Data Acquisition Backbone Core

Jörn Adamczewski, Hans G.Essel, Nikolaus Kurz, Sergey Linev
GSI, Experiment Electronics: Data Processing group

Again: motivation and use cases

Again: system design and achievments since July (Huelva)

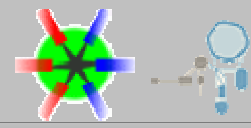
New: MBS application as example

New: performance measurements (optional)

To do

Work supported by EU RP6 project JRA1 FutureDAQ RII3-CT-2004-506078

Driving forces for DABC



2004 → EU RP6 project JRA1 FutureDAQ*
2004 → CBM FutureDAQ for FAIR

1996 → MBS future
50 installations at GSI,
50 external
http://daq.gsi.de

Intermediate demonstrator

Requirements

- build events over fast networks
- handle triggered or self-trigger front-ends
- process time stamped data streams
- provide data flow control (to front-ends)
- connect (nearly) any front-ends
- provide interfaces to plug in application codes
- connect MBS readout or collector nodes
- be controllable by several controls frameworks

Use cases

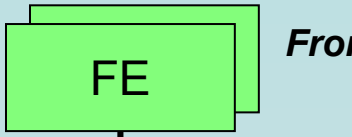
- Detector tests
- FE equipment tests
- Data transport
- Time distribution
- Switched event building
- Software evaluation
- MBS event builder
- General purpose DAQ

Data Acquisition Backbone Core

* RII3-CT-2004-506078

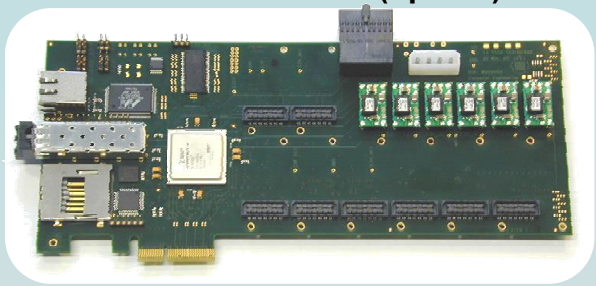
➢ Some use cases

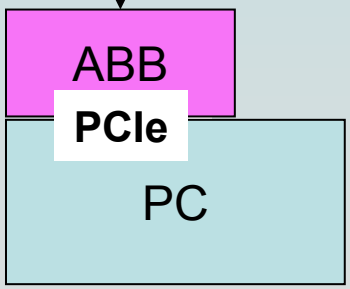
Use case example: Frontend components test



***Front end* board: sampling ADCs, clock distribution**

2.5 GBit/s bi-directional (optical) link: data, clock





***Active Buffer* board *: PCI express card**

FE: Frontend board

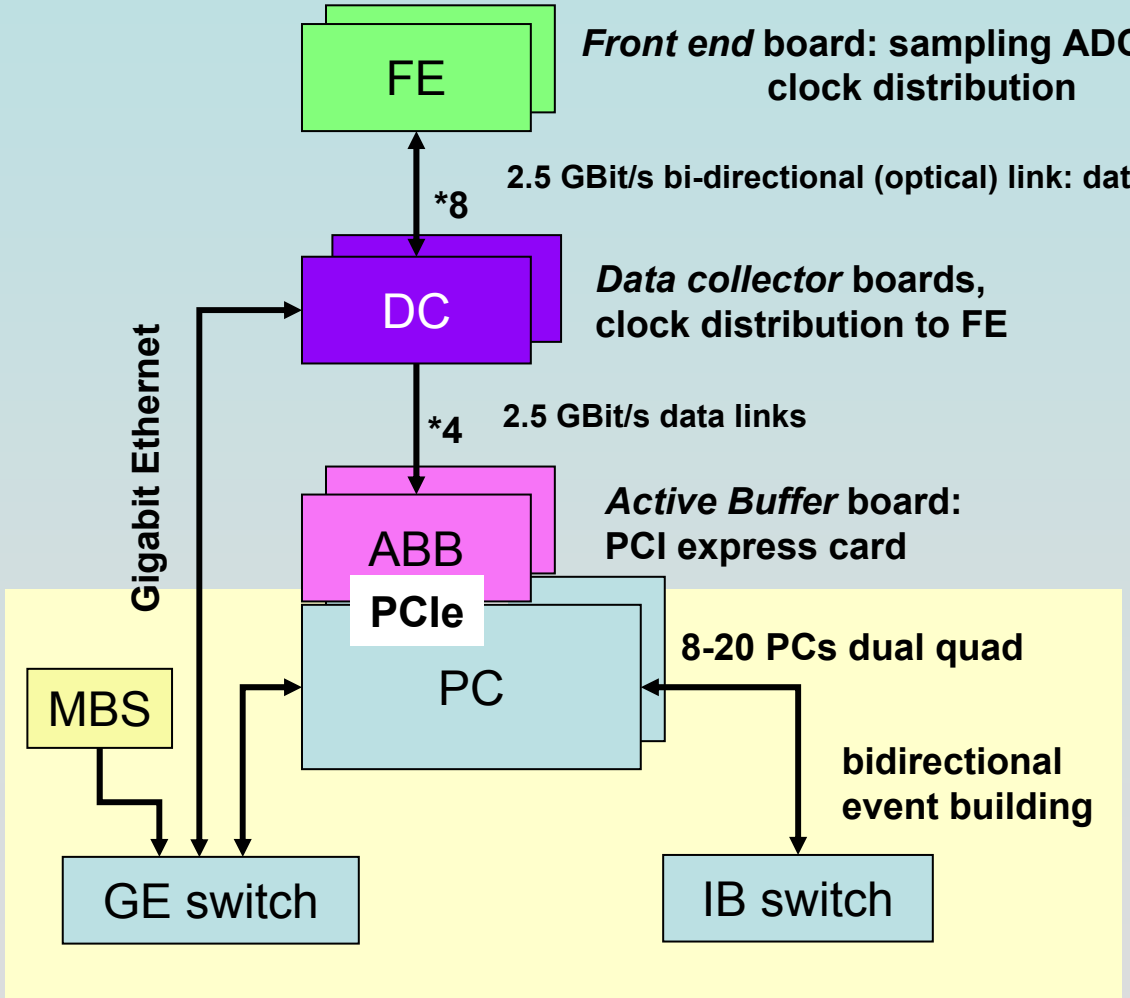
ABB: Active Buffer board

The goal:

- Detector tests
- FEE tests
- Data flow tests

* A.Kugel, G.Marcus, W.Gao, Mannheim University Informatics V

Use case example: Frontend components test

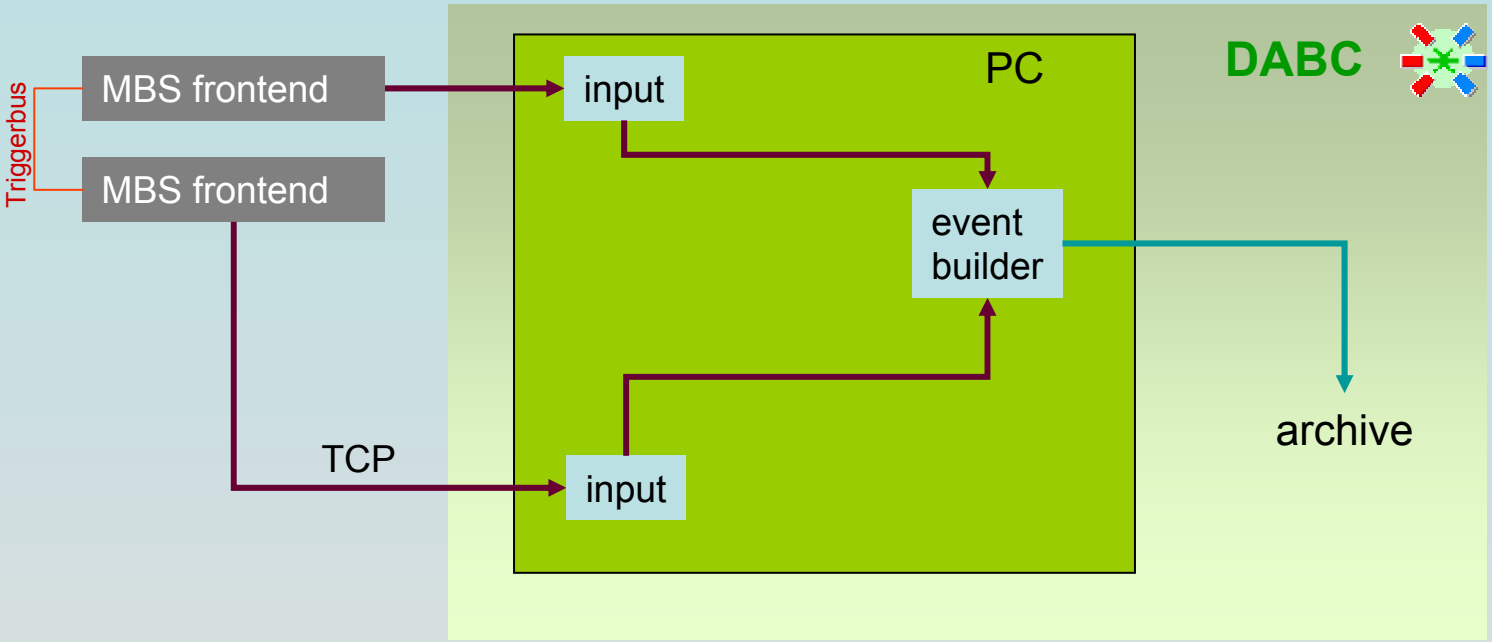


FE: Frontend board
DC: Data combiner board
ABB: Active Buffer board
GE: Gigabit Ethernet
IB: InfiniBand
MBS: MultiBranchSystem

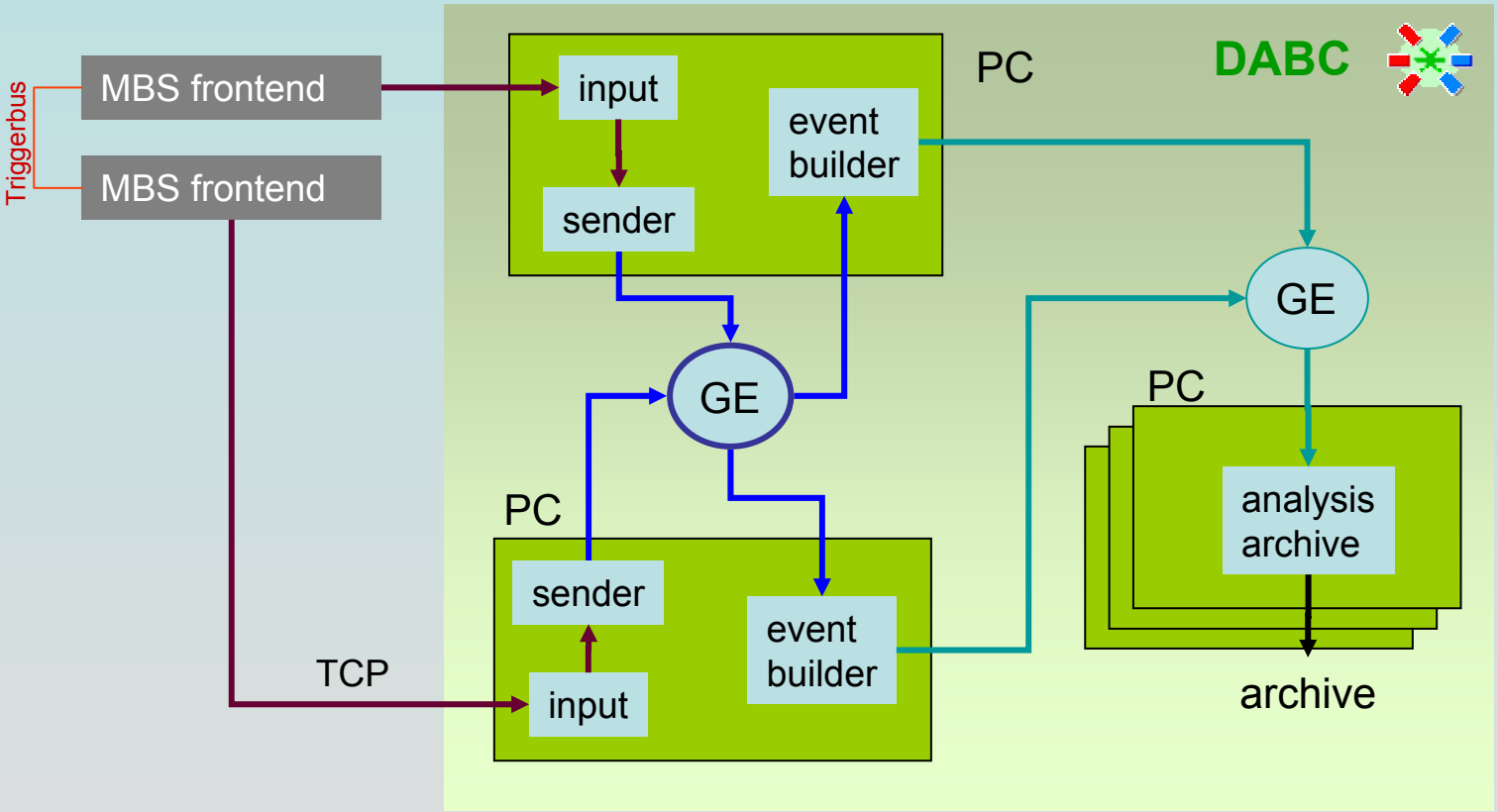
The goal:
- Investigate critical technology
- Detector tests
- Replace existing DAQ

