR channel cores supporting the interactions we just discussed occurring in a link.



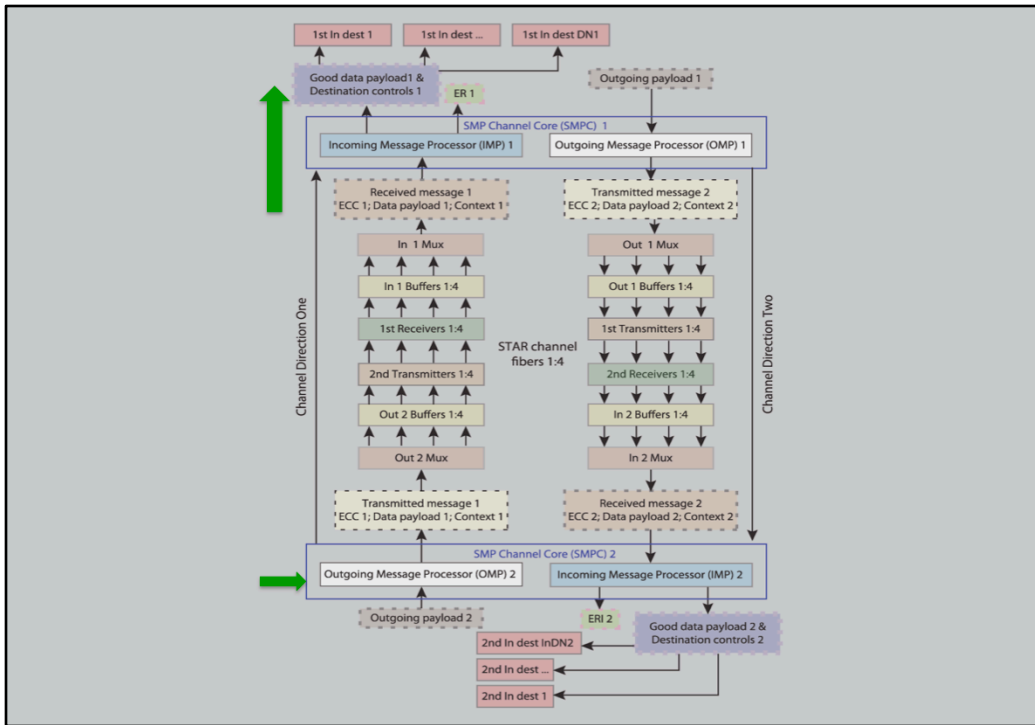
During the normal operations, ER 1 is not asserted,



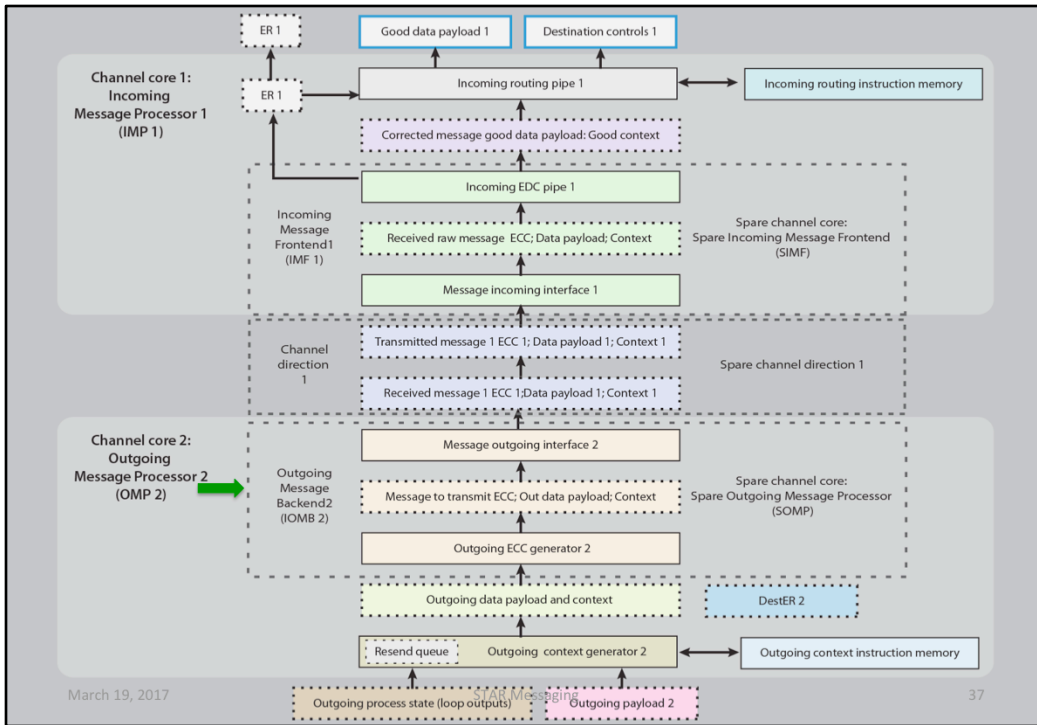
so the incoming selector



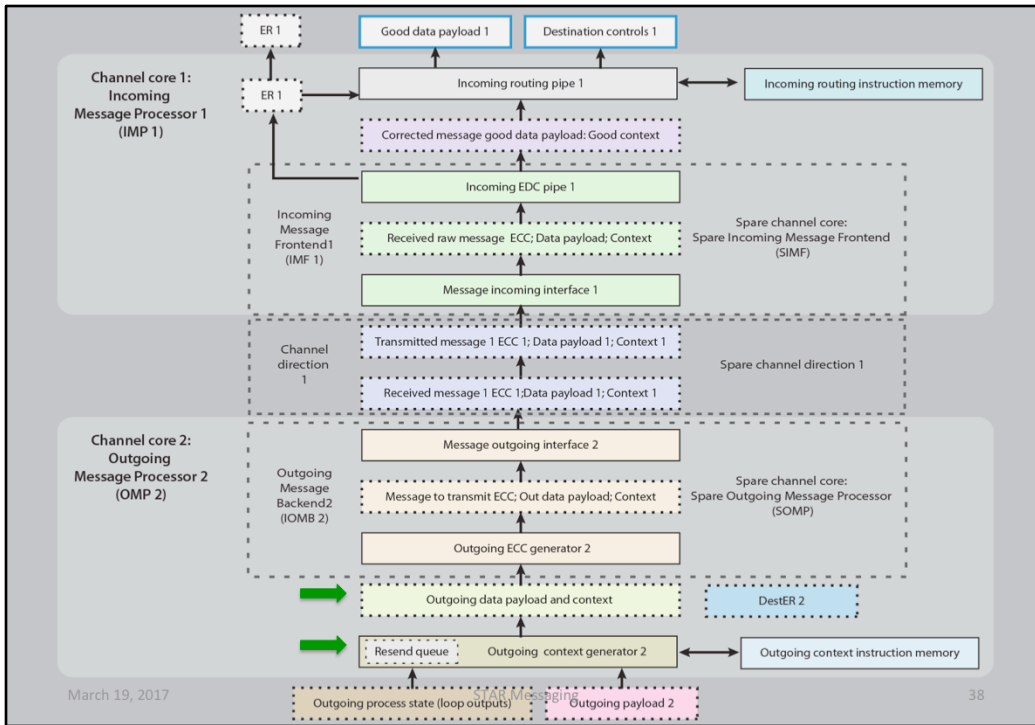
selects the received message from channel direction one.



