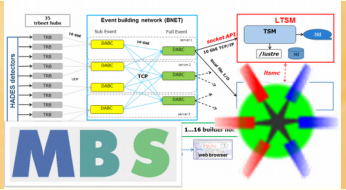


MBS release V 7.0

EEL meeting 07.03.2023

Jörn Adamczewski-Musch, Nikolaus Kurz, GSI/EEL

07.March 2023

MBS history and status

- MBS is general purpose DAQ system from GSI for 30 years
- Recent releases: V4.3 (2004), V5.1 (2010), V6.3 (2017) [5]
- Mostly used readout systems:
  - VMEbus + modules (GSI and other)
  - PCIe + optical fibre (gosip) + GSI front-end boards
- See Niks overview talk of 2019 (at daq.gsi.de)
- **New release 7.0 ready in January 2023 – Release notes**

Release 7.0

- Installed at GSI MBS cluster at `/mbs/v70`
- Default MBS version (*mbslogin prod*):
  please rebuild user readouts (*make clean; make;*)!!
- V6.3 is still available as *mbslogin old*
- Code repository now at git.gsi.de (restricted access)

New features

- New platforms:
  - Debian 11 “bullseye” (X86 PCs)
  - QorIQ ”jethro” Linux (IFC VME processor)
- Integration of mesytec MVLC VME controller
- White Rabbit timing receiver updates
- New GSI mass storage interface FSQ
- Miscellaneous

New platform: Debian 11 on x86 PC

- Kernel modules and driver software:
  - kinpex/trixor PCIe readout (libmbspex, Qt control GUIs)
  - pexaria White Rabbit timing receiver
- DABC installation (MBS webgui, rgoc command server)
- Go4 installation (`. /analysis/bin/go4login`)

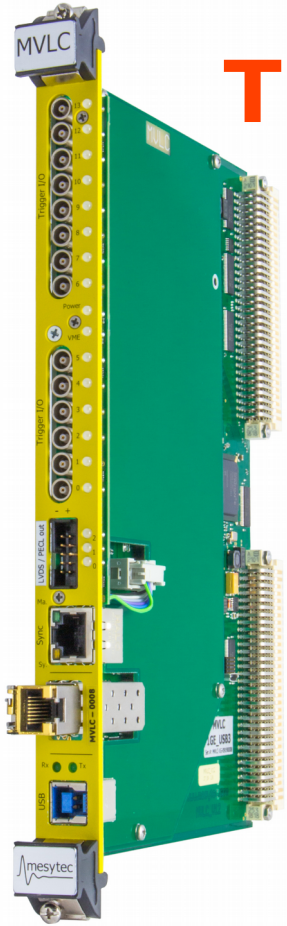
Please upgrade your machine!

New platform: IFC PPC linux for VME

- **CES/RIO4 processor discontinued in 2021** (in use with >80 modules)!
- First try: IoxOS IPV processor
  - Ready integrated to MBS V 7.0!
  - But IoxOS also ceased support end of 2021!
- New hope: IoxOS IFC [1]
  - Freescale PPC P2020
  - QorIQ "jethro" Linux

New platform: IFC PPC linux for VME

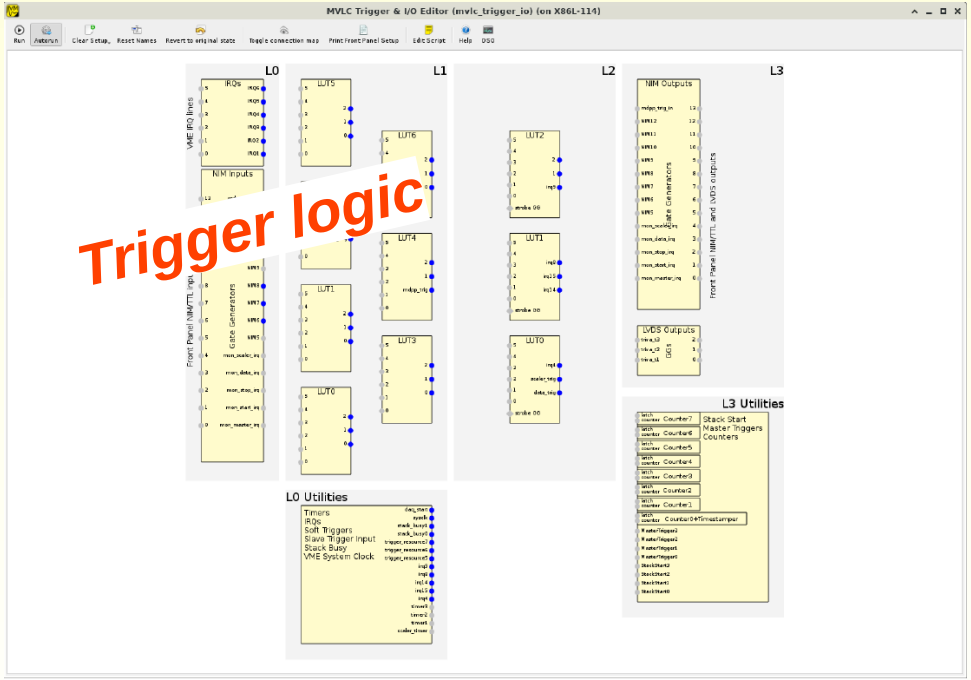
- Kernel modules and driver software
  - TRIVA trigger module
  - VETAR white rabbit timing receiver + etherbone libs
  - MBS pipe shared memory
- Fast direct mapping of VME addresses
- Example readouts and tests for all known VME modules