Scales up to 10k channels, 160 CPUs

Use case example: simple MBS



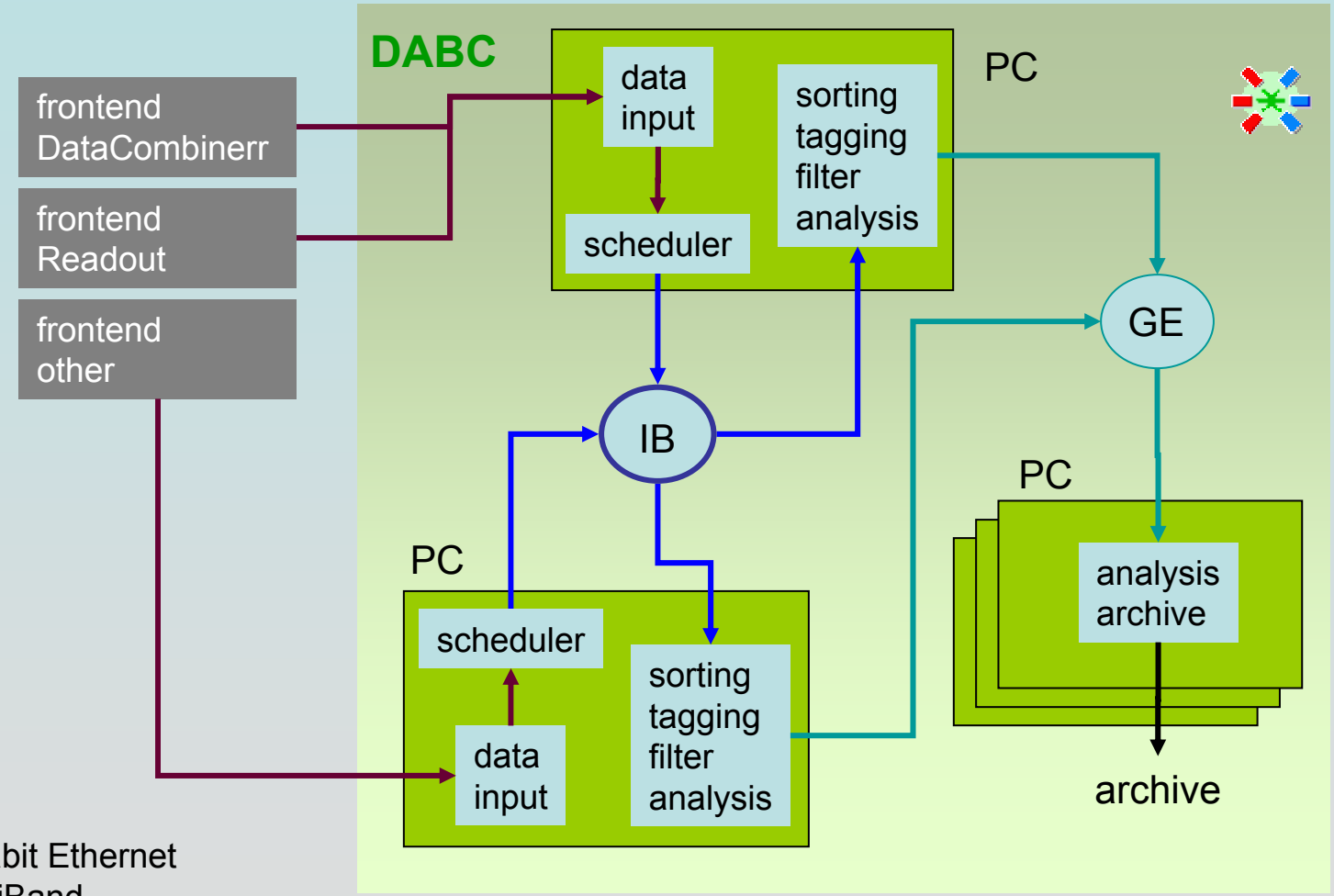
Use case example: complex MBS



GE: Gigabit Ethernet

➢ DABC structure

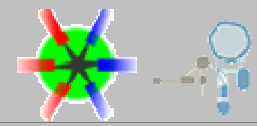
DABC design: global overview



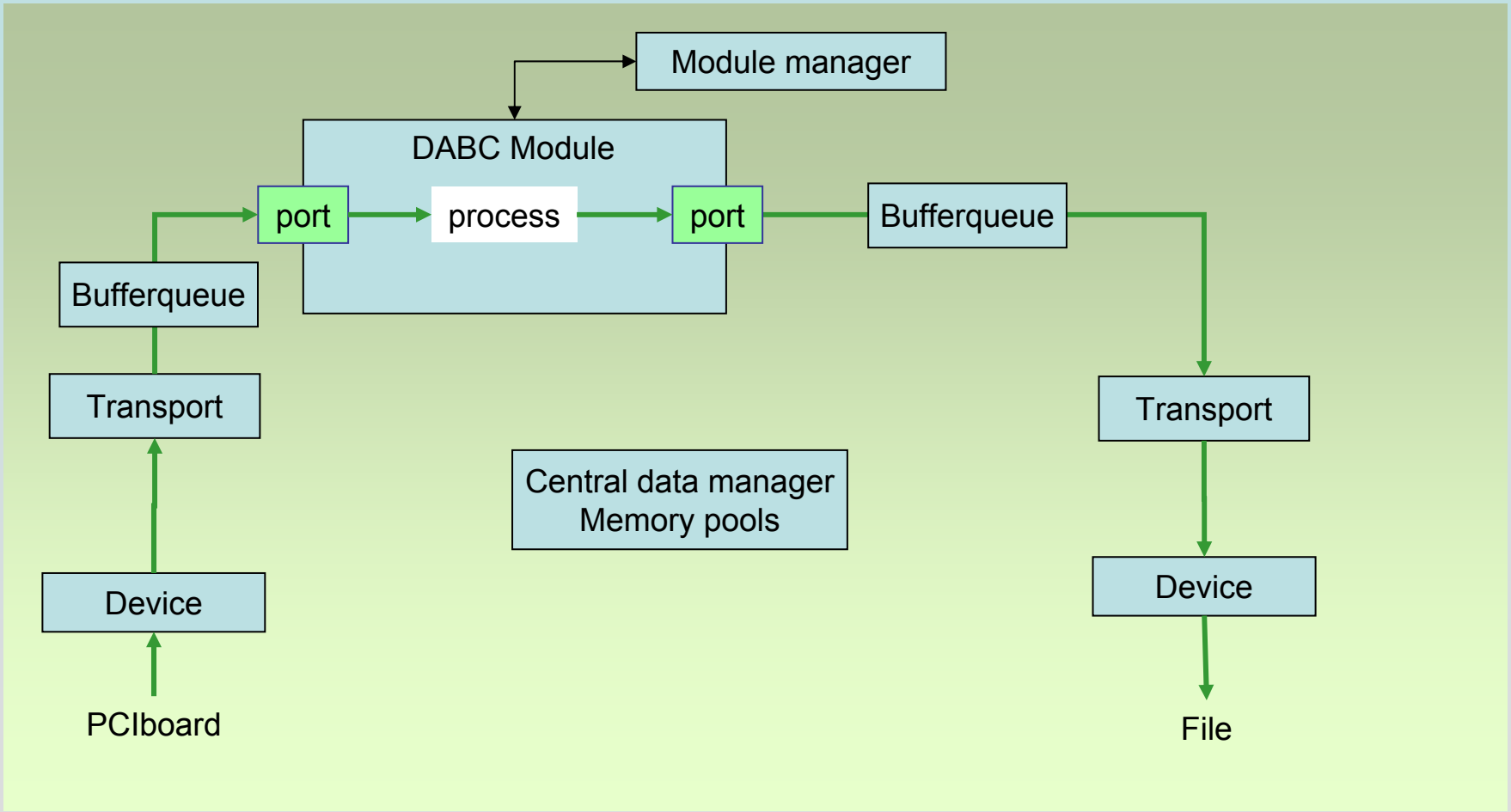
GE: Gigabit Ethernet
IB: InfiniBand

➢ DABC data flow

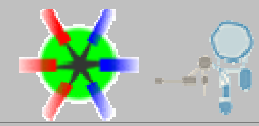
DABC design: Software objects of data flow engine



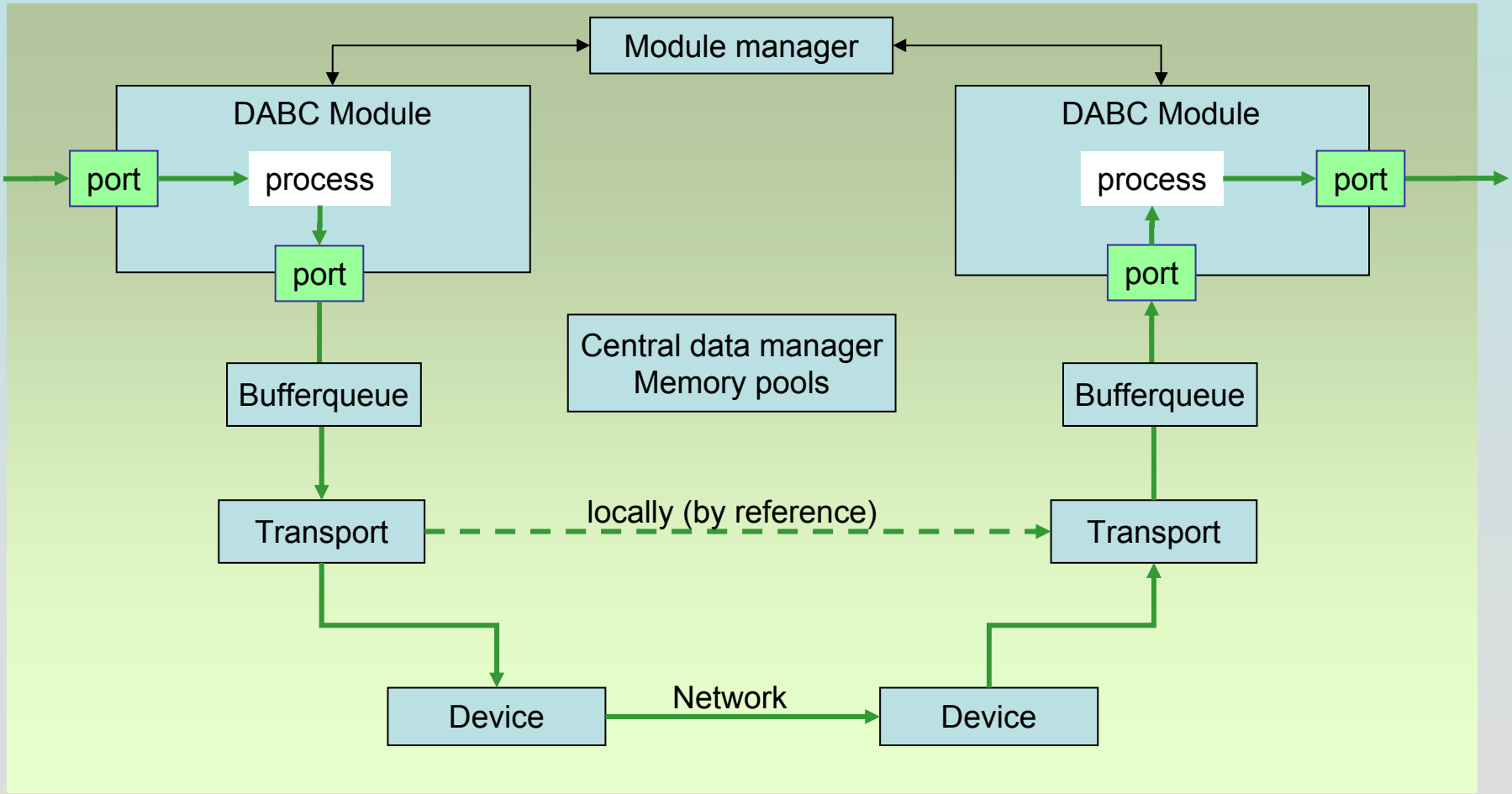
A *module* processes data of one or several data streams.
Data streams propagate through ***ports***, which are connected by ***transports*** and ***devices***



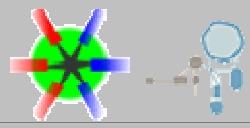
DABC design: Software objects of data flow engine