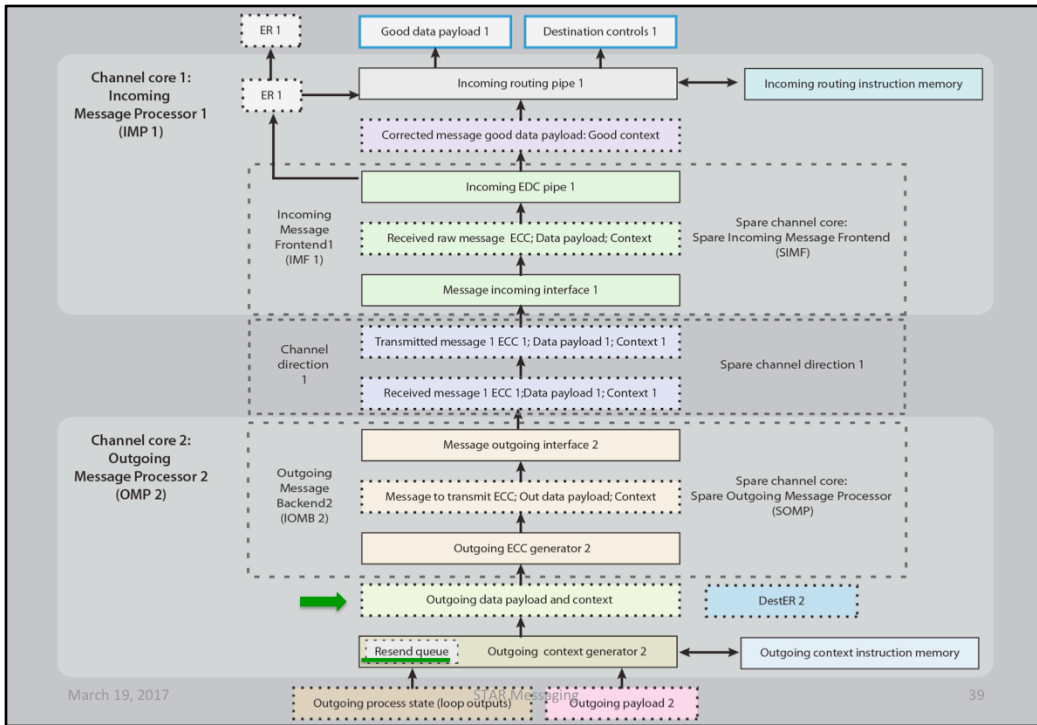
The outgoing selector stimulates the OMB 2



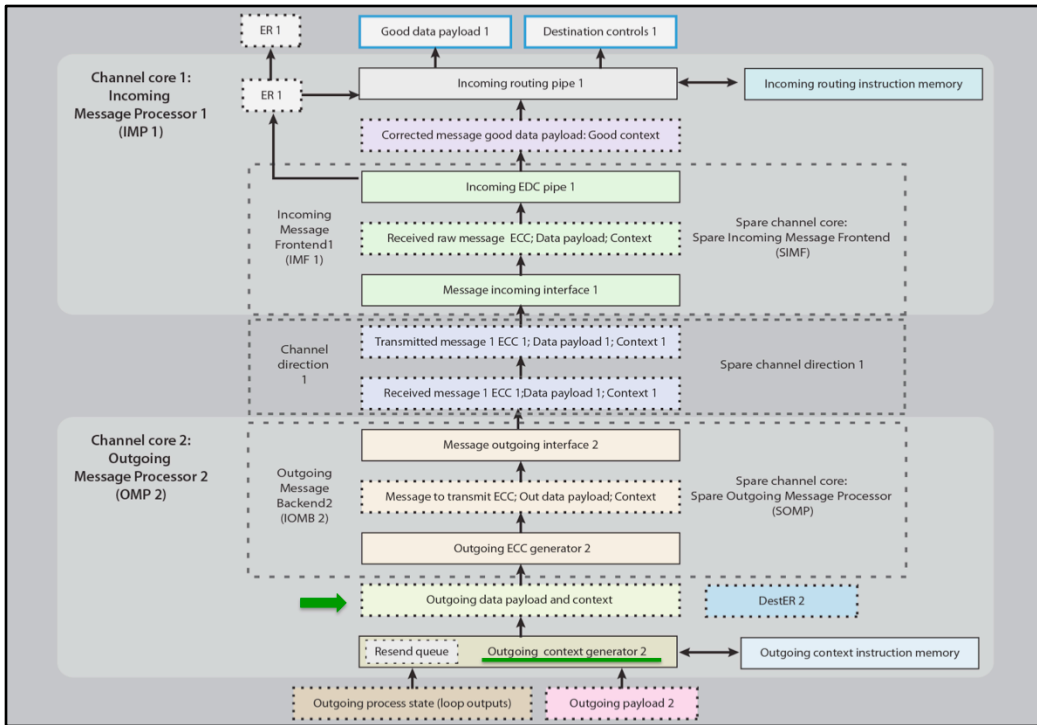
This is seen here, in greater detail.
 OMB 2 is stimulated



