Atlas RCE Infrastructure Support and Experiences

M. Wittgen

RCE Workshop

07/27/16

Outline

- Atlas specific software release distribution and resources
- CSC
  - Shelf Manager Tools
- AFP
  - HSIO2 Deployment in P1
- Application Middleware
  - Trigger DAQ
  - Other Lightweight Middleware
- Outlook
  - Shelf Manager Tools
  - Test Stand Software Distribution
- Conclusions

Atlas specific (SDK) distribution

- Snapshot of SLAC SDK version V0.11.1 with a few enhancements and modification
- Provide a slower SDK release cycle for Atlas
  - Upgrade to newer versions as interesting new features become available
- One installation tarball
- Additional features/packages (might be moved into mainline release)
  - Snapshot of ArchLinux ROOTFS
  - Python based tool to partition/create SD cards for DTM/DPM/HSIO2/Zedboard
  - Preconfigured QEMU
    - Software emulation of ZYNQ CPU
    - Can be used to update ArchLinux ROOTFS image on a server disk
      - Newer SDK version support NFS ROOTFS
  - ARM Cross Compilers included
  - Enhanced Busybox (RAMDISK Linux)
    - Updating SD cards through RSYNC
    - Includes tools to change UBOOT environment settings
- DTM/DPM IPs are assigned by external DHCP server
 - CERN networking requirement

Atlas RCE Resources

- Hosted at CERN
  - Access to SW/FW does not require a CERN login
- RCE portal (Drupal website)
  - Hub to quickly navigate to resources
  - http://cern.ch/RceProject
- SDK, prebuild FW and ArchLinux snapshot in web accessible filesystem
- Application software, FW sources hosted in GITLAB
  - https://gitlab.cern.ch/groups/rce
- General documentation hosted in Atlas TWIKI
- phpBB based discussion forum
  - http://rceprojectportal.web.cern.ch/RceProjectPortal/forum/

CSC (Shelf Management Tools)

- 6 COBs in one ATCA crate
  - 32 chambers read out by DPMs (RTEMS)
  - 6 DTMs (ArchLinux)
- Software install cycle
  - Boot DPMs to ArchLinux, SCP, boot back to RTEMS
  - Wrapper for SM tools to provide more high level functionality
    - Identify resources by detector label not by shelf
      - For example, use of chamber name ECA01 instead of cscnrc/0/0/0
    - IPs matter for Atlas (fixed)
    - Hiding shelf structure in tools: atca_ip, etc.

rce_talk Tool

- Written in Python
  - Bulk (parallel communication) to N RCEs through SSH (ArchLinux), telnet (RTEMS), TELNET (DPM serial)
- Multiple terminals with keyboard input broadcast to all
- CSC running ser2net daemon on DTMs to provide access to DPM serial consoles through telnet
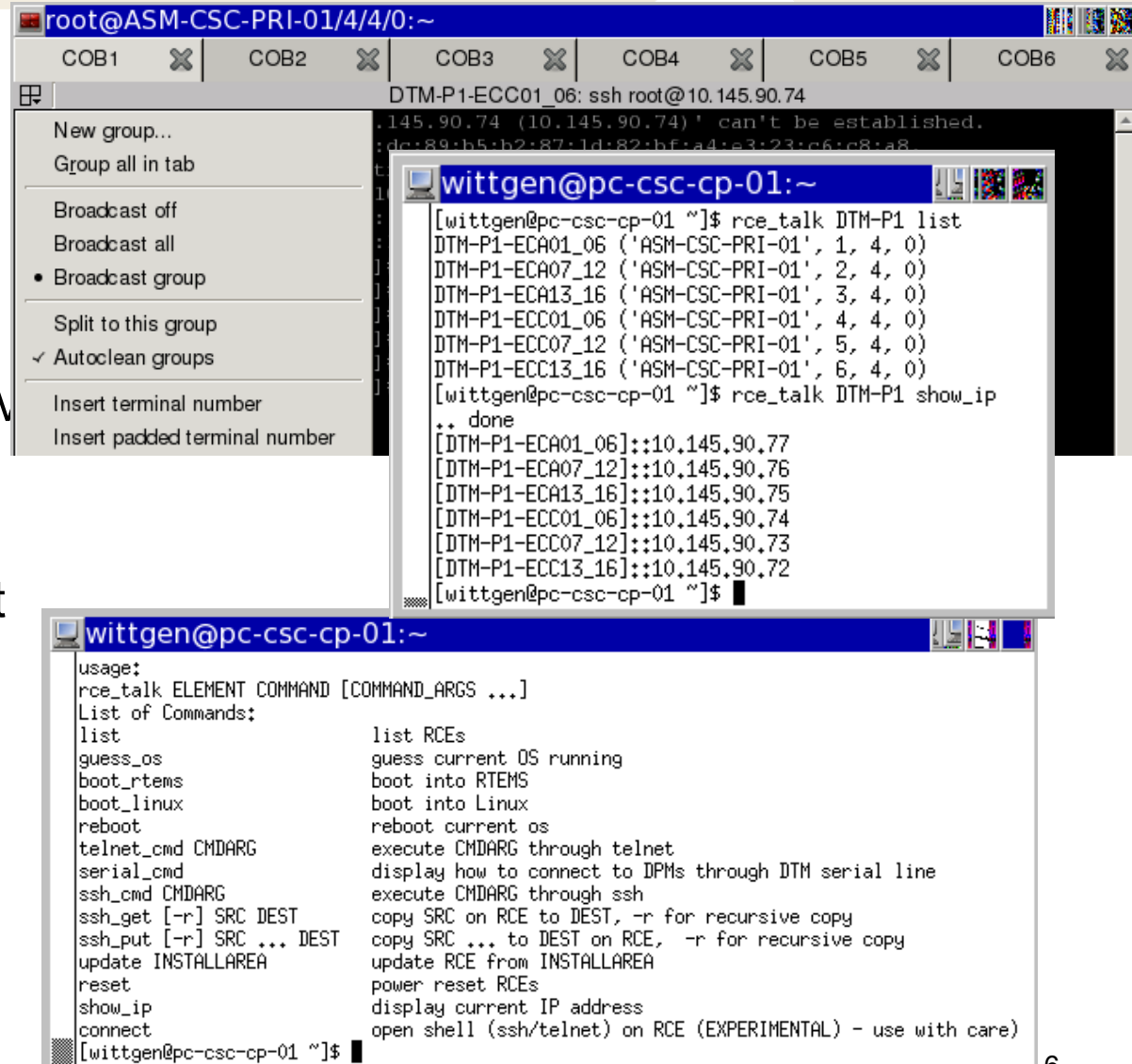
AFP (ArchLinux in Atlas P1)