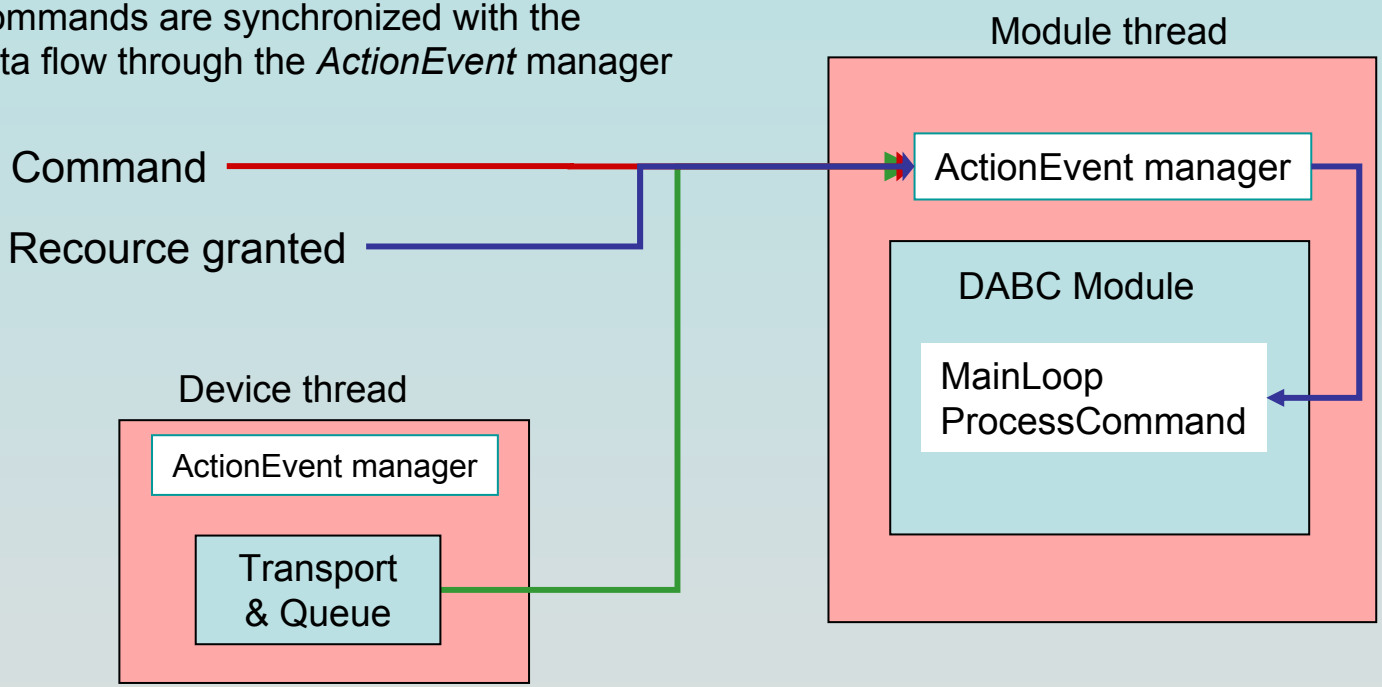
A *module* processes data of one or several data streams.
Data streams propagate through ***ports***, which are connected by ***transports*** and ***devices***



DABC design: Threads and synchronization

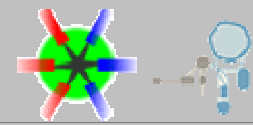


Commands are synchronized with the data flow through the *ActionEvent* manager

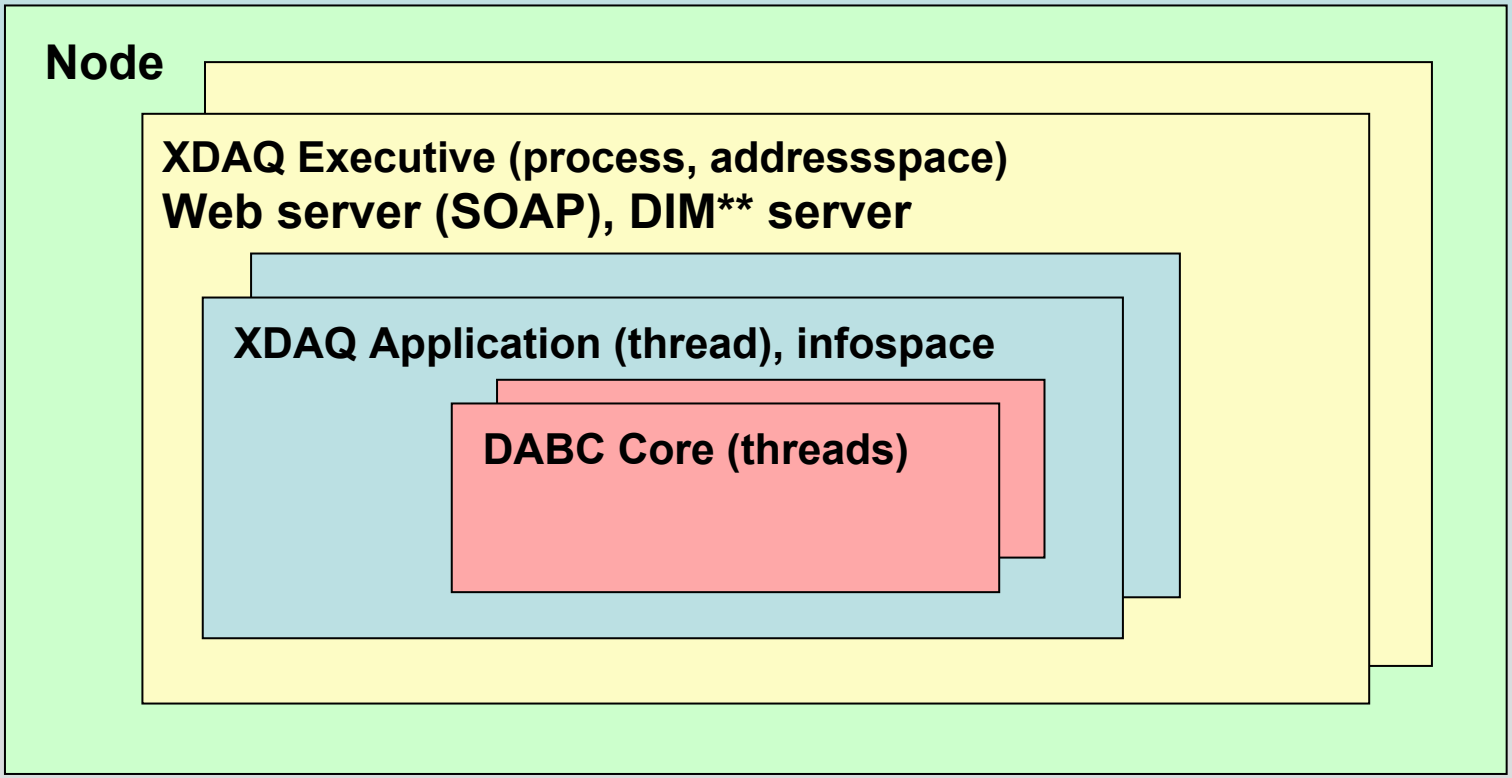


A device, i.e. socket device controls one port (transport).
Once a queue buffer is filled, the transport signals the *ActionEvent* manager, which in turn calls the *MainLoop* function of the associated module, **or**
If *MainLoop* was waiting for resource or buffer, it continues.

DABC design: XDAQ* Infrastructure



DABC core decoupled from XDAQ environment!

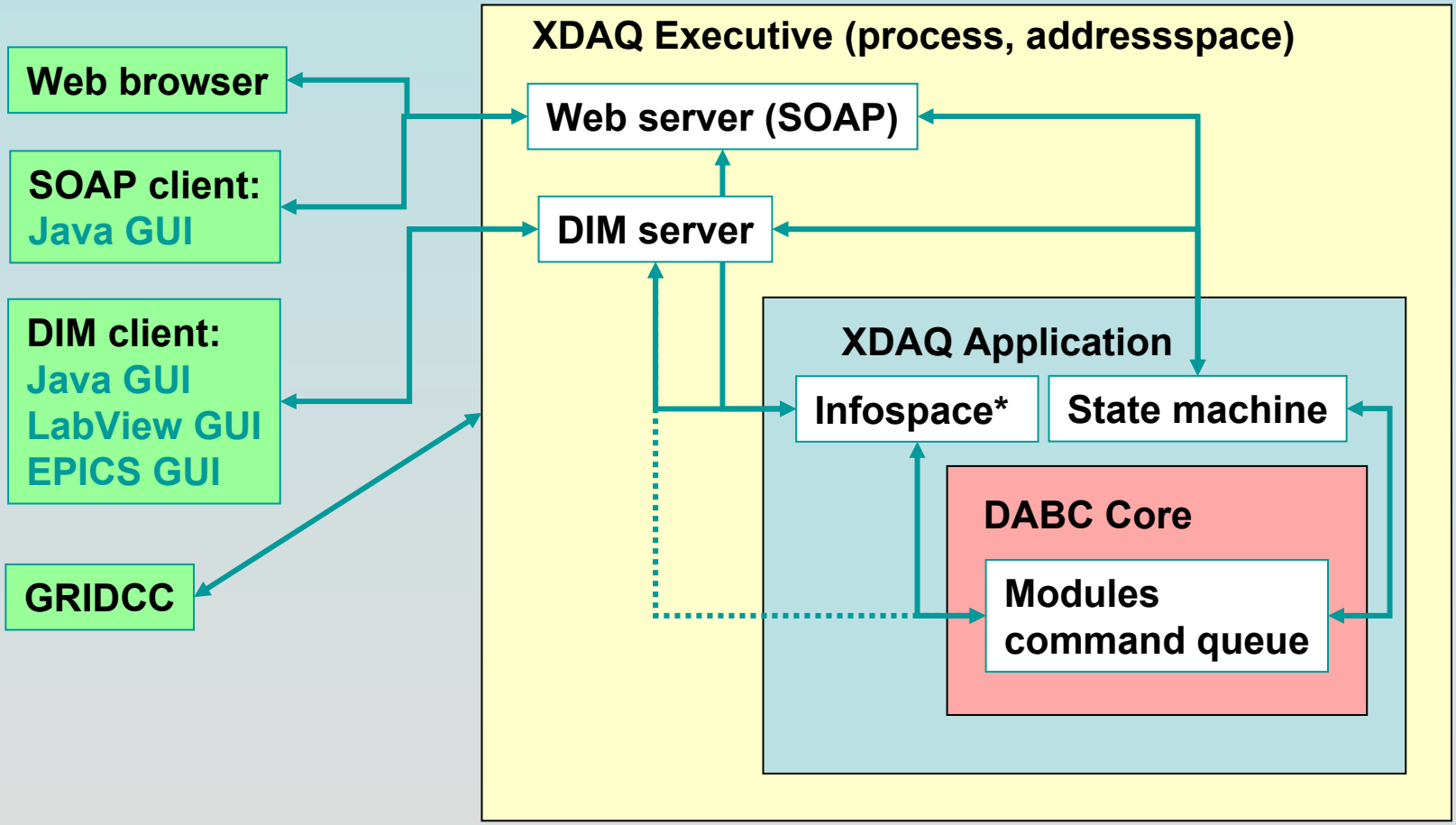


*** Standard DAQ framework for LHC CMS experiment**
Orsini, Gutleber http://xdaqwiki.cern.ch

** DIM: Distributed Information Management System, CERN

DABC design: Controls & monitoring communication

DABC core decoupled from XDAQ environment!

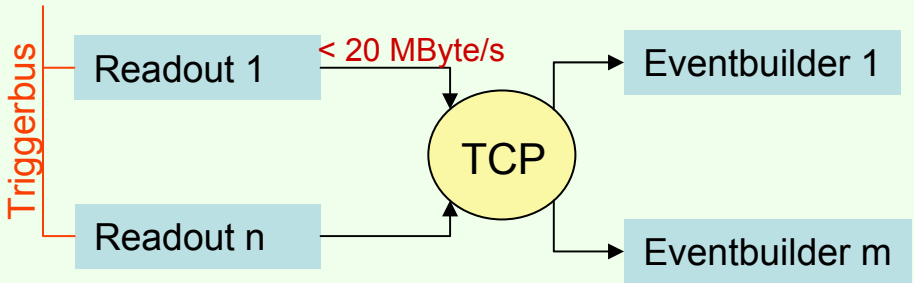


* Infospace: remotely accessible parameters

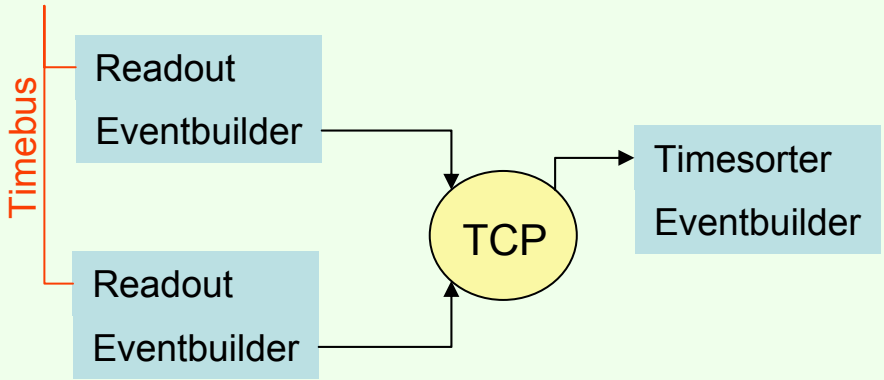
➢ MBS connection

MBS topologies

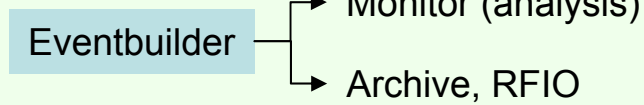
n x m multiple EBs

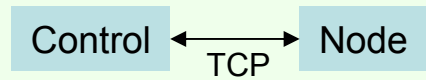


Time stamped (20 ns)