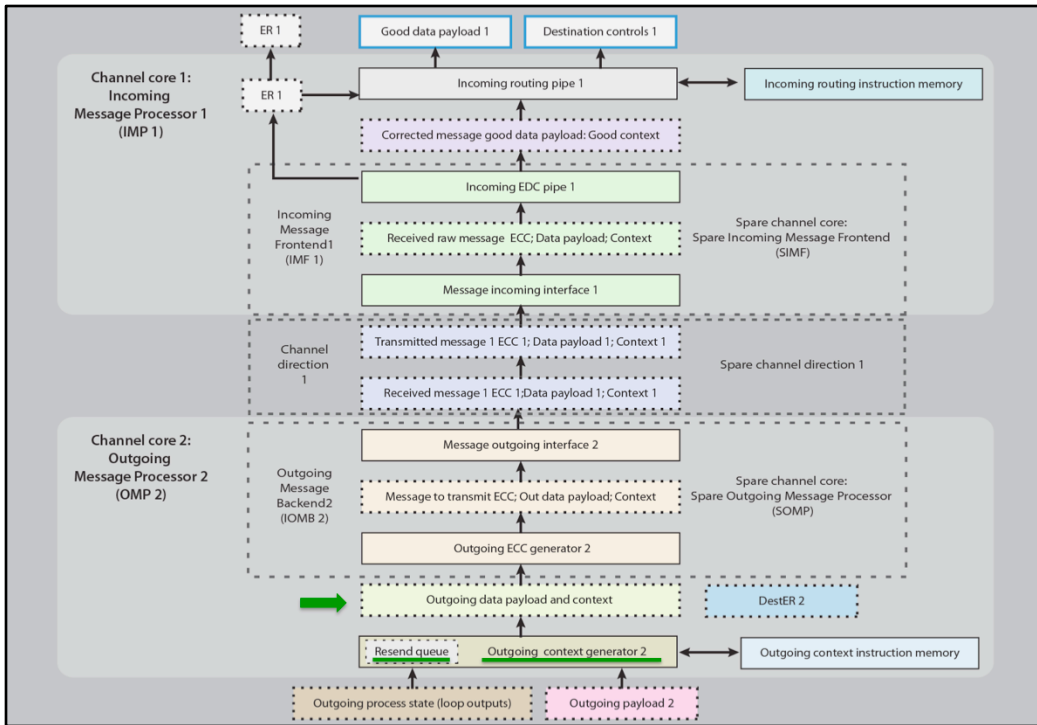
with the package from the outgoing, context generator.



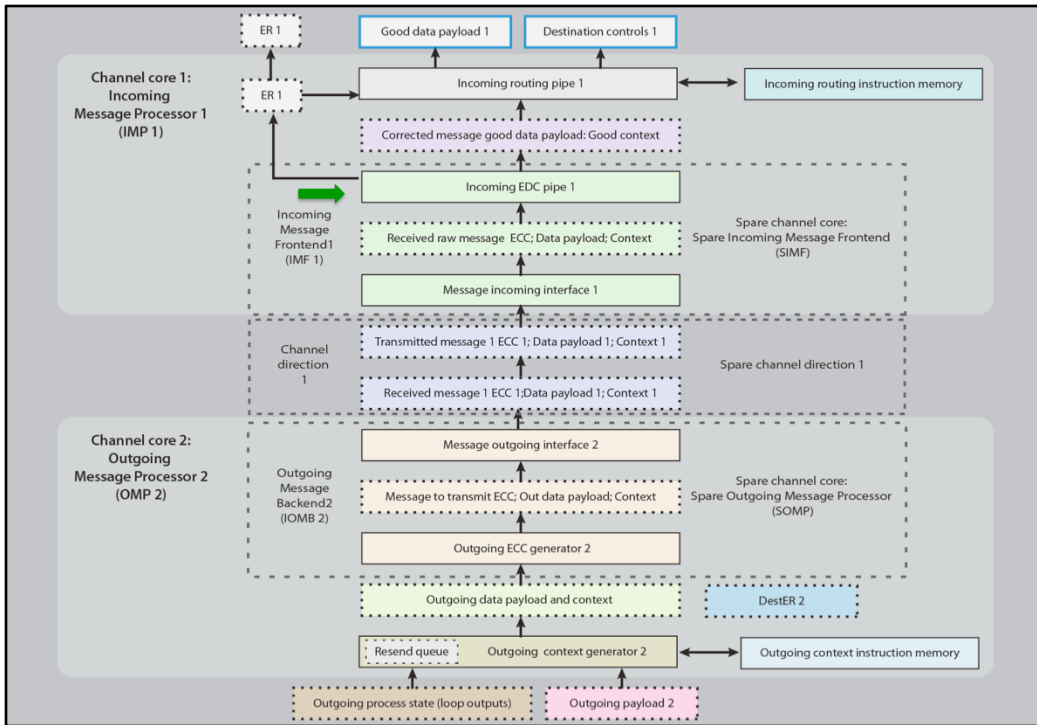
If the resend, queue has a backlog,
 then the package is from the queue directly,



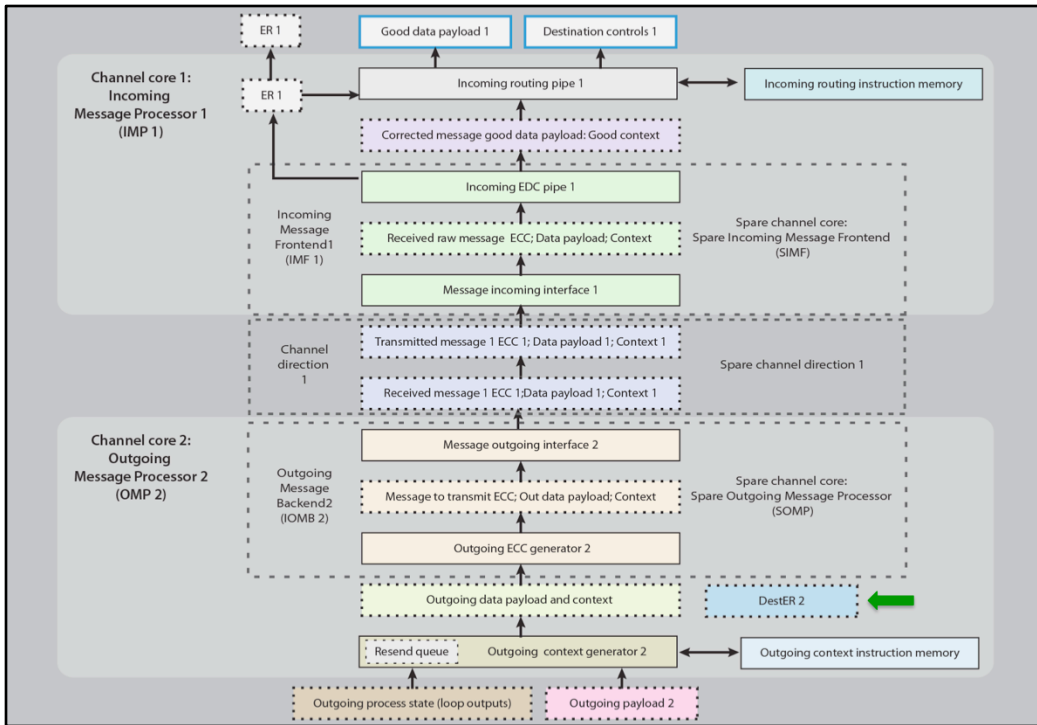
otherwise, the data payload and context is from the generator circuitry.