- Currently operates one HSIO2 based DAQ system in Atlas P1
  - CERN IT requires regular security patch installation of networked devices in P1
- ArchLinux provides rolling update mechanism
  - Could break application software
    - Not ideal for large production system
  - Release based Linux distributions preferred
- Extra work to accommodate stable package versioning and security updates
- CentOS 7 ARM might be better suited in such an environment
  - Production readiness not clear
  - ARM systems might reach critical mass some day at CERN to support such a project

Atlas TriggerDAQ

- Atlas provides a powerful DAQ software framework (TDAQ)
  - Not currently used directly on embedded system other than Intel based SBCs
- Currently using ScientificLinux 6
  - But compiles and runs on CentOS 7
- Precompiled packages used for the host server side in RCE applications
  - gcc49, ROOT, BOOST
  - Stay in synch with TDAQ releases
    - Facilitates future integration into Atlas
  - Distribution through RPMs and a modified YUM (ayum) that does not require ROOT privileges
  - Considering to move SDK/ArchLinux distribution to ayum or similar
    - Better modularity, update and dependency management

TDAQ Middleware

- TDAQ provided middleware
  - Inter process communication (IPC): CORBA based
  - Histogramming, information server, distributed process management
- Packaged have been compiled for ArchLinux/ARMV7L and released as RPMs through ayum
- Could provide direct integration of RCE into Atlas partition infrastructure without need of additional middleware
- Lacking official support of ARM platform

TDAQ Continued

- TDAQ tends to be heavyweight
  - Lacks package modularity
    - Python, JAVA dependencies hard to eliminate in package build process
  - Outdated build tool (CMT), no concept of cross compilation
    - Added for unofficial ARM TDAQ release
  - Switch to CMake in progress
- No concept of remote installation
  - Full of burnt-in CERN AFS paths
    - Relocation might fail
  - RPM distribution mechanism tends to break
    - With more remote installations might become less CERN centric
- Pixel RCE application switched to CMake as well
  - Defines cross compilation target for ArchLinux/ARMV7L

Lightweight Middleware

- CSC (RTEMS)
  - Sun/RPC with JSON objects as payload for DAQ run control integration
  - UDP packets with JSON payload to publish FW counters to information service
  - Outdated RTEMS network stack prevents adaption of more modern SW frameworks
- Pixel
  - Remote Call Framework (RCF) as IPC replacement
    - http://www.deltavsoft.com/
  - No IDL compiler required (de/serialization of C++ objects done programmatically)
  - Alternatively TDAQ can be used
- TDAQ is moving to ZEROMQ as transport layer
  - JSON/ZEROMQ seems to be a good alternate over binary objects
    - http://zeromq.org/

