High Performance Simultaneous Data Acquisition Solutions for Fusion

Peter Milne

D-TACQ Solutions Ltd, East Kilbride G75 0QD UK

Intelligent Digitizer:ACQ196CPCI

96 simultaneous analog inputs in one slot.

- 96 channels x 500kSPS 16 bit differential, protected inputs.
- Optional Analog Outputs, Digital IO on rear transition module.
- Low latency and streaming firmware available.

Intelligent Digitizer: ACQ216CPCI

16 simultaneous analog inputs at 12MSPS.

Captures a gigabyte to onboard memory in 3s

- 16 Channels, 14bit, 12MSPS/channel.
- 8 Channels x 25 MSPS/channel, 4 x 50MSPS software selectable.
- Front panel 16 dual-pin LEMO, or isolated SMA.
- Soft switchable input ranges +/-2.5V, +/-4V, +/-10V
- Analog input mezzanine – choice of 3.
- Superior Connectivity: 64bit/66MHz PCI, dual Gig-Ethernet, CSMB
- Application Specific Compute Power: Xilinx VirtexII FPGA

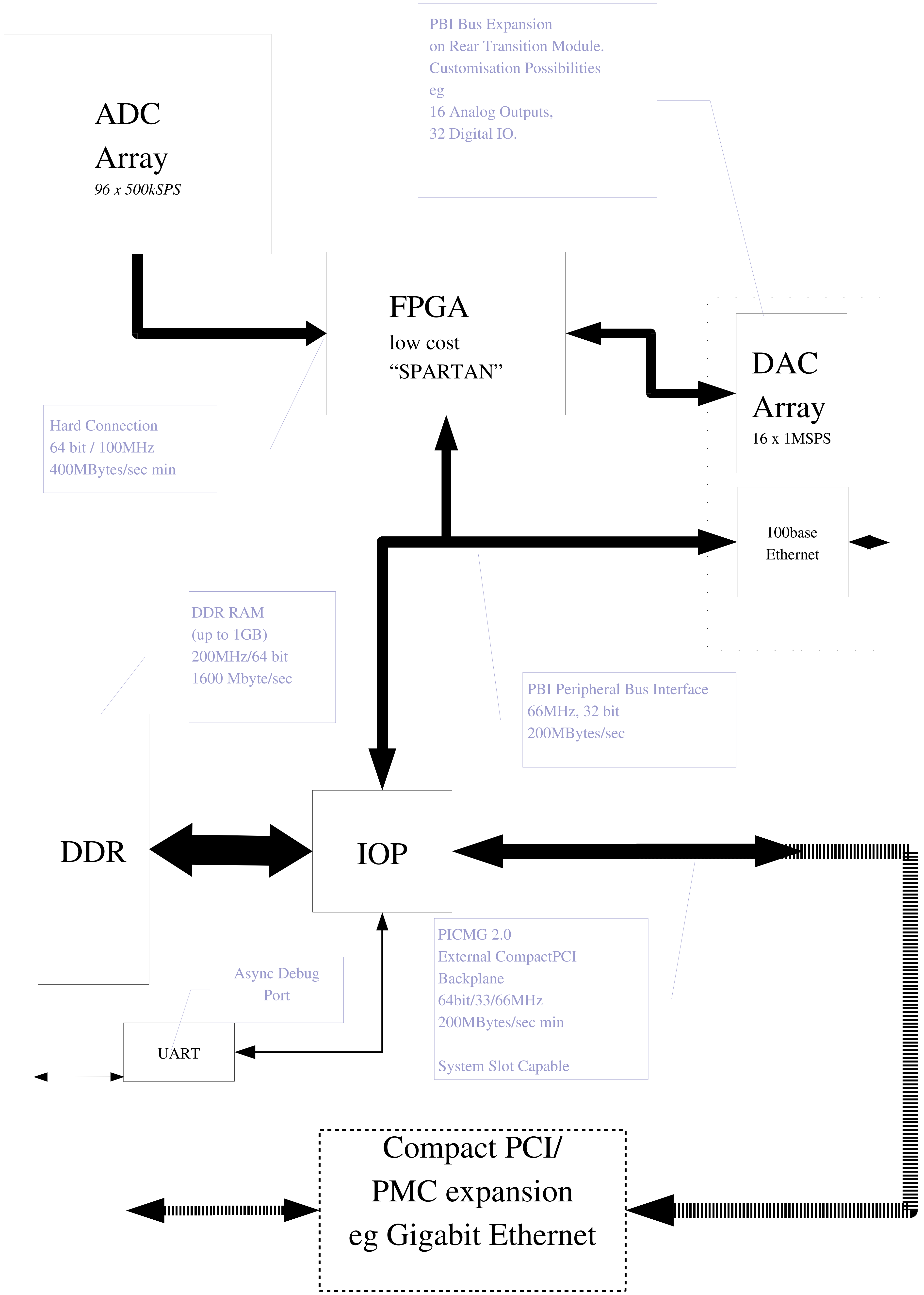
Shared Platform Features:

Open Standard Hardware, Open Source Software

- Standard CPCI 6U card, Compact PCI Peripheral or system slot capable
- Cards can run standalone, in synchronized groups or under host control.
- Fully Differential inputs, transient overvoltage protected
- Up to 1Giga Byte acquisition memory
- Embedded microprocessor running Linux – data management and networking

ACQ196CPCI Data Flow

High Bandwidth and standard bus expansion capability



Low Latency Control Solutions

Plasma Control

64 bit, 66MHz PCI bus interface.
Compact PCI Peripheral slot device.

Capable of efficient low latency transfers, using bus master DMA,

Incoming:: sub 10uSec from convert clock to 96 channels data transferred to host memory..
Outgoing: sub 5 uSec from host memory to convert complete.

AI – analog input
AO – analog output
DIO – digital IO

ACQ196CPCI,
96 AI Channels, 2usec conversion, 16bit
16 AO Channels, 1usec conversion, 16bit.
128MB memory.

Input range: +/-10V, differential, protected to 100V

Can be configured as standalone networked Linux system.
Optional large size FPGA for real time signal processing.

Rear Transition Module
RTM-AO16

16 AO/
68W SCSI2

Control Algorithm can be implemented in either on board FPGA
(sub 10uSec end-to end latency)
or on
external Host Computer.
(sub 100uSec end-to-end latency)

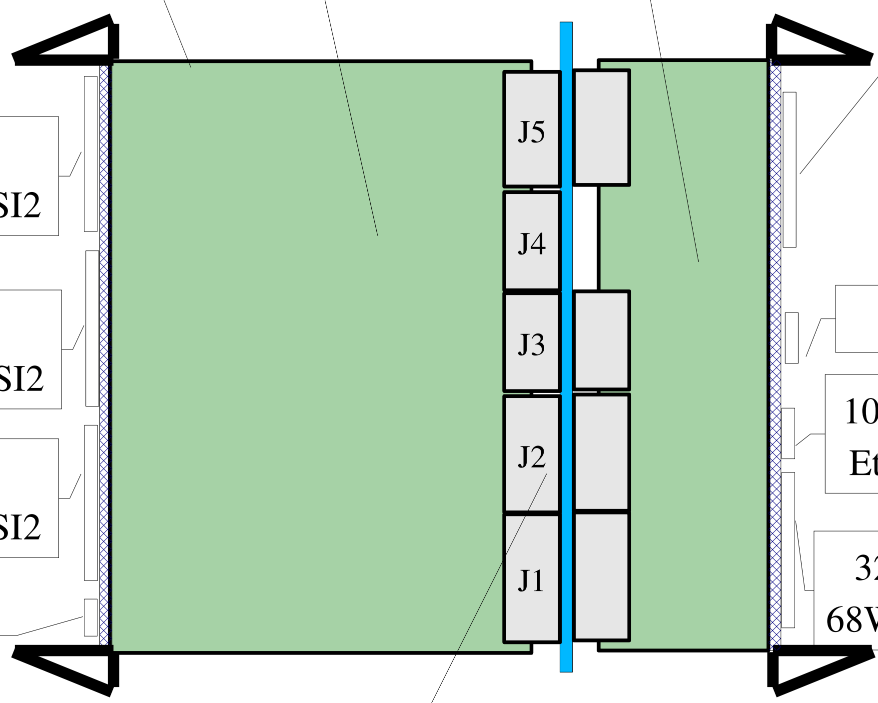
Possibility for mixed control - handle the highest priority data on board, slower, more complex data off board

32 AI
on 68W SCSI2

32 AI
on 68W SCSI2

32 AI
on 68W SCSI2

EXT CLK,
TRIG on LEMO



RS232

100BaseT
Eth RJ45

32 DIO/
68W SCSI2

Multiple Cards Share Clock and
Trigger via PXI compatible signaling lines in backplane

Typical Deployment for Plasma Control:

- 2 Crates, one Compact PCI, one industrial PC host computer using a high end server motherboard and processor.
- Link to Compact PCI crate via a PCI bridge extender.
- 2 ACQ196CPCI cards with 192 AI, 32 AO.
- Control algorithm runs on host computer, the ACQ196CPCI cards transfer data to a shared buffer in host computer memory by DMA.
- ACQ196CPCI firmware driven control loop provides valuable timing information.