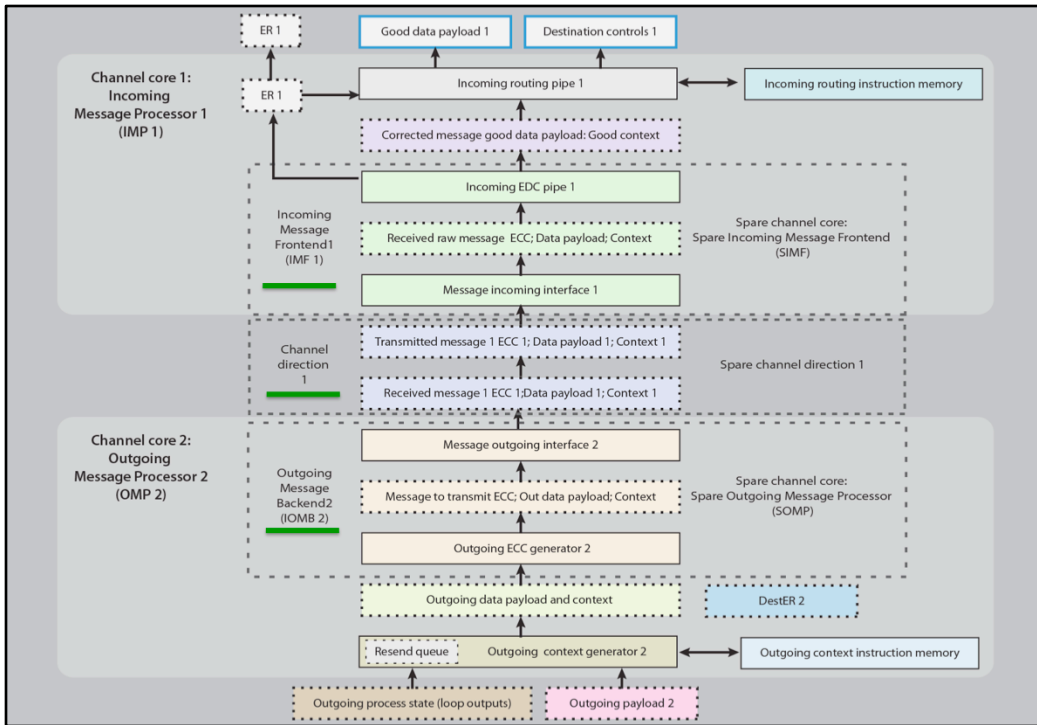
These are logged at the resend queue in case this message fails to be received correctly.
 This failure is detected at the receiver's EDC pipe.



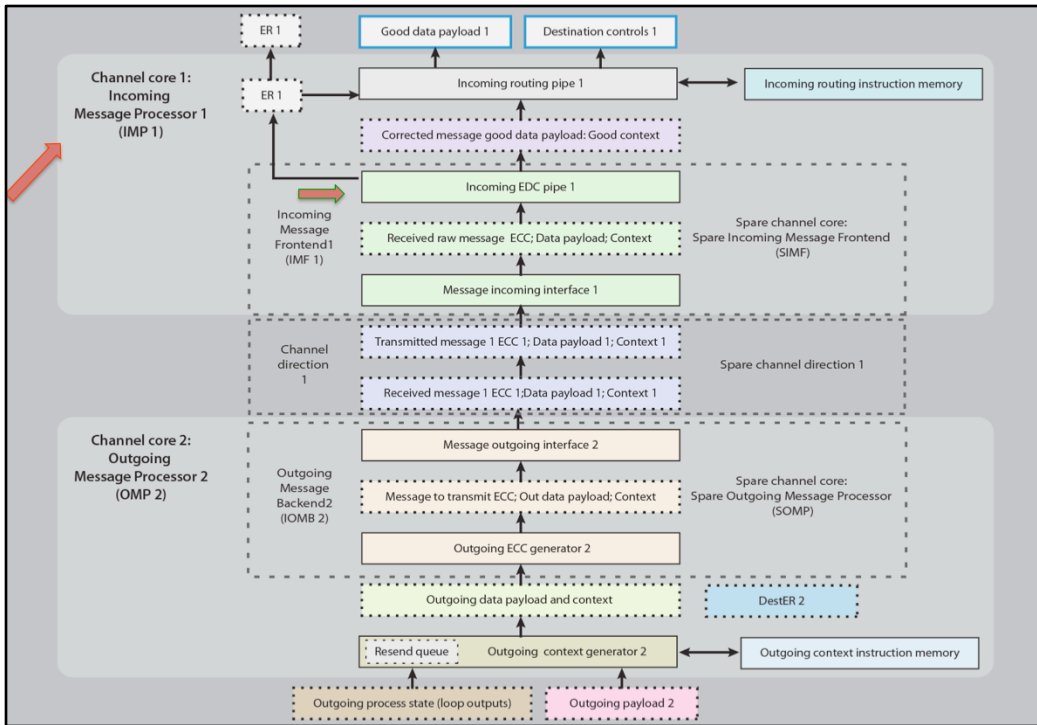
This failure is detected at the receiver's EDC pipe.



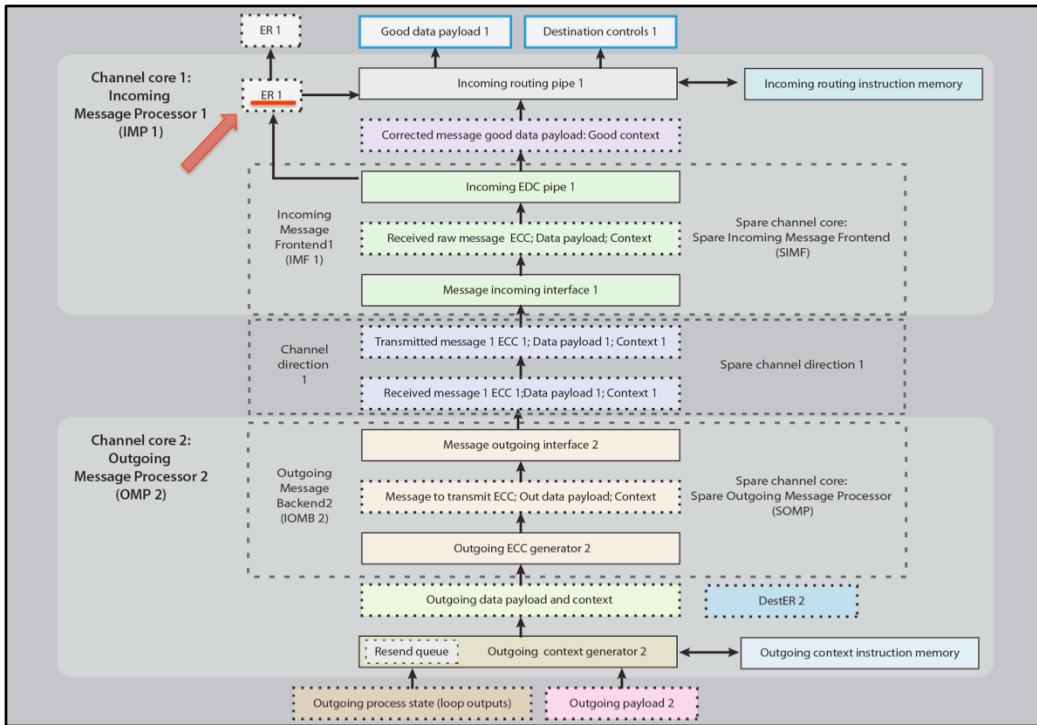
DestER 2 is not set, again because ER 1 has not been asserted.