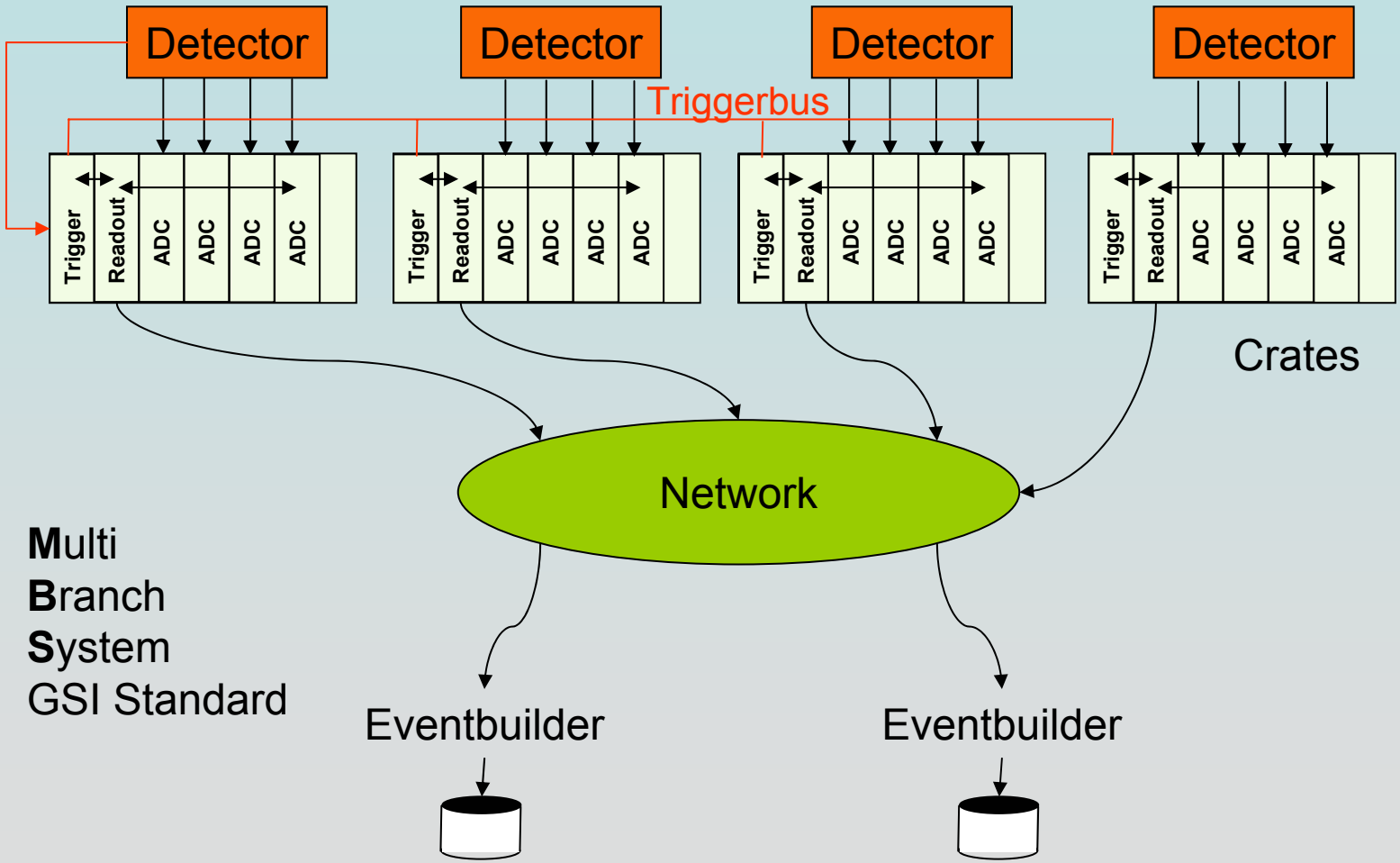


EB data transport

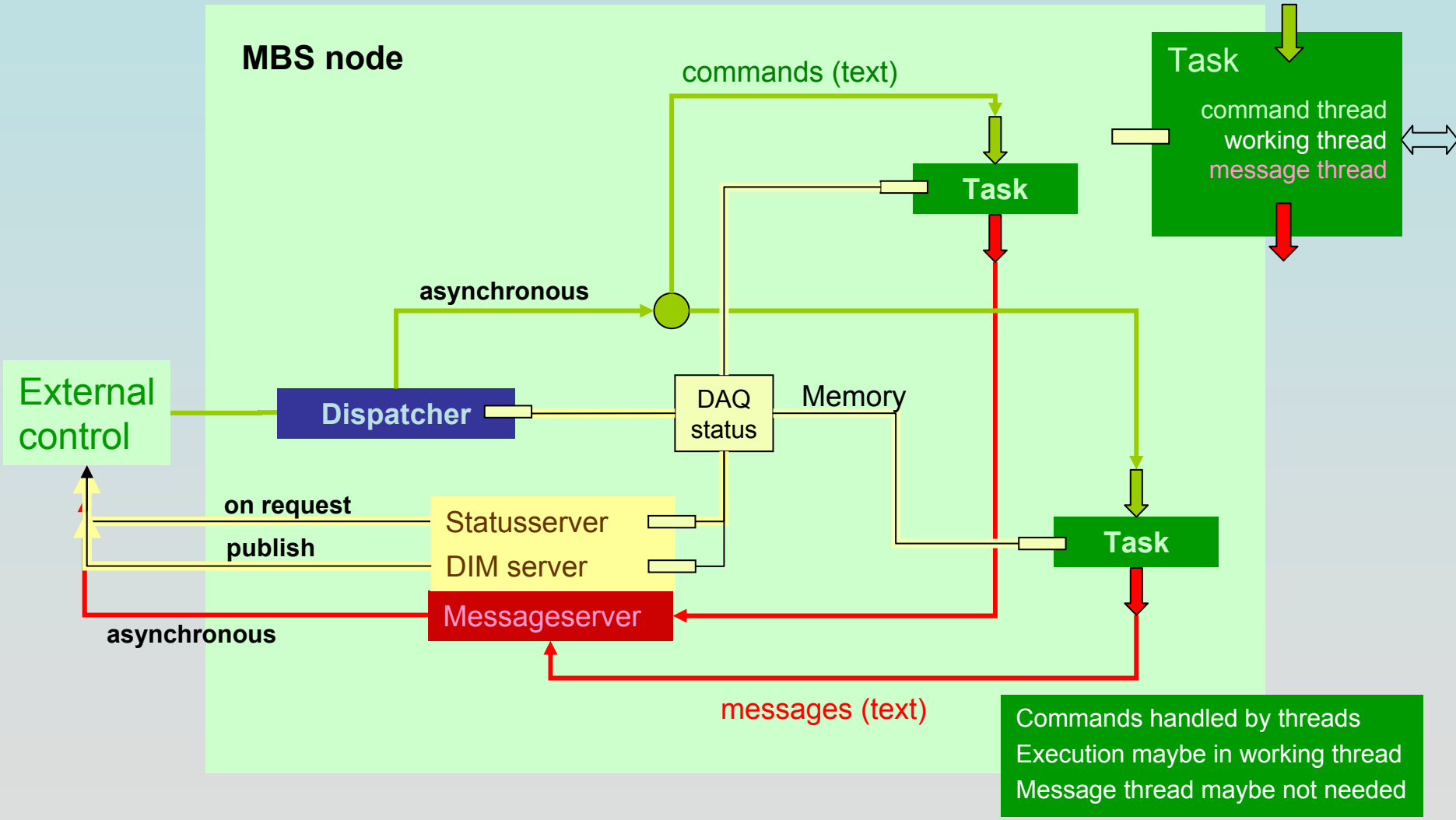




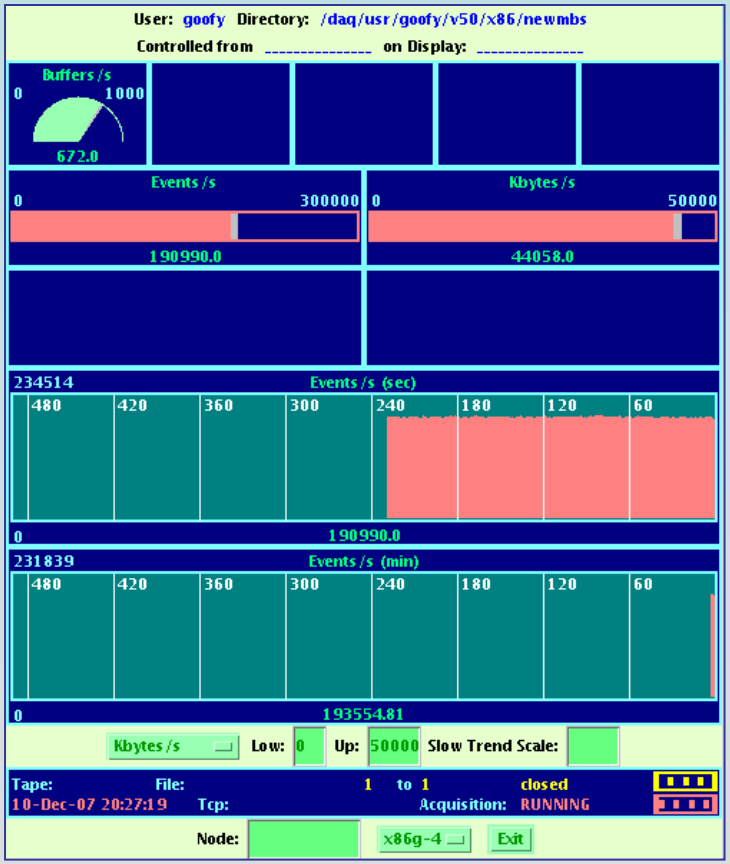
MBS example topology



MBS node control and monitor



MBS GUI (monitor)

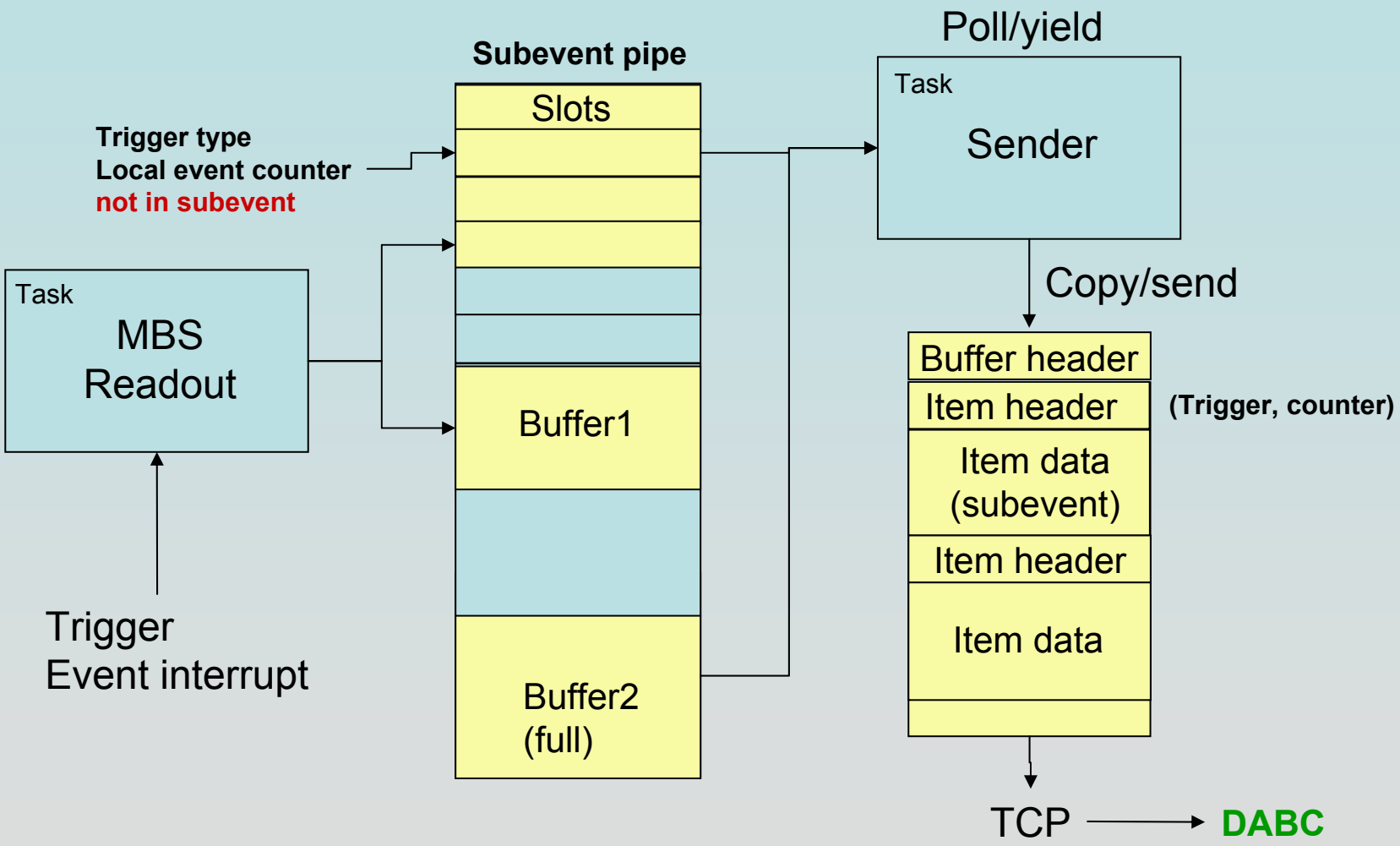


Counters		Rates /sec	
Events	67748699	Events	191850
Buffers	238551	Buffers	675
Streams	238551	Streams	675
Kbytes	15267264	Kbytes	44274
File Kb	0	File Kb	0
Tape Kb	0	Tape Kb	0
EventServ evt	0	EventServ evt	0
EventServ Kb	0	EventServ Kb	0
StreamServ buf	0	StreamServ buf	0
StreamServ Kb	0	StreamServ Kb	0