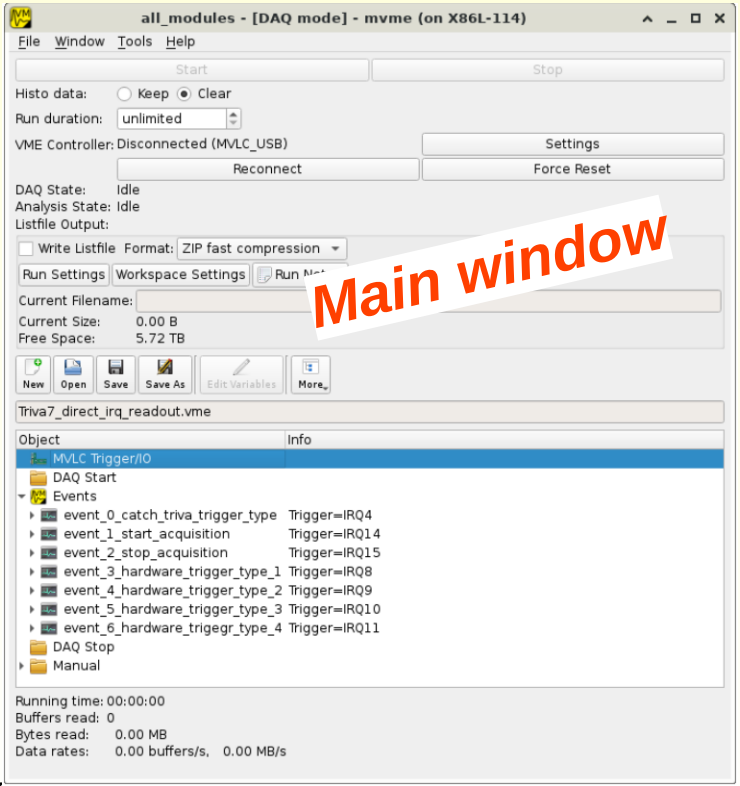


The mesytec VME controller

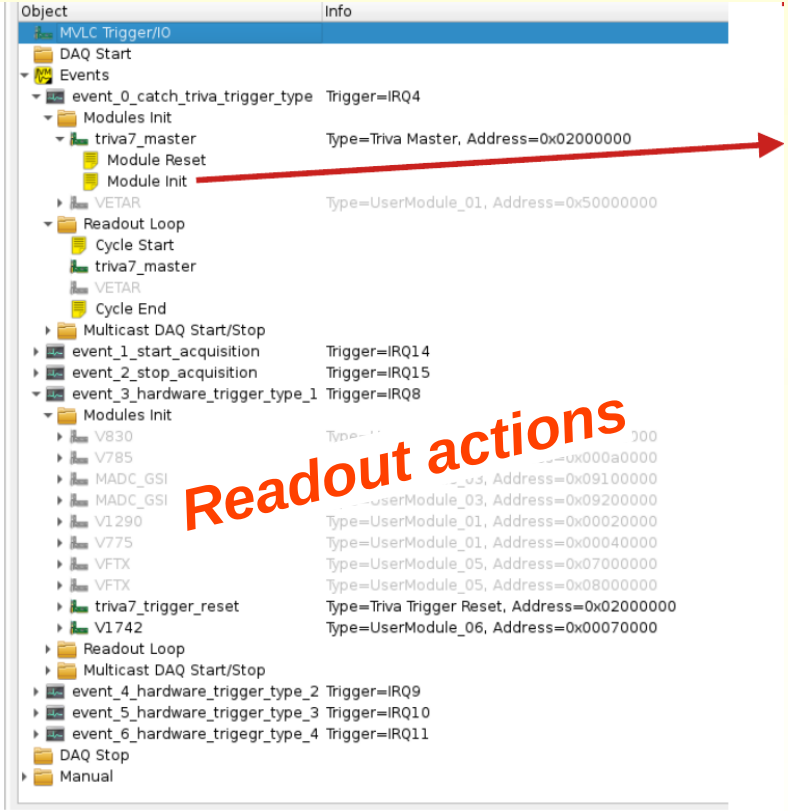
- MVLC:
  - VME controller/sequencer with **low (no) latency**
  - Reduced readout methods compared to VME processor
  - **Control and data transfer via USB 3** or ethernet (GSI: USB)
  - **Host can be "any" commodity MBS PC (X86L-*)**
  - Free programmable I/O
  - Powerful trigger logic (not mandatorily required at GSI)
- MVME:
  - Data acquisition and "analysis" software (C++)
  - **Tool to prepare initialization and readout sequences**
  - Graphical tools to program **trigger logic**