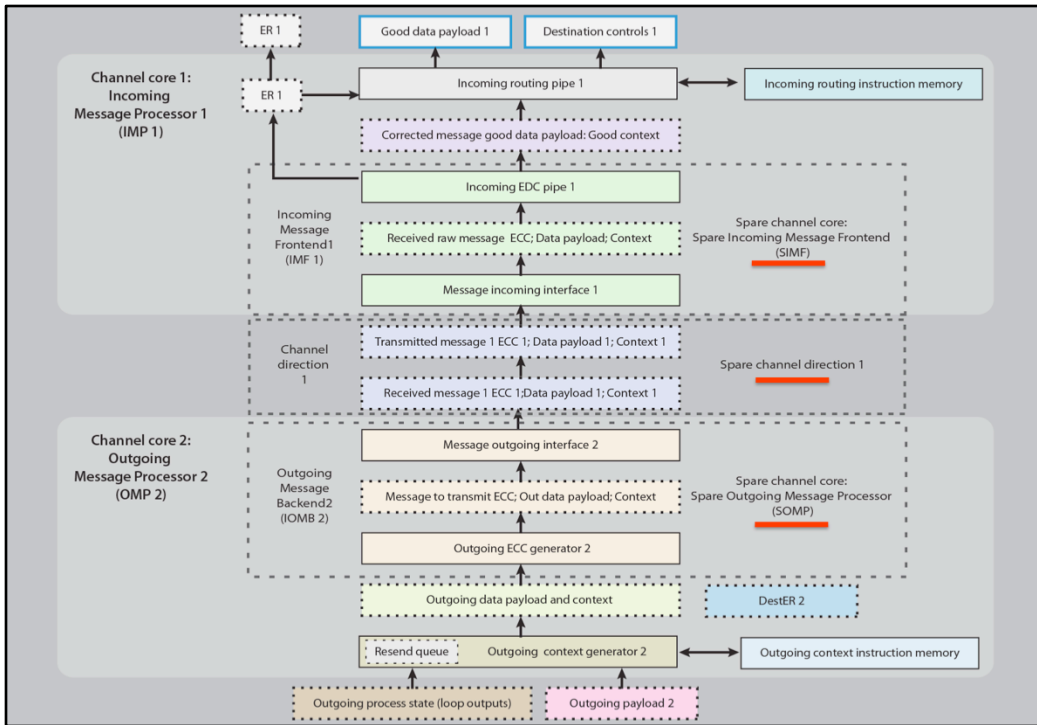
In normal operation,
the left side is used to communicate.



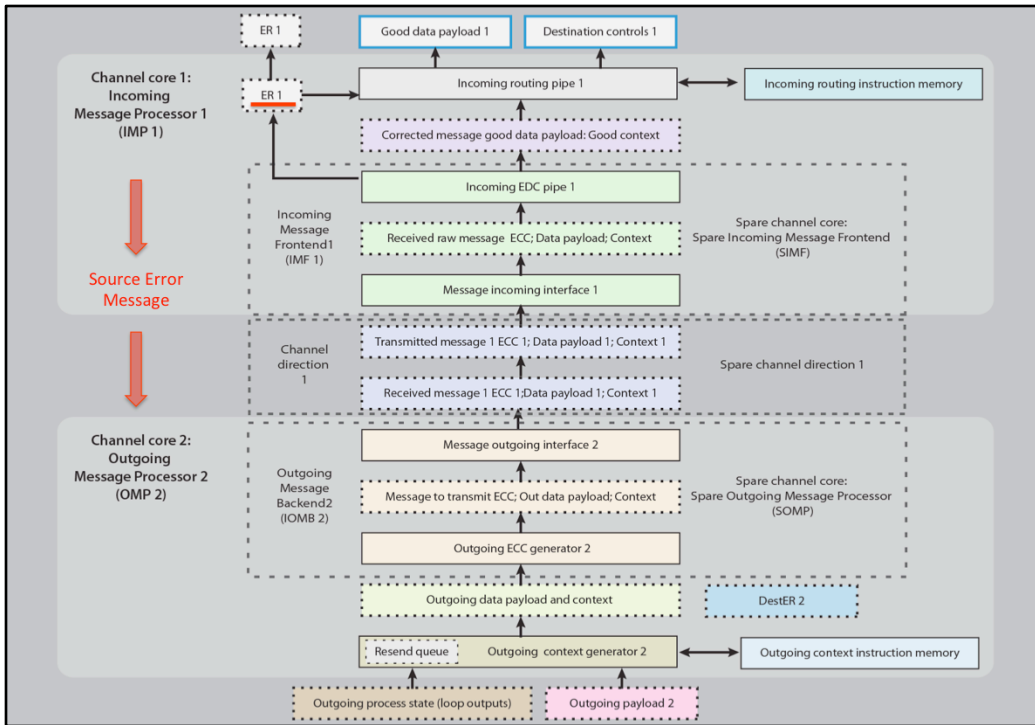
Suppose something bad happens.
 The destination detects, in a few local clock cycles,
 that a received message is fatally flawed.