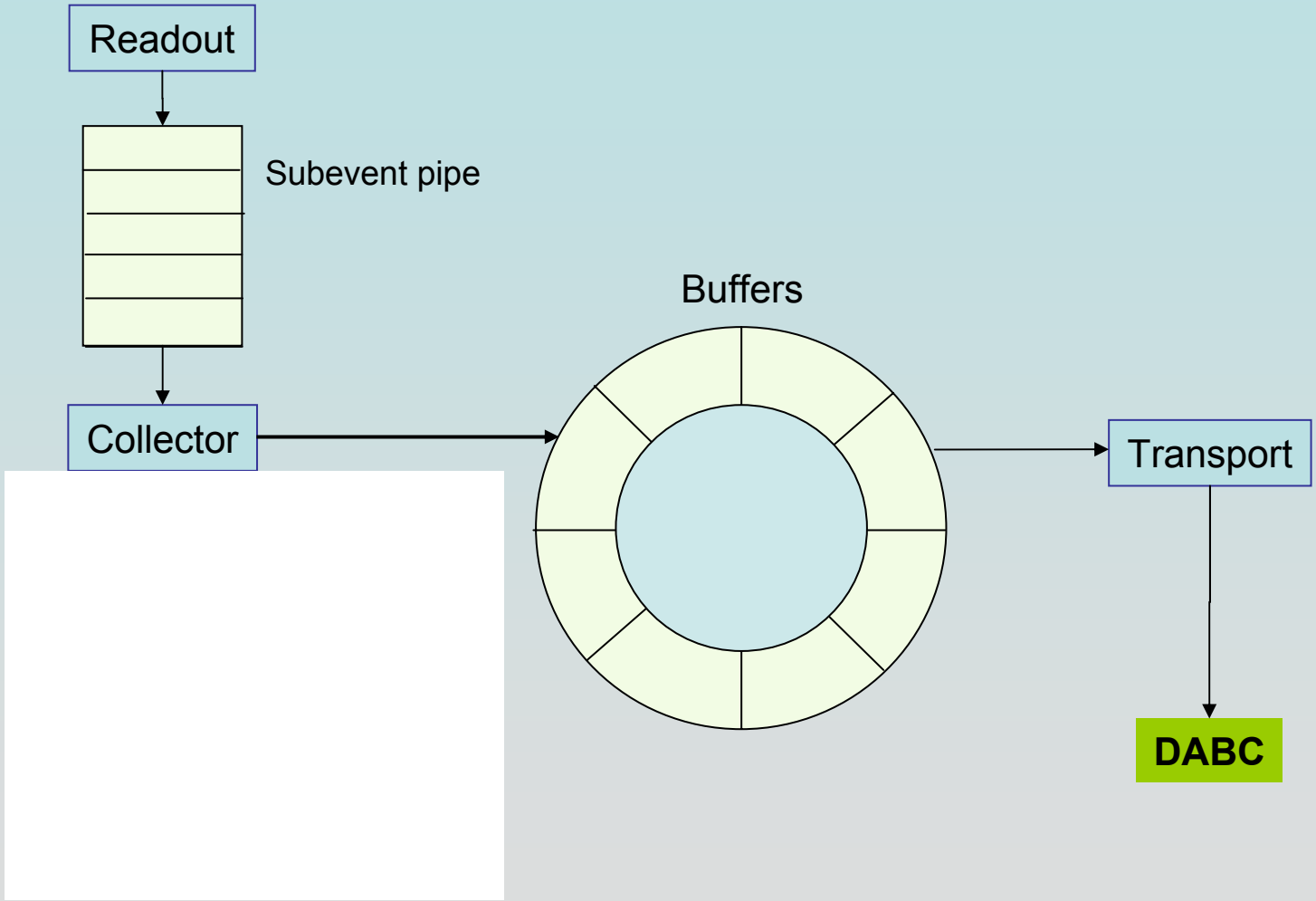
Task Table	
Prompter	Not running
Dispatcher	Not running
Message Logger	
Utilities	none
Transport	/daq/usr/goofy/mbswork/v51/bin_PCx86/m_util
Collector	Not running
Rate Meter	/daq/usr/goofy/mbswork/v51/bin_PCx86/m_msg_log
Read Camac	Not running
Read Meb	Not running
Event Server	Not running
Stream Server	Not running
Esone Server	Not running
Histogram Server	Not running
Fastbus SMI	Not running
SE sender	Not running
SE receiver	Not running
SE async receiver	Not running
Rising receiver	
Time sort receiver	Not running
Setup Table	Not loaded
MultiLayer Setup	Not loaded
Readout Table	Not loaded

Stop node survey: Stop

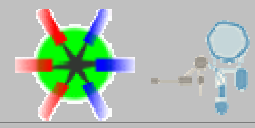
MBS data channel to DABC, first idea



MBS data channel to DABC, as implemented



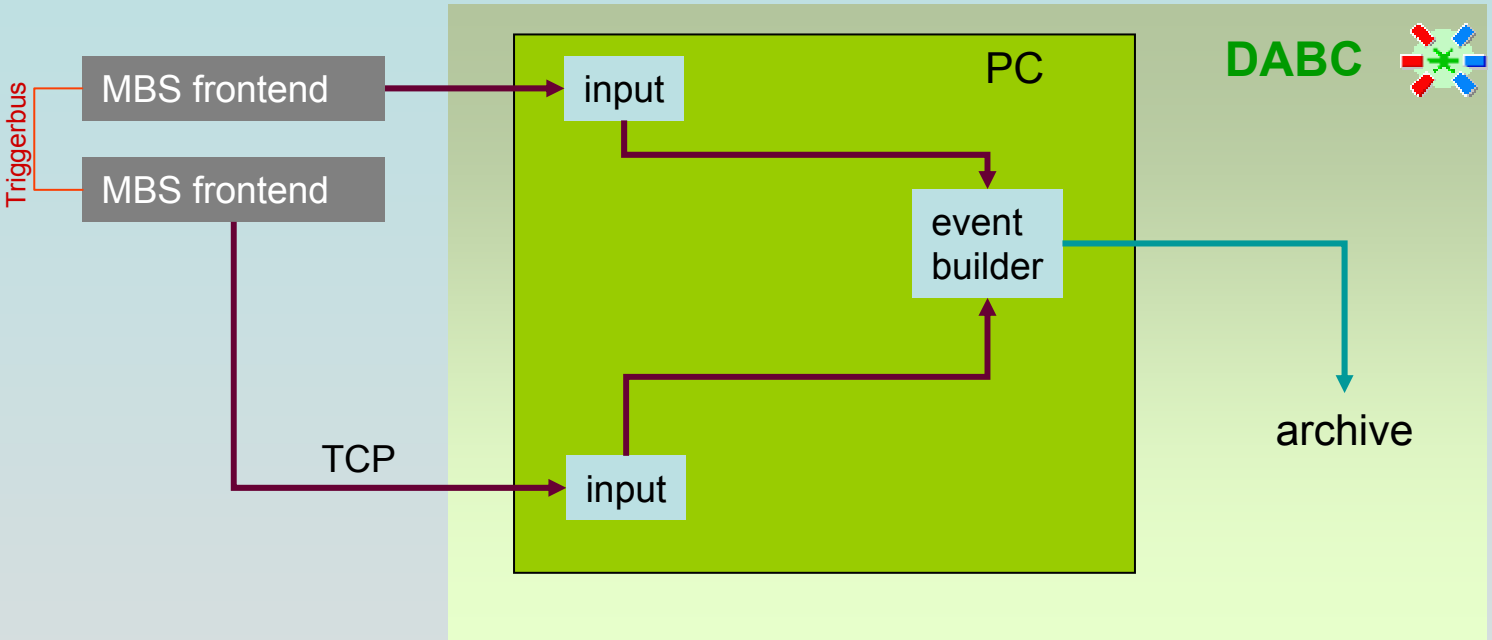
MBS Upgrade for DABC event builder



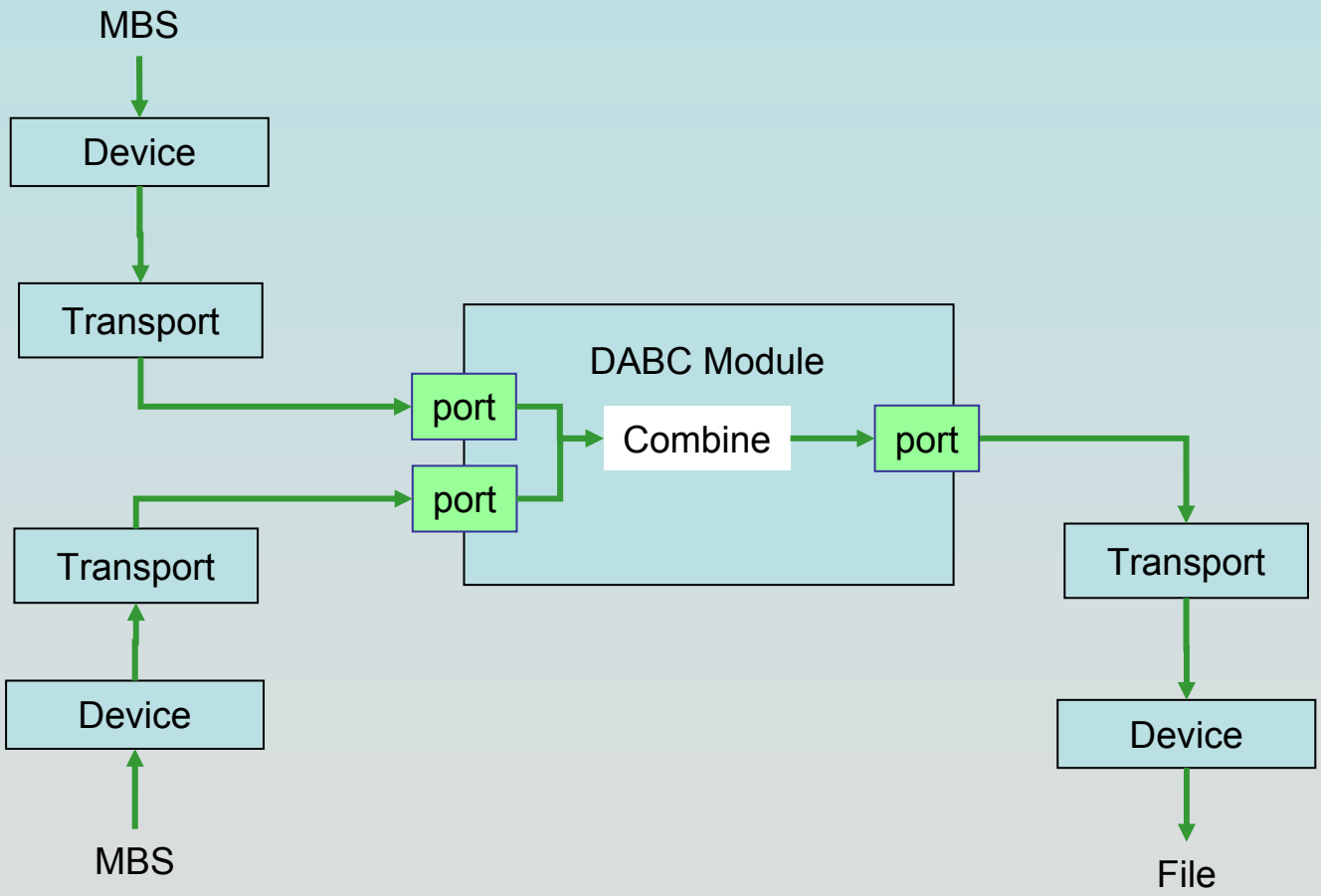
- **General upgrades**
  - ☑ Large buffers (up to now 32K limit) in MBS.
  - ☑ Large buffers in LMD files.
- **DABC specific mode**
  - ☑ MBS transport in DABC mode blocks, if no DABC is connected.
  - ☑ MBS transport sends variable sized buffers.
  - ☑ Using large buffers and one buffer per stream: no event spanning.
- **New LMD file format**
  - ☑ No buffer structure.
  - ☑ File header, data elements, index table (random access).
  - ☑ No size limit (> 2 GB).
  - ☑ Supported by event API.
- **To do**
  - Java based MBS GUI upgrade to integrate into DABC GUI (DIM)
  - Event server in DABC

➢ MBS application in DABC

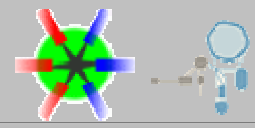
DABC: Local event building for MBS



DABC: Local event building for MBS



DABC application: Local Mbs event builder



DABC Device

MbsFrontend

MbsFrontend ()
Create commands
Register parameters

CreateTransport (port)
Create MbsRead

DABC Transport

MbsRead

MbsRead (port)
Connect MBS transport
Register port

GetBufferSize ()
Return buffer size

Read (buffer)
Read header
Read data
Return

~MbsRead ()
Disconnect MBS

Thread

DABC Module

MbsCombiner

MbsCombiner ()
Create commands
Register parameters
Create ports (n x input, 1x output)

MainLoop ()
Wait input port 1...n
Get empty buffer
Loop: Add subevent ref. to buffer
Size > max? Put buffer to port
Return
EndLoop

~MbsCombiner ()