MVME software

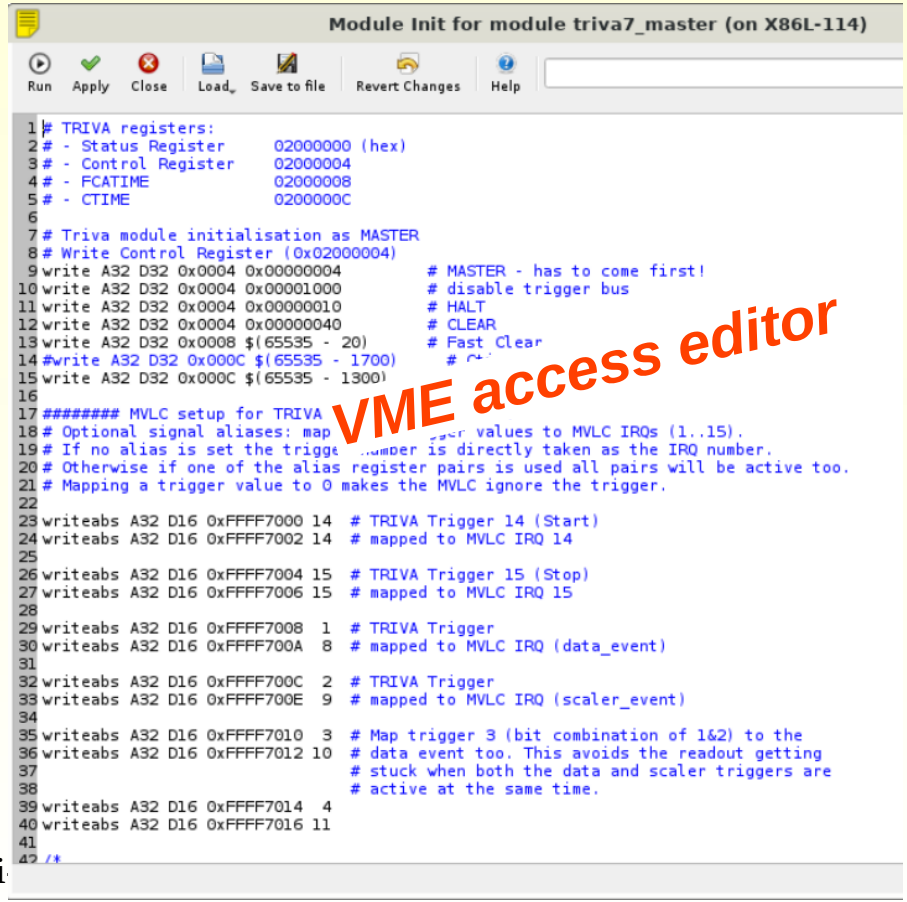




MVME software



Readout actions

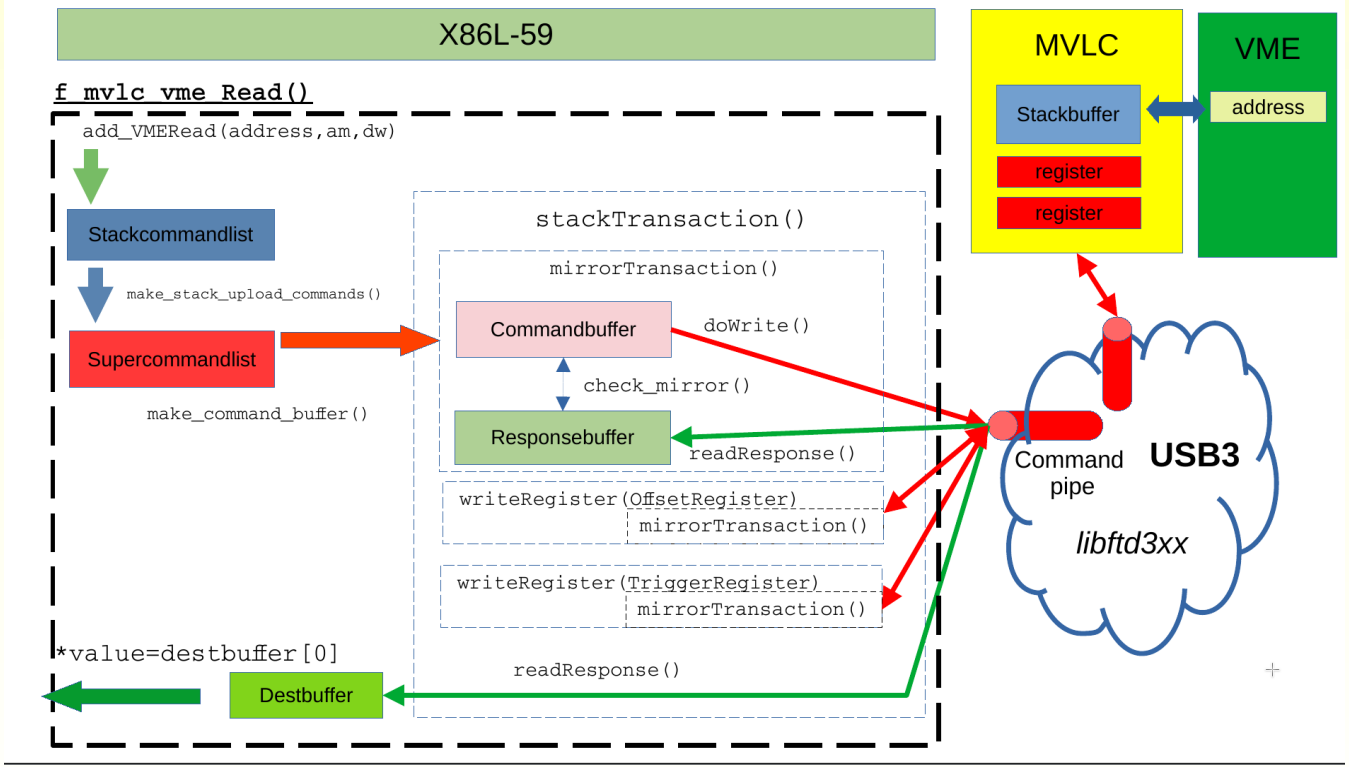


VME access editor

Integration of MVLC to MBS

- **Goal: event building of multiple branches with mvlc and non-mvlc readout!**
- To do:
  - ***Control of TRIVA with MBS commands*** (set trigmod, start acquisition,...)
  - Different readout code for different TRIVA trigger types (mvme + mesytec upgrades! **OK**)
  - ***MVLC USB data is received by MBS process*** (regular MBS subevents to pipe buffer)
  - Readout of ***VETAR White Rabbit timing receiver*** (produce MBS time stamp format)
  - Adjust readout for all types of VME modules used at GSI (mvme editor, **OK**)

MVLC software: VME read transcaction



MVLC software integration

- Re-implemented mvme C++ functions into **MBS C library (*lib_mvme.a*)**
- **New MBS readout process *m_read_mvme***
  - Used instead common readout *m_read_meb*
  - Access to **TRIVA registers** via USB/mvme
  - Reads **mvlc data stream** from USB
  - **Formatting of mbs subevents** for pipe
  - Optionally formatting of **VETAR White Rabbit timestamps**
  - !No user defined readout function – **user readout to be defined with mvme tools**
- New **linux kernel module *mbsmem.ko*** for host PC (shared memory pipe)
- New **command line tool *mec*** (pendant to *goc*): VME memory access via USB/mvlc