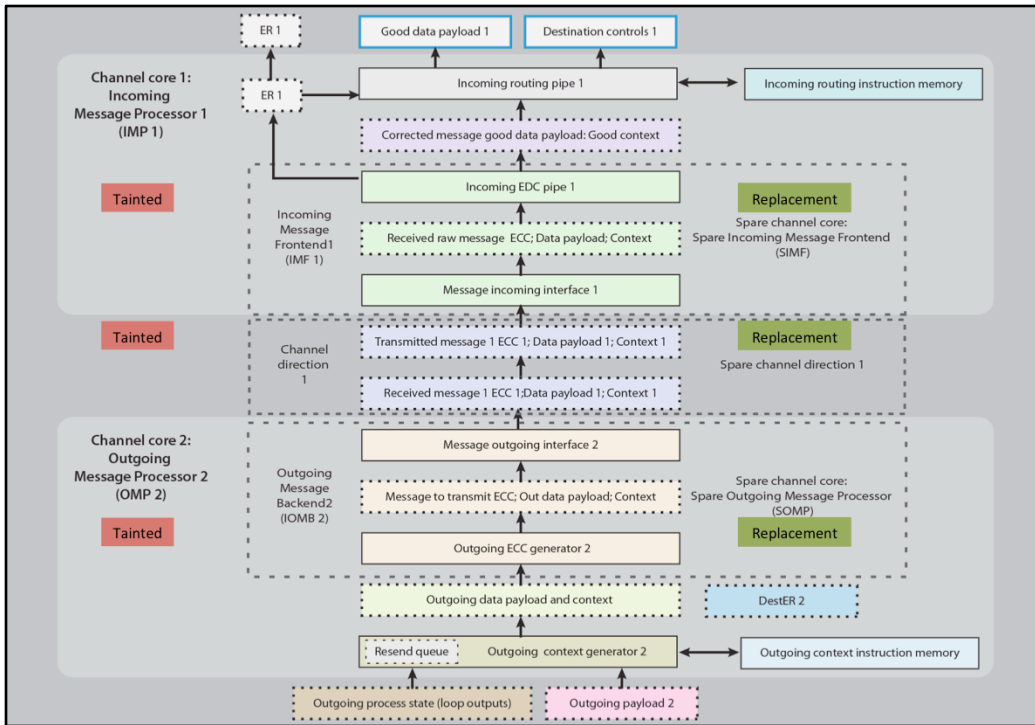
The incoming selector asserts the ER 1



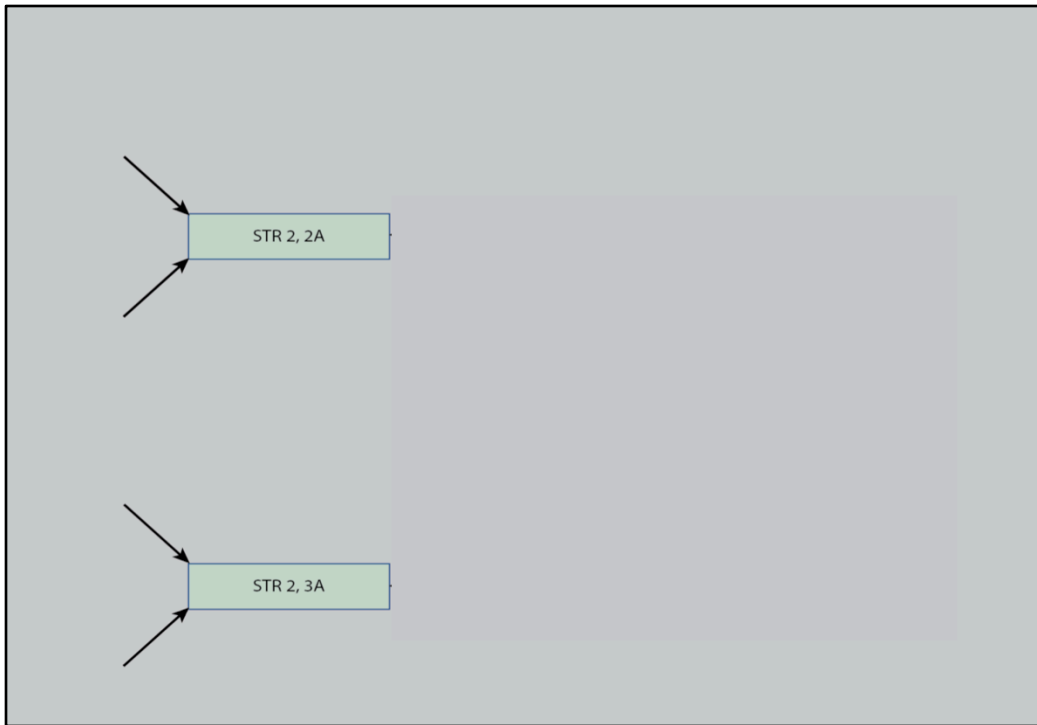
Assume that using more optical fibers has been exhausted.
 Summarizing the fault resilient mode of operation, the right hand side replaces the left side.



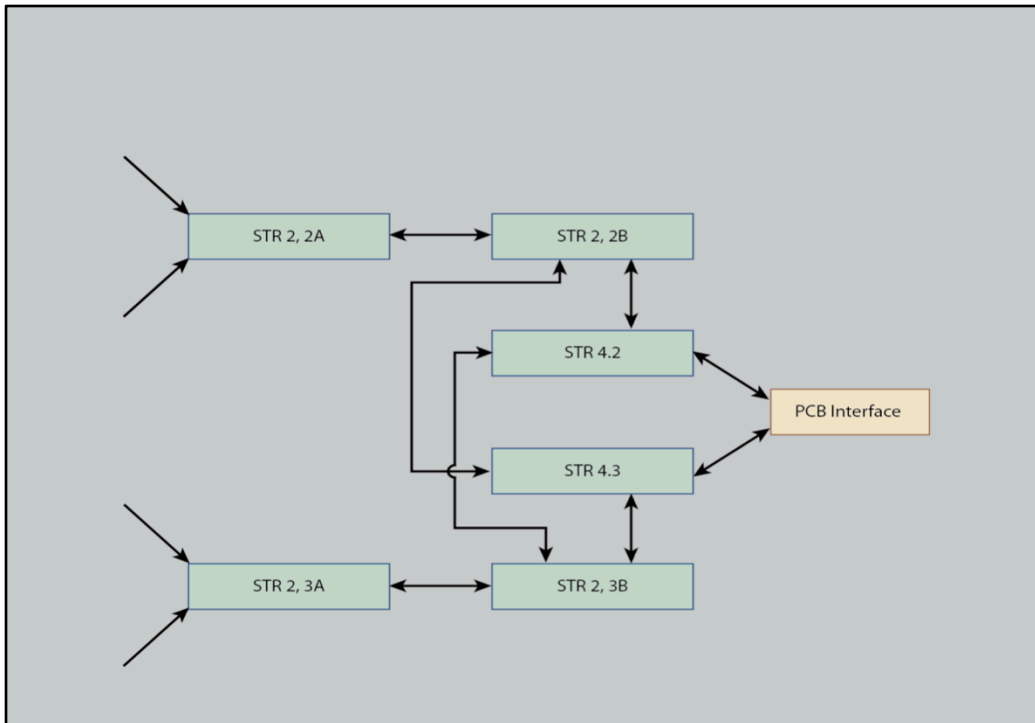
When an unrecoverable error has been detected,
the destination sends via one or more communication/status channels, a
source error message.
When all four optical fibers are in use,
the link treats both source and destination as tainted in this channel
direction.
Before this point, the link will attempt repair
by using more optical fibers and their associated transceivers.