Thread

DABC Device

MbsFile

MbsFile ()
Create commands
Register parameters

CreateTransport (port)
Create MbsStore

DABC Transport

MbsStore

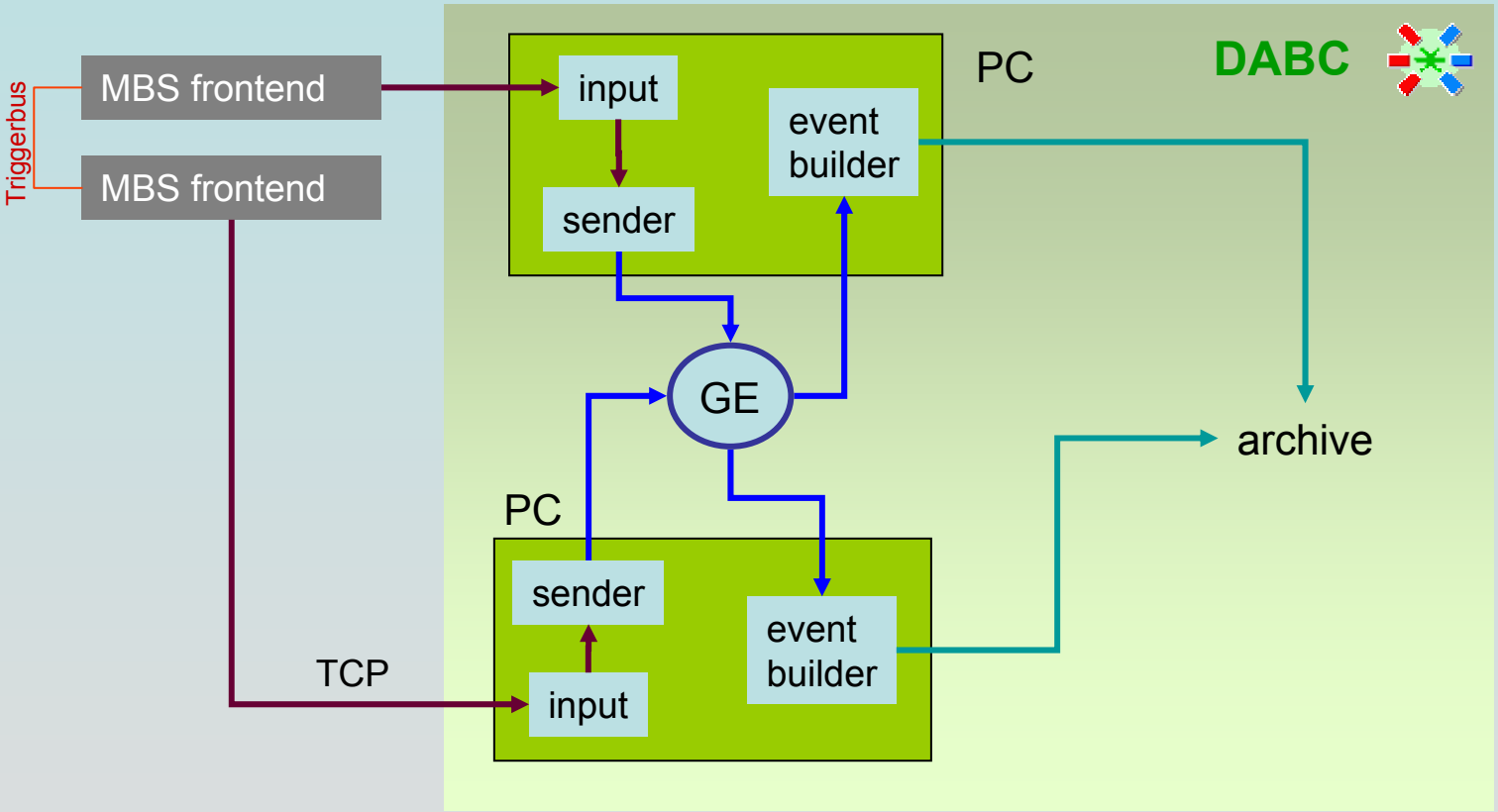
MbsStore (port)
Register port

Write (buffer)
Filesize > max?
close/open
Write fragments to file

~MbsStore ()
Close last file

Thread

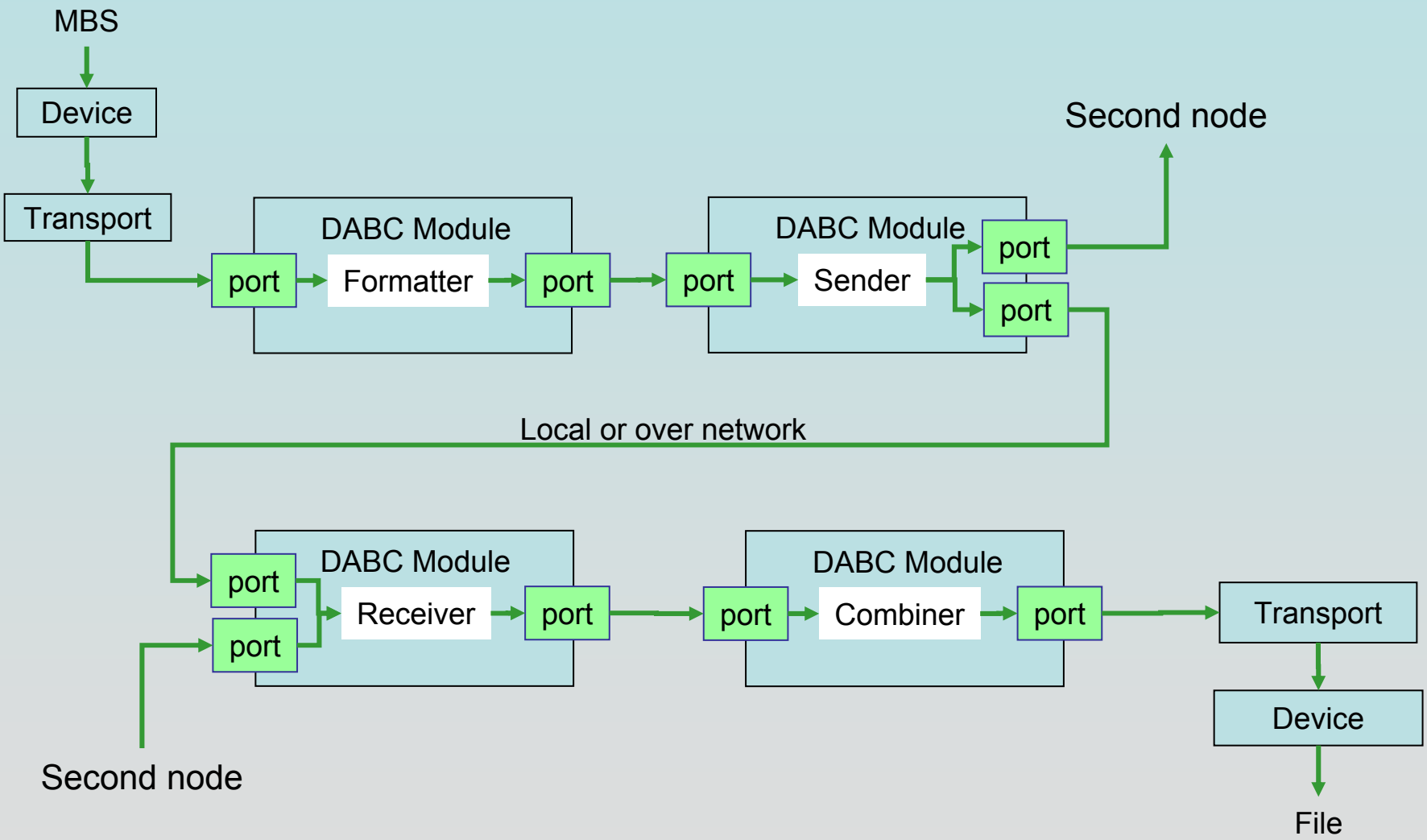
DABC: Network event building for MBS



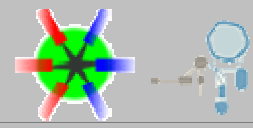
GE: Gigabit Ethernet

➢ DABC structure

DABC: Network event building for MBS



DABC application: Mbs event builder over network I



DABC Module

MbsCombiner

MbsCombiner ()
Create commands
Register parameters
Create ports (1 x input, 1x output)

MainLoop ()
Wait input port 1
Get empty buffer
Loop: Add data ref. to buffer
Size > max? Put buffer to port
Return
EndLoop

~MbsCombiner ()

Input as before

Thread

DABC Module

BNetSender

BNetSender ()
Create commands
Register parameters
Create ports (1 x input, n x output)

MainLoop ()
Wait input port 1
Put buffer to next output port

~BNetSender ()

Provided by DABC

Thread

DABC Device

BNetNetworkDevice

BNetNetworkDevice ()
Create commands
Register parameters

CreateTransport (port) ()
Create BNetNetworkTransport

DABC Transport

BNetNetworkTransport

NetNetworkTransport (port)
Register port

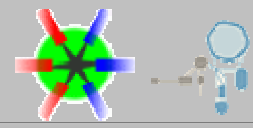
Write (buffer)
Send fragments

~BNetNetworkTransport ()

Provided by DABC

Thread

DABC application: Mbs event builder over network II



DABC Device

BNetNetworkDevice

BNetNetworkDevice ()
Create commands
Register parameters

CreateTransport (port)
Create BNetNetworkTransport

DABC Transport

BNetNetworkTransport

BNetNetworkTransport (port)
Register port

Read (buffer)
Receive fragments

~BNetNetworkTransport ()

Provided by DABC

Thread

DABC Module

BNetReceiver

BNetReceiver ()
Create commands
Register parameters
Create ports (n x input, 1 x output)

MainLoop ()
Wait for all input ports
Add input buffers to output list
Put list to output port

~BNetReceiver ()

Provided by DABC

Thread

DABC Module

MbsEventBuilder

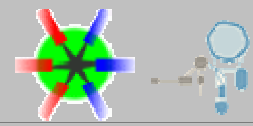
MbsEventBuilder ()
Create commands
Register parameters
Create ports (1 x input, 1x output)

MainLoop ()
Wait input port 1
Get empty buffer
Build event from input list
Put event to buffer
Put buffer to port (file store)

~MbsEventBuilder ()

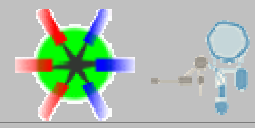
Thread

➢ DABC setup



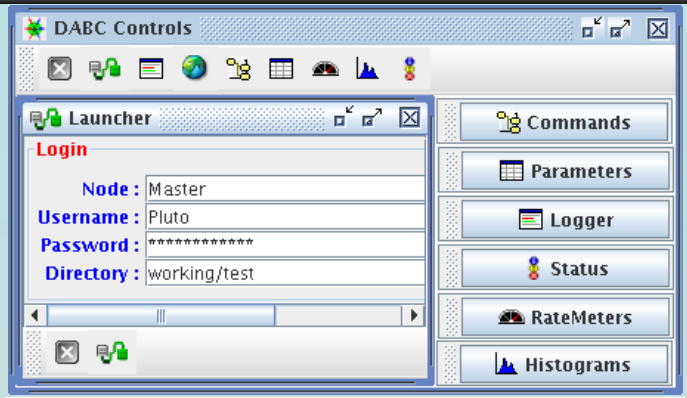
- **Configuration via XML files**
- ***ApplicationPlugins* (entry point to application libraries)**
  - Call application factories
- **Application factory classes**
  - CreateDevice
  - CreateModule
  - Device->CreateTransport (Module->GetPort)
  - ConnectPorts (Module1->GetPort, Module2->GetPort)
  - CreateMemoryPool
- **State commands**
  - Startup
  - Initialize
  - Run / Stop
  - Hold / Resume
  - Shutdown

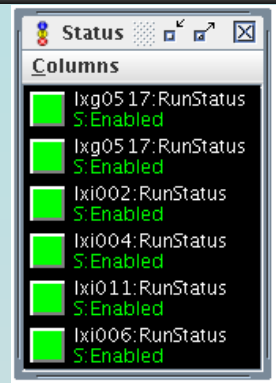
DABC: Commands and parameters

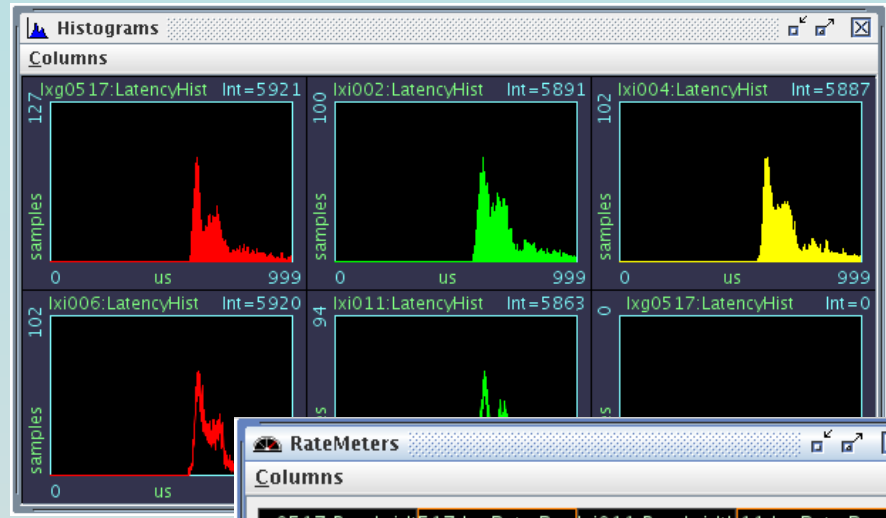


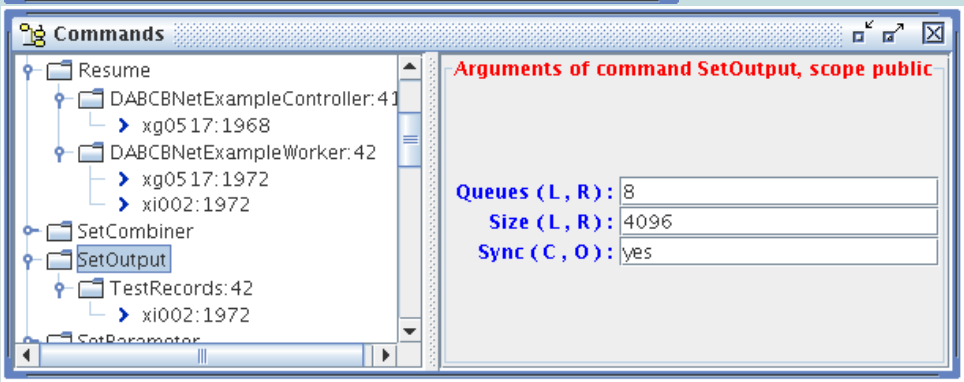
- **Commands**
  Objects with command description (XML) and *ProcessCommand* function.
  Name string: **/ server / node / application / type.thread.name**
  - server: DIM namespace
  - node: name:ID (port)
  - application: namespace::name:ID
  - type: DEV, MOD, POOL, PLUG
  - thread: name of module or device or...
  - name: command (description by related parameter record)
- **Parameters**
  Same name structure as above
- **Parameter records**
  Recognized by GUI, graphical presentation
  - Status
  - Rate
  - Histogram
  - Command description
  - and more