MVLC with MBS “how to”

1. Configure MVLC with MVME GUI tool.
   - It is available as alias “mvme” on all MBS PC hosts for v7.0.
   - Example readout stacks for all VME modules are provided by EEL/DAQ group
2. Exit MVME tool. DAQ setup will remain in MVLC FPGA until power down.
3. Startup MBS with settings:
   - In setup.usf: MASTER=19
   - in startup.scom: m_read_mvme instead m_read_meb.
   - Command “START MVLC” to start acquisition.

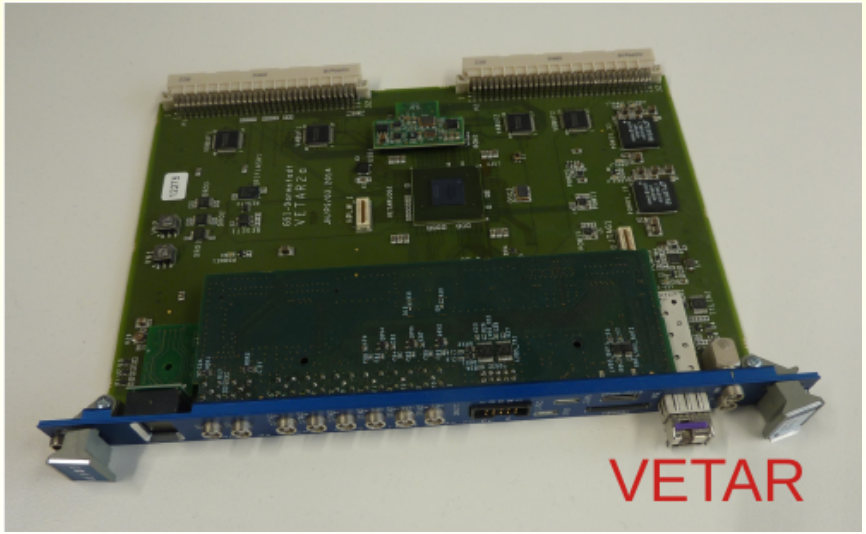
Summary MVLC with MBS

- All VME modules used at GSI proved to work
- Performance improvement of factor 3 compared with RIO4 readout for a typical FRS setup
- Performance improvement of factor 5 compared with RIO4 for 2eSST readout of CAEN v1742 (150 MB/sec)
- Switching from pure MVME to MBS DAQ in a few seconds (Use *mvme* to load MVLC sequences for MBS)
- No other hardware changes required when replacing RIO4 with MVLC
- MBS with MVLC controller works as standalone, master or slave TRIVA trigger MBS sub(system)
- No multi-event readout required for performance reasons

White Rabbit timing receivers

Read **FAIR time stamp** (<4 ns resolution) latched at the moment **of accepted trigger** into event data

Used for **synchronization / time sorting between "free running" (aka streaming readout) branches**



Updates for White Rabbit with MBS

- Readout of VETAR: **for IPV, IFC, MVLC** (see next slide!)
- Support of **FAIR timing release *fallout*** [4]**:**
  - Software installations on MBS X86_64 Linux (etherbone, saftlib) for debian9, debian 11
  - Firmware deployed and tested on PEXARIA and VETAR
- Support of **new time latch unit ECA_TLU**:
  - Old TLU (GSI TM LATCH V2) deprecated; has bugs in latest fallout firmware
  - Direct access readout of ECA_TLU has been demonstrated; fixes problems
  - **Requires initial configuration with higher level saftlib!** (automatically at boot time)

VETAR readout with MVLC

- **MVLC has no OS** => no wishbone.ko, vetar.ko kernel modules! No etherbone libs.
- Must use direct TLU register access
- Configuration of VETAR implemented with stack code in MVME tool => **"kernel driverless" mode (also useful for other platforms...)**
- Readout of VETAR timestamp by default with TRIVA stack frame
- Formatting of "MBS timestamps" from TLU words done in *m_read_mvme*
  (no *f_user.c* with common VETAR readout code!)