Assuming all 4 optical fibers are in use,
 when the uncorrectable error is detected,
 both the source and destination, are replaced in the flawed direction.
 A spare STAR channel and its spare circuitry, replace the faulty STAR channel,
 and its circuitry.
 This is completed in, at most, a microsecond
 across a computer floor of about 40 meters on a side.



At a different level in the system, binary trees leave HPC components vulnerable to bottlenecks.



Extending the network to a Mostly Binary Tree, enables optical PCBs, etc., to have dual, or better, interfaces.

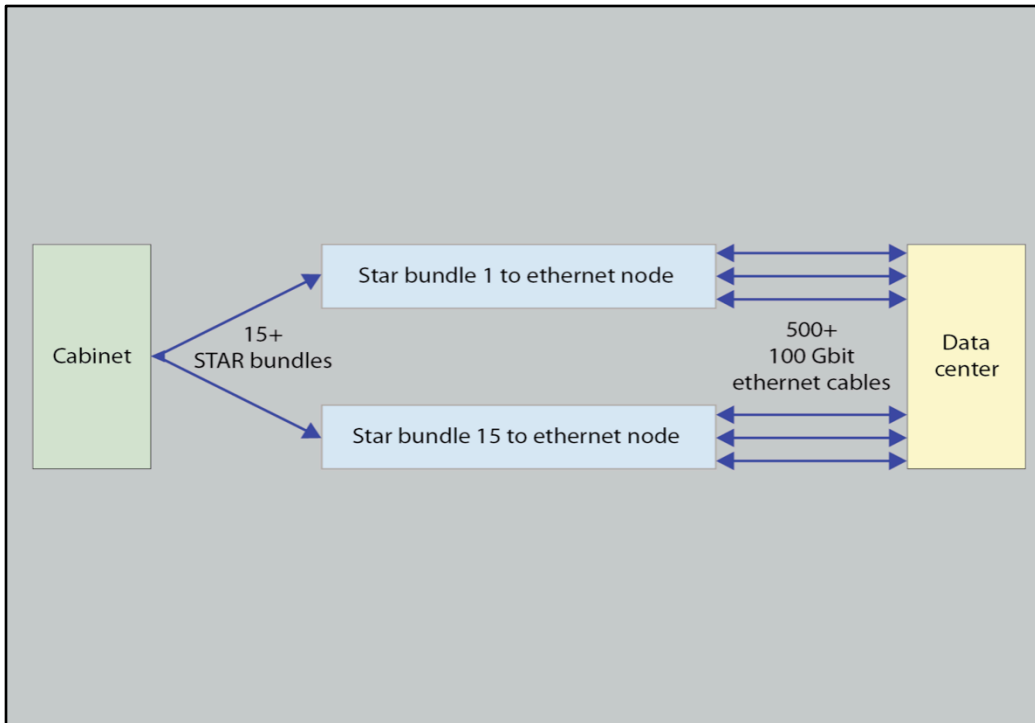
One rack can hold a major sparse matrix system in its cores

with only one access of the DRAM, until the algorithm is done.

Exascale can be achieved in 72 cabinets with the SiMulPro core architecture (from this morning) and the STAR protocol.

This scales seamlessly, from one to 1 K cabinets.

Consider communication between a supercomputer and its data center, where the supercomputer does not stall.



Using the mostly binary network, allows a cabinet, to provide up to 15 or more Star bundles to interface to the datacenter.