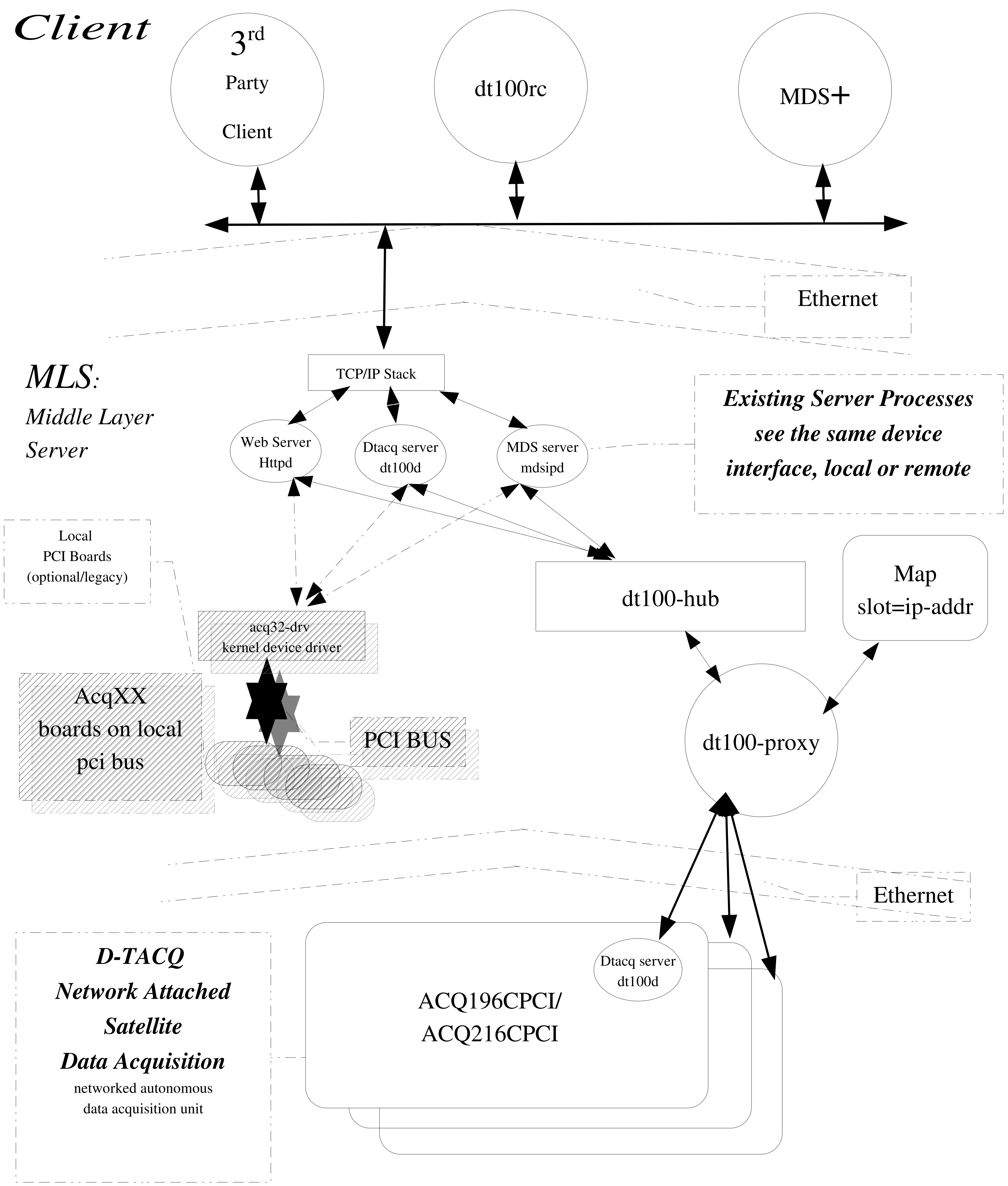
An alternate configuration has been prototyped:

- In this case, ACQ196CPCI is in slot 1 of the crate, and distributes data to a DSP processor farm and reflective memory mounted in the crate.