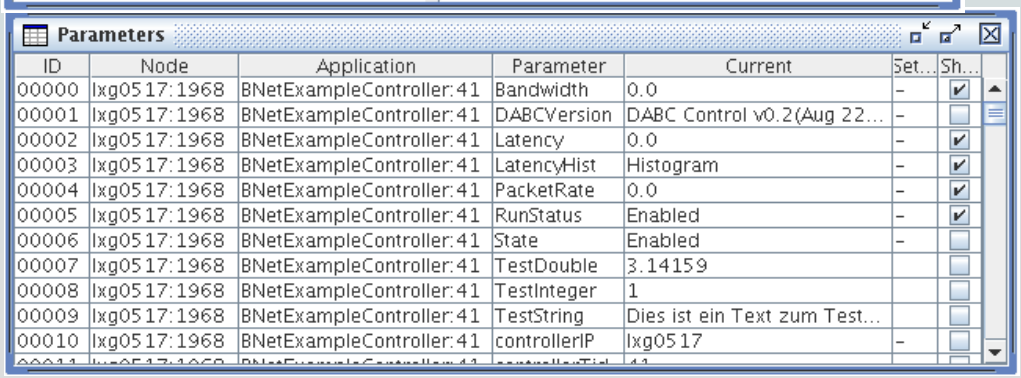
Java GUI DIM based, run time built from services





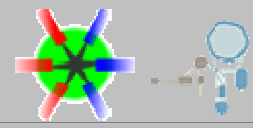






ID	Node	Application	Parameter	Current	Set...	Sh...
00000	lxg0517:1968	BNetExampleController:41	Bandwidth	0.0	-	✓
00001	lxg0517:1968	BNetExampleController:41	DABCVersion	DABC Control v0.2(Aug 22...	-	
00002	lxg0517:1968	BNetExampleController:41	Latency	0.0	-	✓
00003	lxg0517:1968	BNetExampleController:41	LatencyHist	Histogram	-	✓
00004	lxg0517:1968	BNetExampleController:41	PacketRate	0.0	-	✓
00005	lxg0517:1968	BNetExampleController:41	RunStatus	Enabled	-	✓
00006	lxg0517:1968	BNetExampleController:41	State	Enabled	-	
00007	lxg0517:1968	BNetExampleController:41	TestDouble	3.14159		
00008	lxg0517:1968	BNetExampleController:41	TestInteger	1		
00009	lxg0517:1968	BNetExampleController:41	TestString	Dies ist ein Text zum Test...		
00010	lxg0517:1968	BNetExampleController:41	controllerIP	lxg0517	-	

DABC further tasks



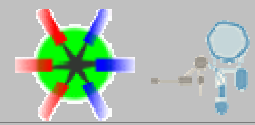
Achieved

- Infrastructure
- Data flow engine
- Very first (0.1) Java GUI
- PCI support
- MBS event building
- Final Programming Interface definitions

Todo

- Controls, GUI
- Data formats
- Combiner/Time sort
- Event building
- Final Programming Interface implementation
- Final packaging
- MBS GUI
- Documentation

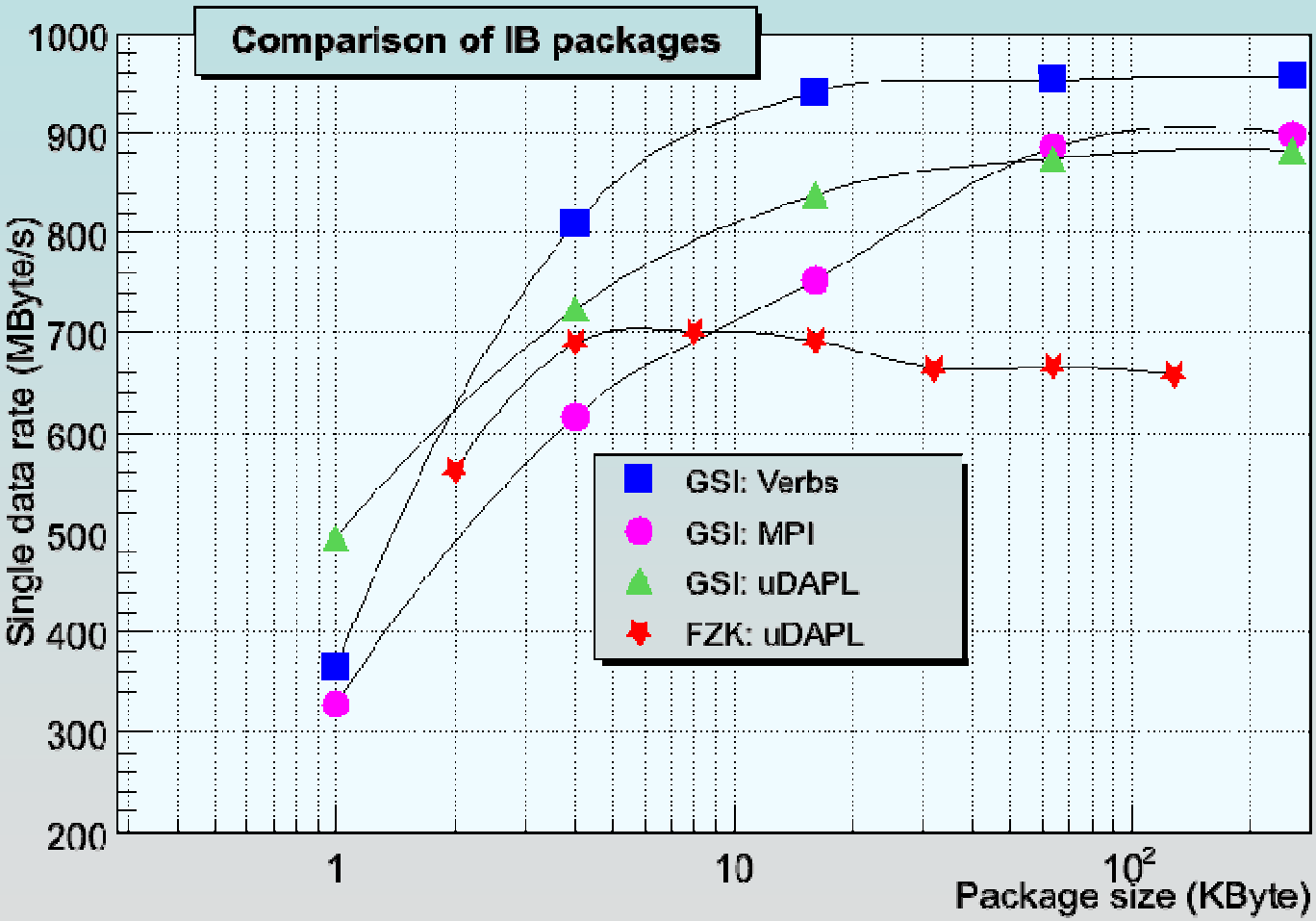
Resources and time



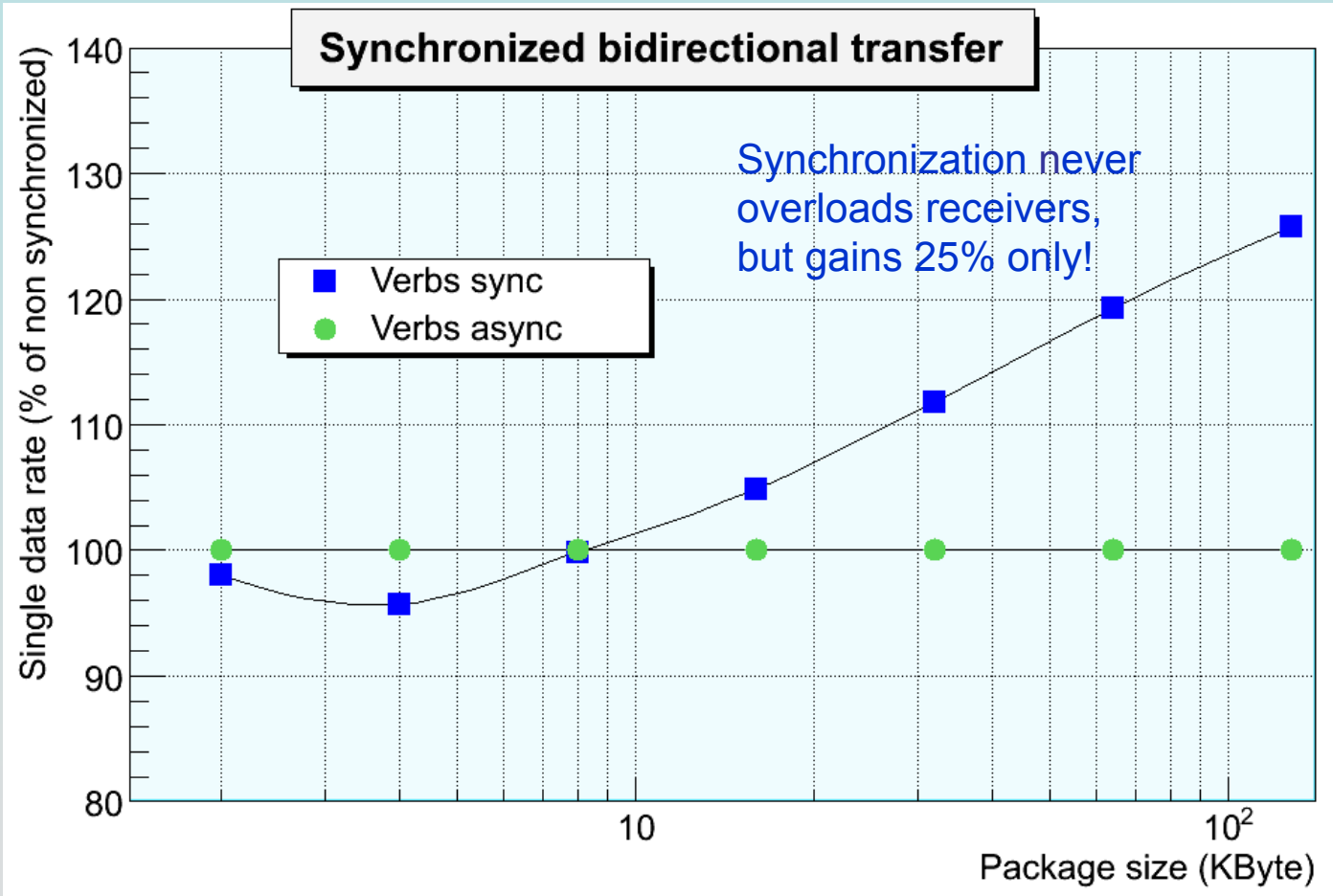
- People of data processing group
  H.G.Essel
  J.Adamczewski
  N.Kurz
  S.Linev
- People of controls group
  maybe one FTE
- People from CBM
  hopefully

- **CBM requires in 2008 a data taking system**
  Start with small system, grow on demand
  Preliminary controls
- **NUSTAR ?**
  In discussion
- **MBS** a first test bed