New mass storage interface FSQ

- "**F**ile **S**torage **Q**ueue" (Thomas Stibor, GSI IT [6])
- **FSQ server** with access to */lustre* and TSM
- DAQ writes data with **lightweight FSQ protocol**, no IBM libs required
- Files are copied to **/lustre first, then archived to tape**
- Developed and **tested with HADES at beamtime FEB22**
- Will be **common GSI storage interface for experiments**

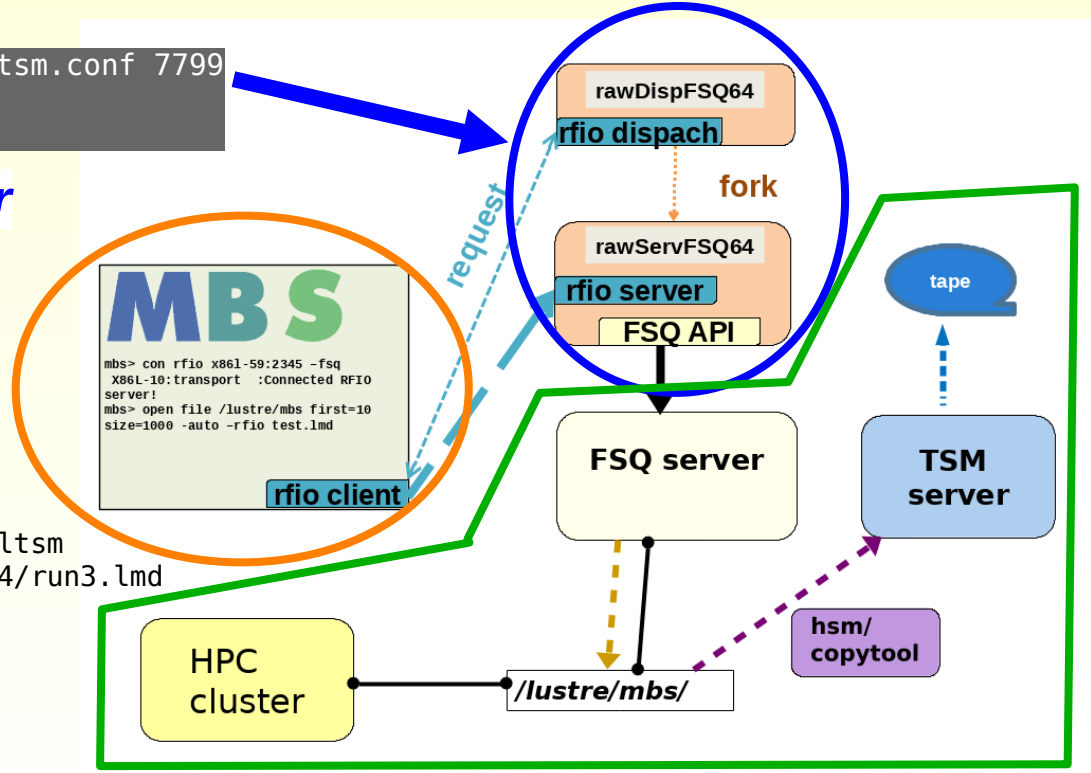
MBS with RFIO2FSQ gateway

```
adamczew@x86L-59: rawDispFSQ64 -c ./myltsm.conf 7799
```

