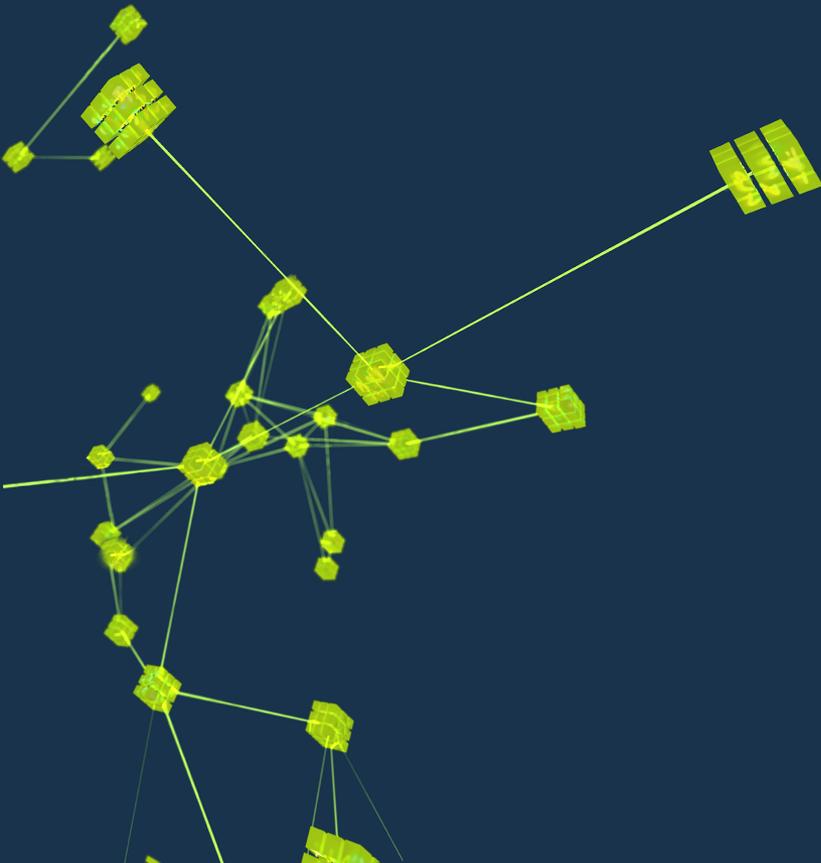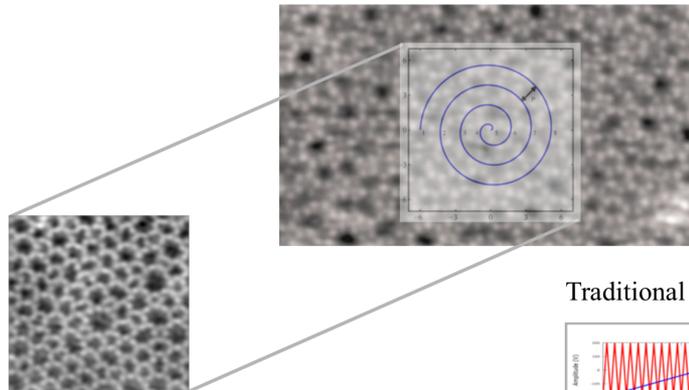# EPICS

Rtems 5
+ EPICS 7 + VMEbus

RTEMS 5

- We run RTEMS 5 (formerly known as RTEMS 4.12 aka RTEMS 4.11.99)

- On VMEbus PPC - CPUs mvme6100 (in produktion) and mvme2500 (not well tested)

- On raspberryPi's

- RTEMS5 Build System uses GCC 7.3.0

- C++11 -> smart pointers are supported

- RTEMS 5 support included into EPICS 7 (merge request)

- Uses Posix Api

- Most of the existing libcom (posix) is used for RTEMS 5 with only some minor changes, does not interfere with existing RTEMS 4.x

- EPICS 7 on IOC's with QSrv is in production on VMEbus with mvme6100 with multiple ethernet interfaces
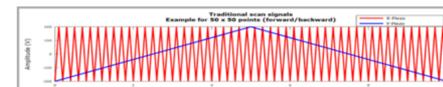
- Heavily use of NTNDArray data type

Large atomically resolved STM image of the vitreous silica film revealing the Si positions

V_S = 2 V, I_T = 50 pA,

Scan range = 12.2 nm
       x 6.6 nm

Recording Time > 10 min

Scan range:
5 nm x 5 nm
Recording Time:
up to 3000 f/s

Traditional Scanning (triangular for x/y)

- EPICS 7 at all works well

- Problems with mixed up output on console (RS232)
  overlapping output is happening on stdout and stderr has been
  redirected (libmisc/redirector)

```
gonzo> epicsSocketDestroy: failed to close a socket because "Device or 2r0e1s8o/u0r6c/e1 2b u1s1y:"1
1:18.994 [sis3316Ops.cpp:624] sis0: updaftea tthraeadl  stsopopiungr
c20e18:/0 6R/T12E M11S:1_1F:A19T.0A89L _[SsiOs33U1R6COEp_s.EcpXp:I5T
bsp_fatal_extension(): RTEMS terminated –- no way back to MotLoad so I reset the card
Printing a stack trace for your convenience :-)
```

- Lack of a dynamic link loader, no cexp, no gdb

- Hardware/Configuration issues on VMEbus (e.g. with SIS3316, data
  transfer/DMA)

EPICS

- Tests with mvme2500

- Fix hardware issues on VMEbus (bus controller), DMA, …

- Development of a dynamic link loader for RTEMS -> cexp
  "The sources of the dynamic linker, a feature that RTEMS is currently missing, shall be integrated with the RTEMS source code and published with the same license as RTEMS itself. The feature shall become an integral part of the main development branch of RTEMS"

- RTEMS libdebugger for PPC
  https://git.rtems.org/rtems/tree/cpukit/libdebugger
  hard to implement due to the many PPC register models