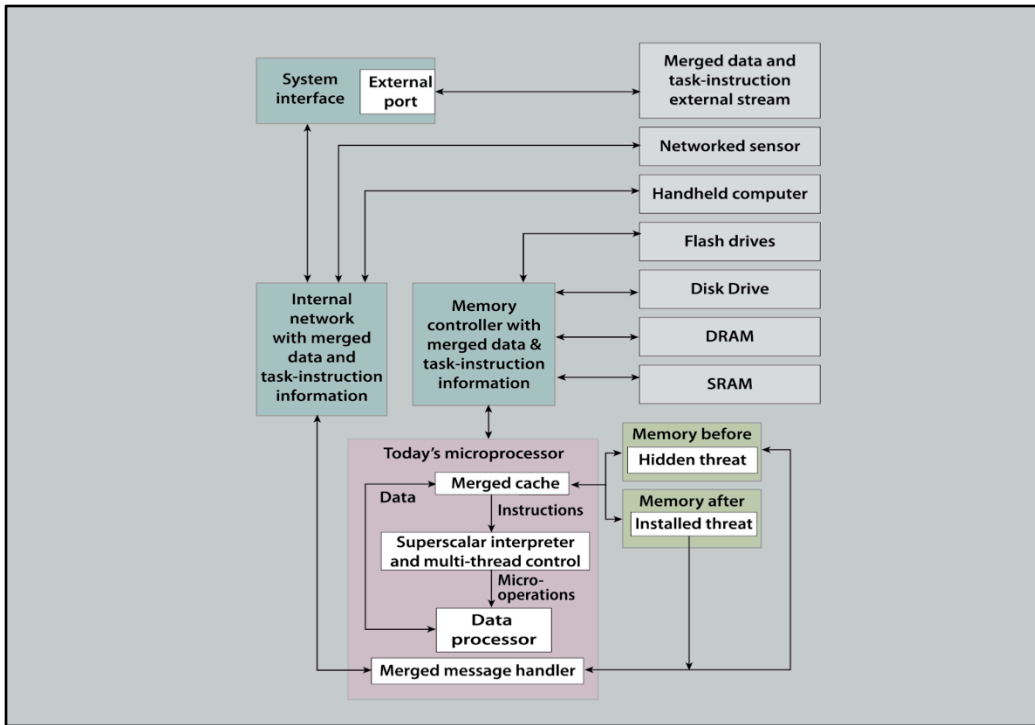
Using these STAR bundles,

each cabinet can simultaneously communicate with 500 or more Ethernet networks,

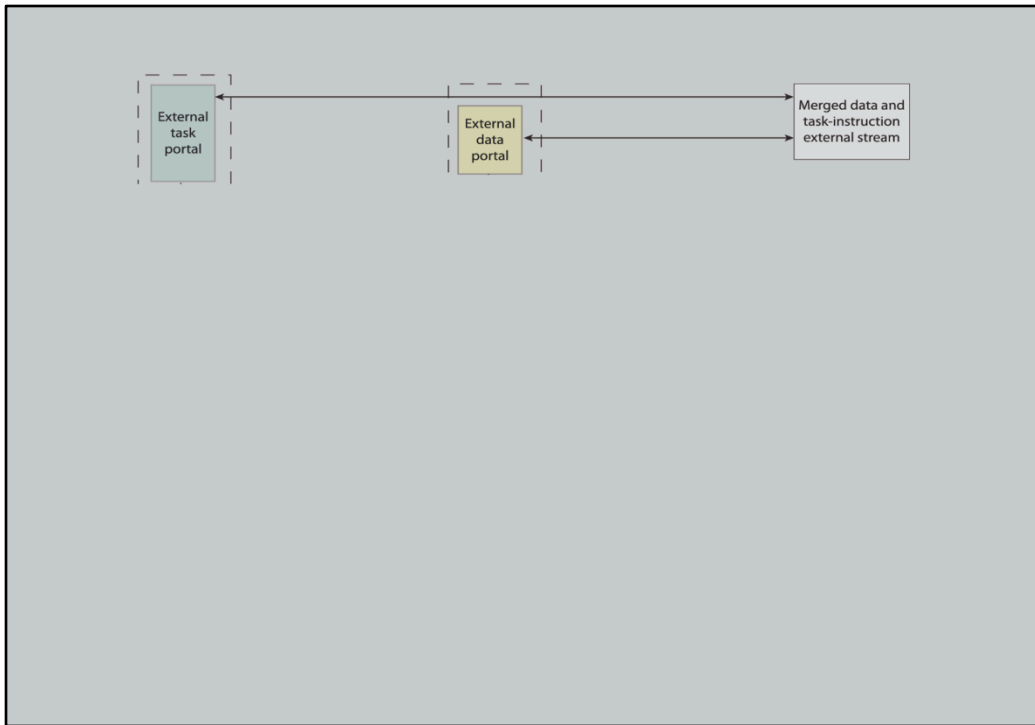
each with 100 Gbit/second bandwidth.

The Star, Mostly Binary, Interface

canonically resolves bandwidth delivery to and from supercomputers for years to come



Today's data centers are vulnerable to malicious software attacks, for example viruses and rootkits. One common weakness is faulty access of data memory, which leads to installed threats, which can infect various components of the data center, and also further infect other sites.

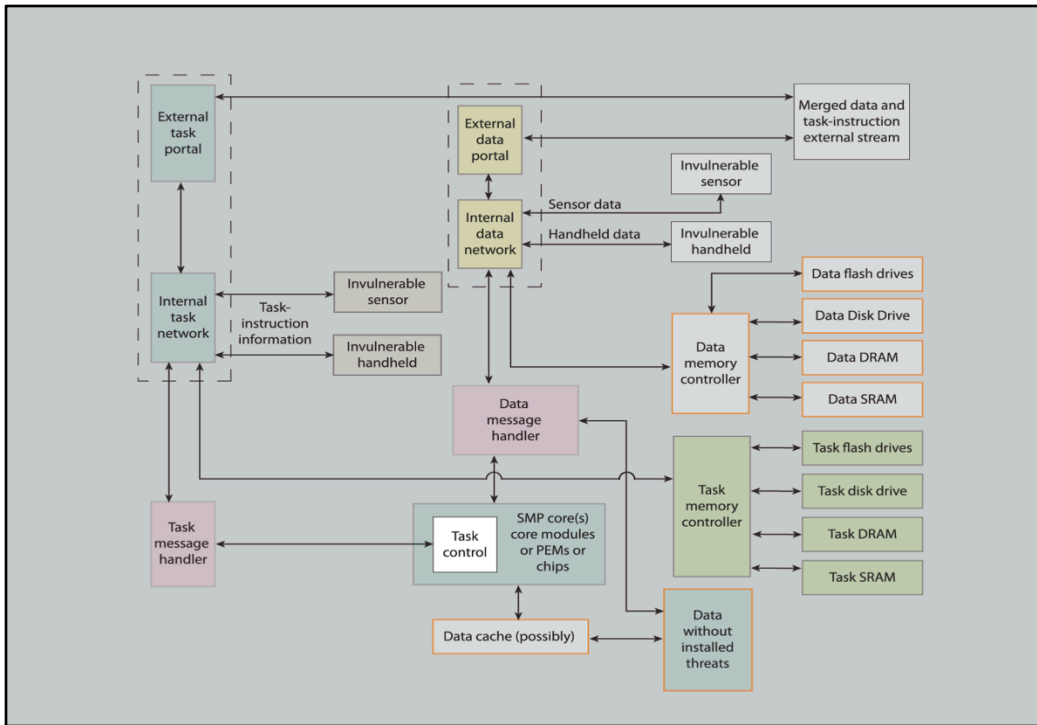


Tomorrow's invulnerable data center

interfaces to less secure, general purpose, networks through a new interface. This new interface operates 2 primary portals, for 2 separate, internal, STAR network components.

One portal supports access to task management and program configuration, going to the task channel of the various STAR links.

A second portal supports data transfers to the STAR data channels.



This invulnerable, data center physically separates data memory, task-instruction memory, and their memory controllers. There are no transfer paths from one form of memory to the other. No data-related operation can alter a task, or an instruction, residing in the task-instruction memory. This removes the opportunity for viruses and rootkits to infect cores, DPCs, handhelds and networked sensors.

References

1. Using MPI: Portable Parallel Programming with the Message-Passing Interface (3rd ed), by Gropp, Lusk, & Skjellum, © 2014 M.I.T., MIT Press, Cambridge, Mass, US
2. MPI-The Complete Reference vol. 1, The MPI Core (2nd ed), by Snir, Otto, Huss-Lederman, Walker, & Dongarra, © 1998 M.I.T., MIT Press, Cambridge, Massachusetts, US
3. MPI-The Complete Reference vol. 2, The MPI Extensions, by Gropp, Huss-Lederman, Lumsdaine, Lusk, Nitzberg, Saphir, & Snir, © 1998 M.I.T, MIT Press, Cambridge, Massachusetts, US
4. Parallel Programming in C with MPI and OpenMP by Quinn, © 2004 McGraw-Hill Companies, Inc., New York, New York, US
5. Optimizing HPC Applications with Intel® Cluster Tools, by Supalov, Semin, Klemm, & Danken, uploaded to Springer's Open Access Oct, 2014.
6. Introduction to InfiniBand™, white paper downloaded from www.mellanox.com website on Feb 22, 2017, rev. 1.90
7. Introduction to High-Speed InfiniBand Interconnect, www.hpcadvisorycouncil.com/pdf/Intro_to_InfiniBand.pdf on Feb 22, 2017
8. Top Ten Exascale Research Challenges, DOE-ASCAC Subcommittee Report, (Feb 10, 2014), US

Thank you, are there any questions?