experiment's server

MBS event builder

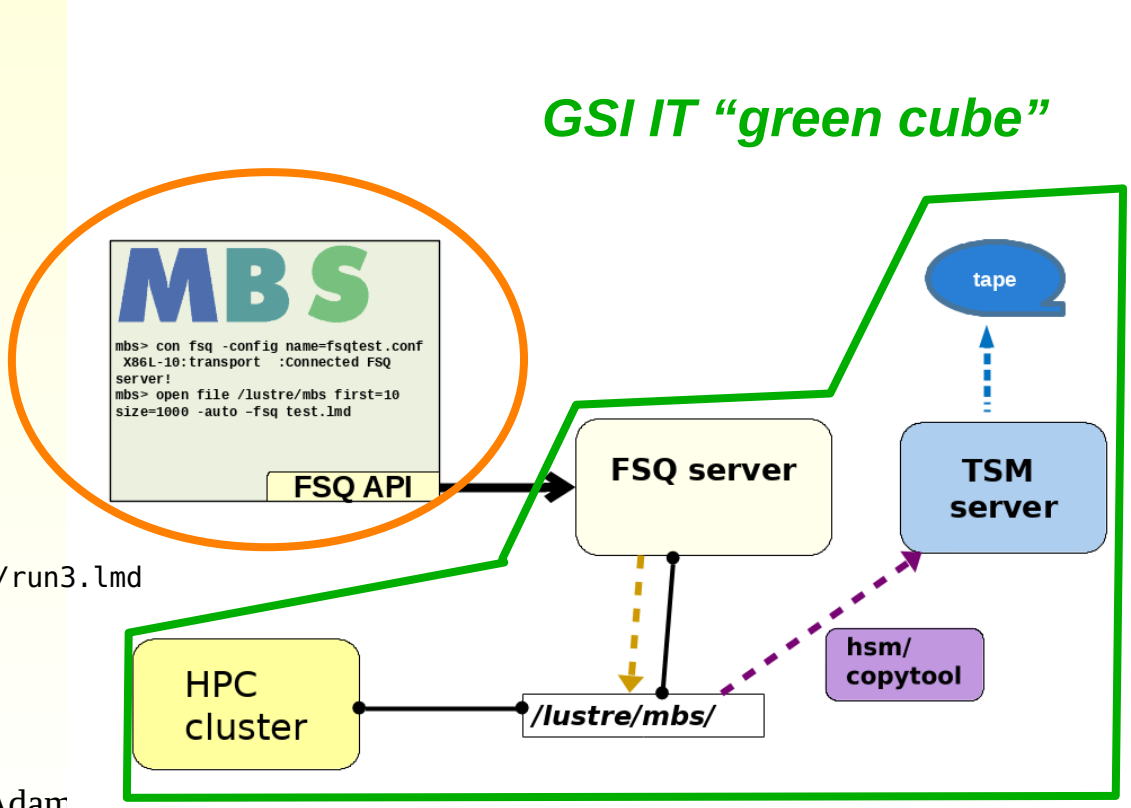
```
MBS> connect rfio < servername >:< port > -ltsm
MBS> open file -rfio -auto /lustre/exp/mar24/run3.lmd
```



MBS transport to FSQ server

MBS event builder

GSI IT “green cube”




```
MBS> connect fsq -config myconfig.conf
MBS> open file -fsq -auto /lustre/exp/mar24/run3.lmd
```