Network test performance 4x4 bidirectional

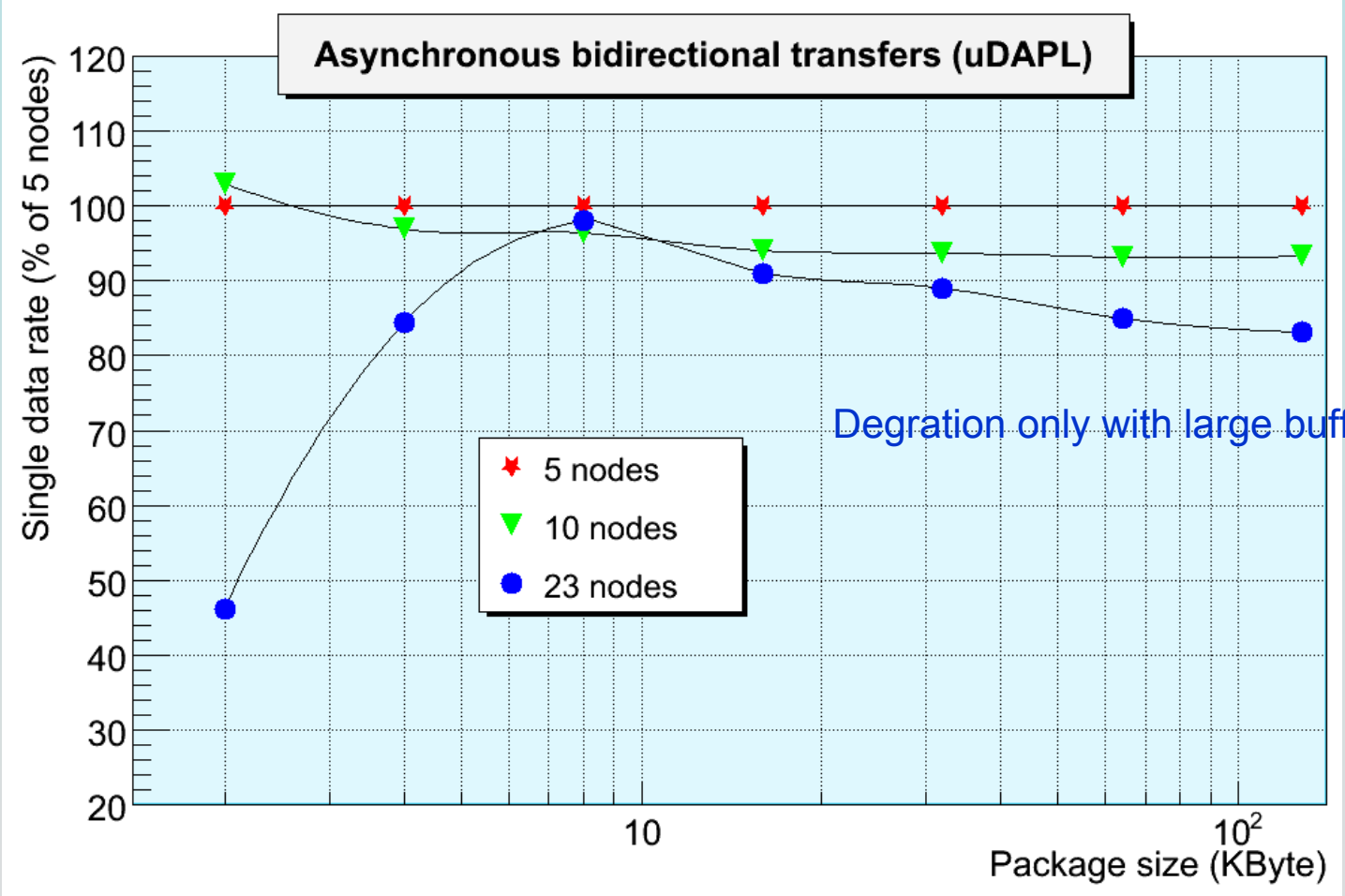


Optimized transfer (23 nodes at FZK), all to all



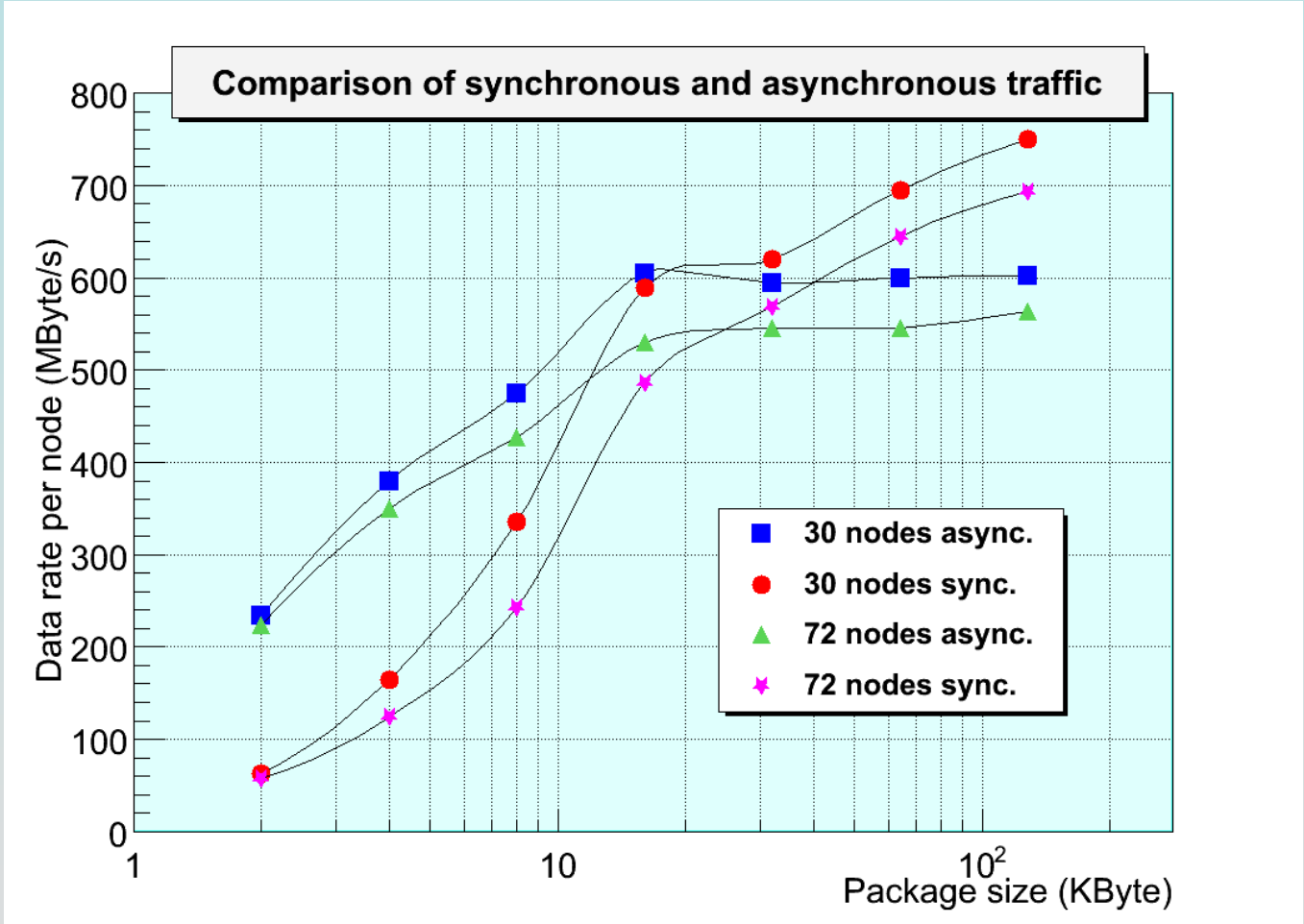
We thank Frank Schmitz, Ivan Kondov and Project CampusGrid at the Institute for Scientific Computing, Forschungszentrum Karlsruhe, for providing resources and support for the large-scale measurements.

Scaling of performance (FZK)



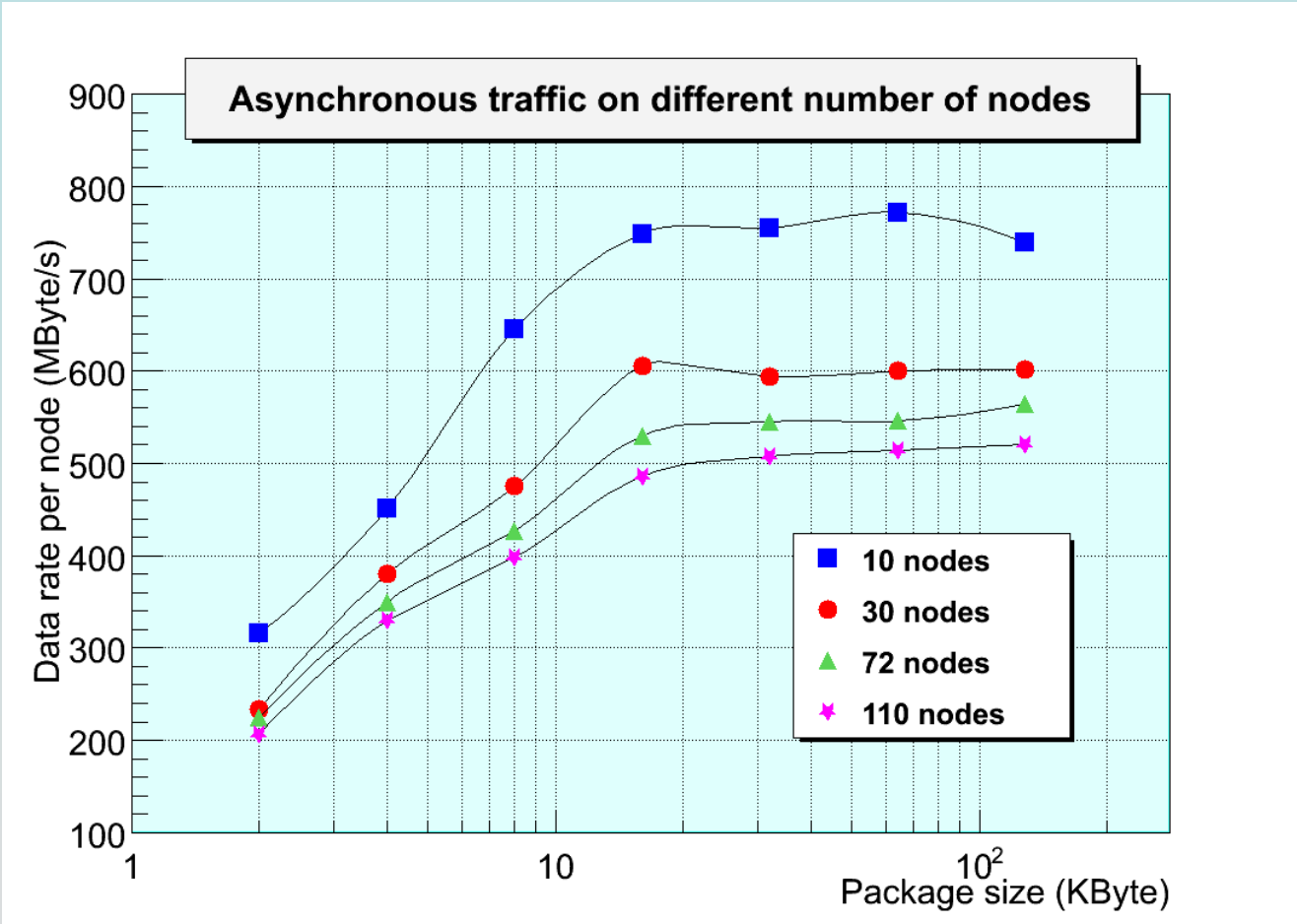
We thank Frank Schmitz, Ivan Kondov and Project CampusGrid at the Institute for Scientific Computing, Forschungszentrum Karlsruhe, for providing resources and support for the large-scale measurements.

Scaling of performance (Mainz)



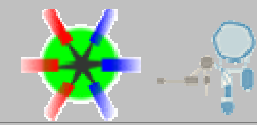
We thank Klaus Merle and Markus Tacke at the
Zentrum für Datenverarbeitung der Johannes Gutenberg Universität, Mainz,
for providing resources and support for the large-scale measurements.

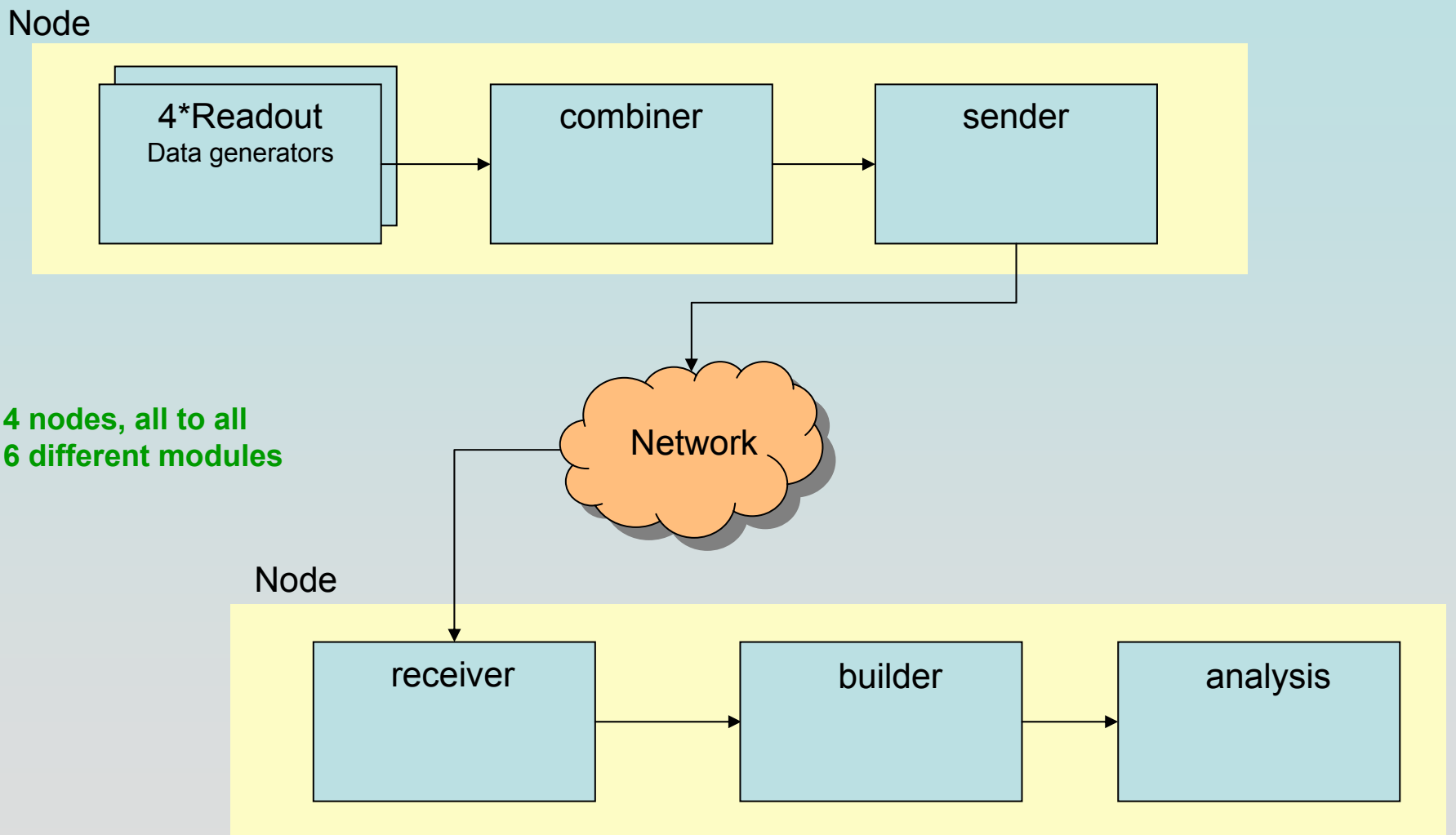
Scaling of performance (Mainz)



We thank Klaus Merle and Markus Tacke at the
Zentrum für Datenverarbeitung der Johannes Gutenberg Universität, Mainz,
for providing resources and support for the large-scale measurements.

DABC: Full event building example measurement





Dataflow engine and event building rate