Lightweight Middleware

- CSC TTCRx Firmware registers displayed in GRAFANA dashboard
  - https://atlasop.cern.ch/tdaq/pbeast_dashboard/#/dashboard/file/
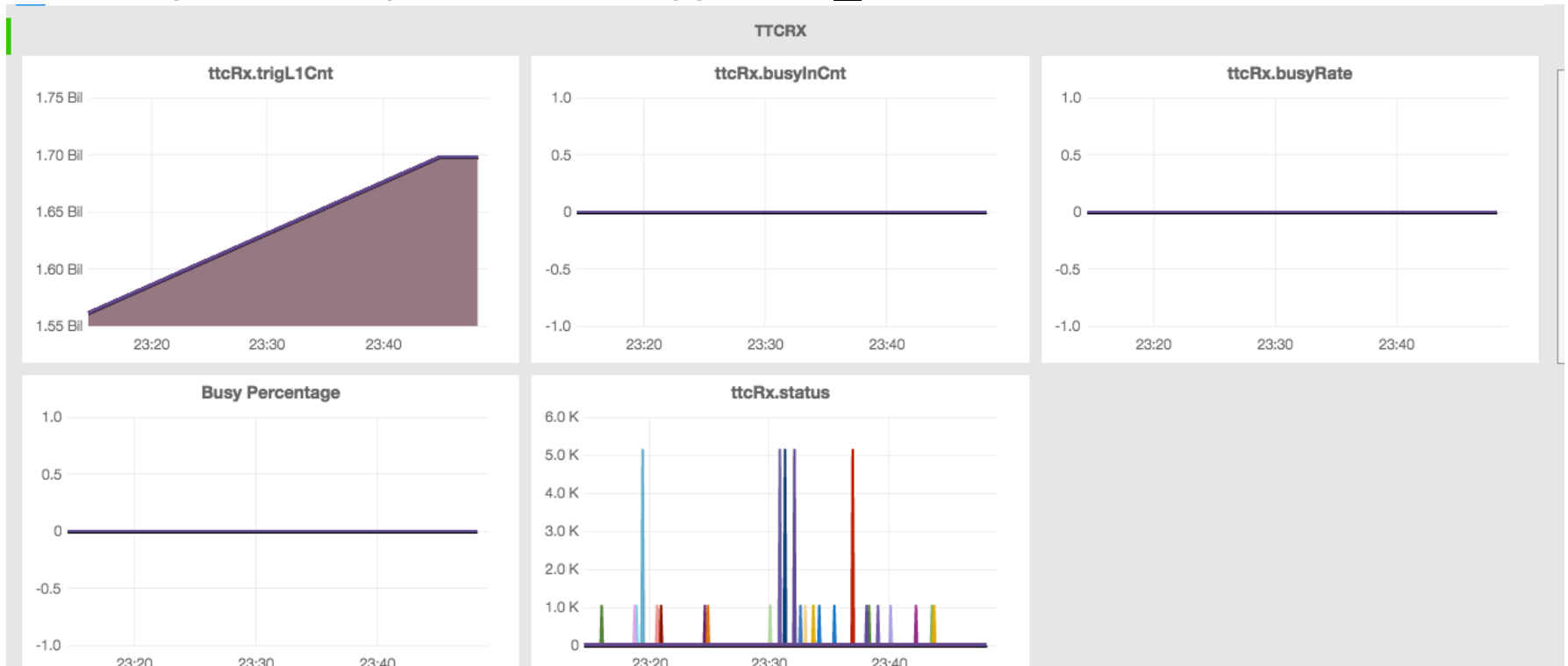
Outlook (Shelf Manager Tools)

- Various Tools developed by different SDK users with similar scope
- Current shelf tools mostly command line based
  - No API provided
  - No strict separation between tool and library functionality
  - Duplication of code needed for integration into existing projects
- Identify common software core and add to SDK
- Wish list
  - Provide shelf manager API
  - Python and C++ bindings

13

Outlook (Software distribution)

- Simplify installation and deployment of test stands
- One example: pixel module testing
  - Currently requires many steps by user
    - Specific OS installation and configuration
    - Base software installation
    - RCE application software compilation
    - Run-time setup
- Move towards an appliance based model
  - Appliance containers with docker
  - www.docker.com

Conclusion

- At the moment Atlas RCE installations at CERN have different requirements
  - Networking more restricted
  - Embedded devices needs to be security hardened
- Might change in the future with a wider spread of OS and DAQ system flavors
- Work in progress
  - Will go through several Atlas distribution cycles as core the SDK and the RCE infrastructure evolve
- Always keep TDAQ integration in mind