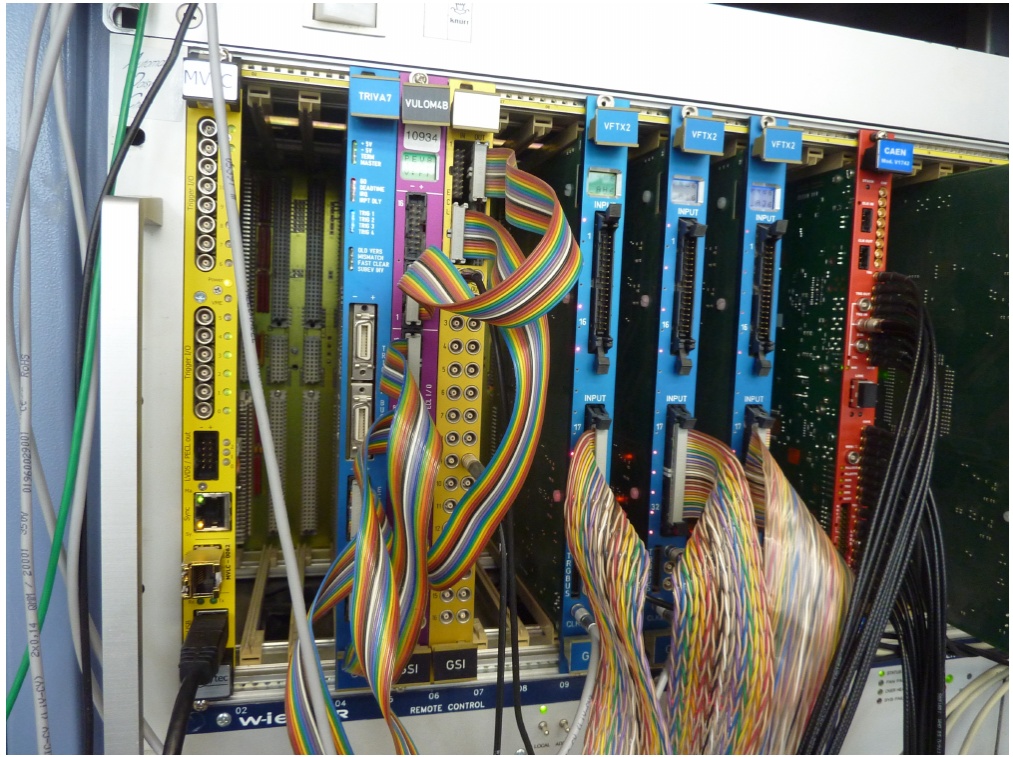
Miscalleneous

- Support of file sizes up to 128GiB
- Improved exit handlers, *f_user_exit()* API
- Fix of *resl* command (failed for long pids on Debian11)
- Several bugfixes (?)

Summary

- MBS release v7.0 provides many developments since 2016
- PCIe X86L- hosts ready for upgrade to Debian 11
- New VME platforms: IFC, MVLC
- White Rabbit trigger time latch unit supported
  (VETAR, PEXARIA)
- New GSI storage protocol FSQ integrated to MBS transport
  (for all MBS Linux platforms, even RIO4)

Thank you!


TRIVA7
VULOM4B
VFTX2
VFTX2
VFTX2
CAEN
INPUT
INPUT
INPUT
GSI
REMOTE CONTROL

References

[1] IOxOS Technologies, mvme - IFC1211 documentation,
Available at https://www.ioxos.ch/produit/ifc_1211

[2] mesytec GmbH Co. KG , mvme - mesytec VME Data Acquisition documentation,
Available at https://www.mesytec.com/products/nuclear-physics/MVLC.html

[3] J.Adamczewski-Musch, N.Kurz, S.Linev, Status and developments for DAQ system MBS v6.3,
in: GSI Scientific Report 2014, Available at http://dx.doi.org//10.15120/GR-2015-1-FG-CS-05

[4] D.Beck, A.Hahn, et.al Timing Release – Fallout-v6.2.1,
Available at https://github.com/GSI-CS-CO/bel_projects/releases

[5] J.Adamczewski-Musch, N.Kurz, S.Linev, GSI Data Acquisition System MBS Release Notes V6.3,
Available at https://www.gsi.de/fileadmin/EE/MBS/gm_mbs_rel_63.pdf

[6] T.Stibor, Lustre TSM File Storage Queue,
Available at https://github.com/tstibor/ltsm

Bonus slides

Command line tool mec

```
X86L-158 adamczew > mec -h
*************************************************************************
m_test_mvme for mesytec mvlc
v0.16 23-July-2021 by JAM (j.adamczewski@gsi.de)
*************************************************************************
usage: m_test_mvme [-h][-x][-m][-d][-r[-t][-n]|-w[-t][-n]|-b|-f]
Options:
-h                          : display this help
-t                          : TRIVA mode (address is offset to TRIVA base) (0)
-x                          : hex mode (read and display as hex numbers) (0)
-m am                       : specify address modifier (0x9)
-d dw                       : specify VME datawidth (D32)
-a address                  : specify VME address (TRIVA offset if option -t)
-r                          : read VME register at address (TRIVA offset if option -t).
-b words                    : VME block read of specified words from given address.
                              Default am(0xb) may be changed with -m
-w value                    : write value to VME address
-n repeats                  : Read or Write repeats (1) number of words consecutively,
                              starting from given address
-f filename.mec             : write values to VME addresses as specified in ASCII
                              configuration file (line format: address value am dw)
```

New MBS commands for MVLC

- START MVLC (replaces START ACQUISITION)
- STOP MVLC (replaces STOP ACQUISITION)
- SHOW MVLC (replaces SHOW TRIG MOD)
- SET TRIG MOD (reimplemented)
- ENABLE/DISABLE TRIG MOD (reimplemented)
- ENABLE/DISABLE IRQ (reimplemented)
- READ MVLC (read from any VME bus address via MVLC controller)
- WRITE MVLC (write to any VME bus address via MVLC controller)

Mass storage file interface

- Old protocol RFIO(gstore) deprecated at GSI
  => **new system LTSM established in 2019** (Thomas Stibor, GSI IT [6])
- LTSM client requires IBM TSM libraries! Not available for all MBS platforms.
- **Gateway application rawServLTSM** was developed **for MBS v6.3**
  - runs on X86_64 PC with TSM libs only; RFIO as interface from MBS
- Direct connection between MBS and storage system?
- Faster availability of files on /lustre for near online analysis?