Transient Capture

Shot Diagnostics: Langmuir Probes, Detector Arrays

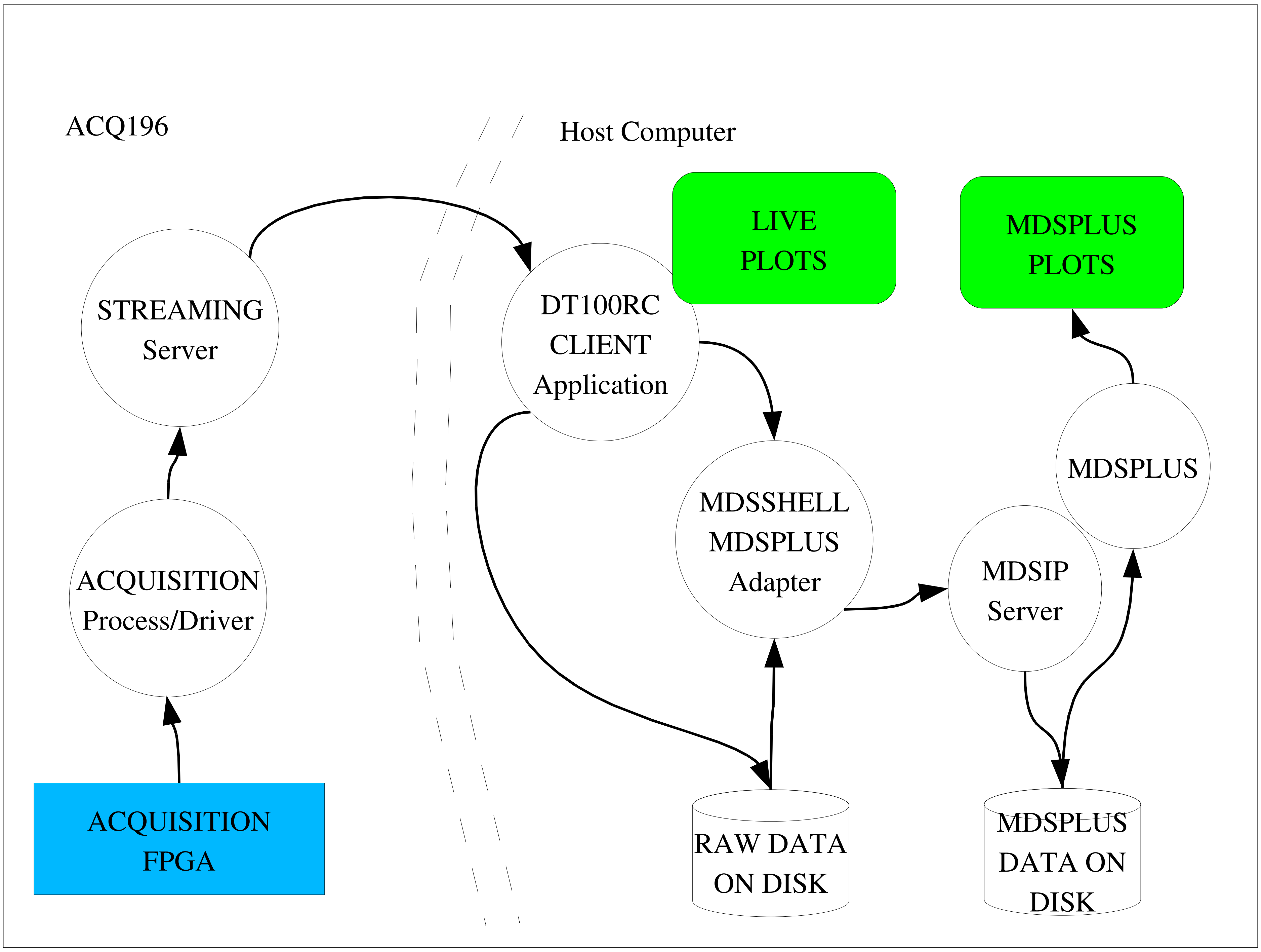
3 Layer Architecture – using the HUB proxy device



Continuous Operation Diagnostics

Long Pulse example: Gyrotron Test System

- 288 Channels 1kHz streaming data to live display, with continuous put to MDSplus
- 96 Channels 250kHz, "Medium" series of Pre Programmed Triggers (PREPs) – Target Push submit to MDSplus
- 16 Channels 12MHz, series of Pre Programmed Triggers (PREPs) - Target Push submit to MDSplus
- All channels must stop on Fault Trigger – data is then available via normal Transient upload. Implementation uses HUB to perform Host Pull to MDSplus
- Implemented in 5 slots in Compact PCI chassis, with 5 ethernet uplinks aggregated by a gigabit switch.
- Host computer is a regular off the shelf server running RHEL. Hot Swap disks for data backup.

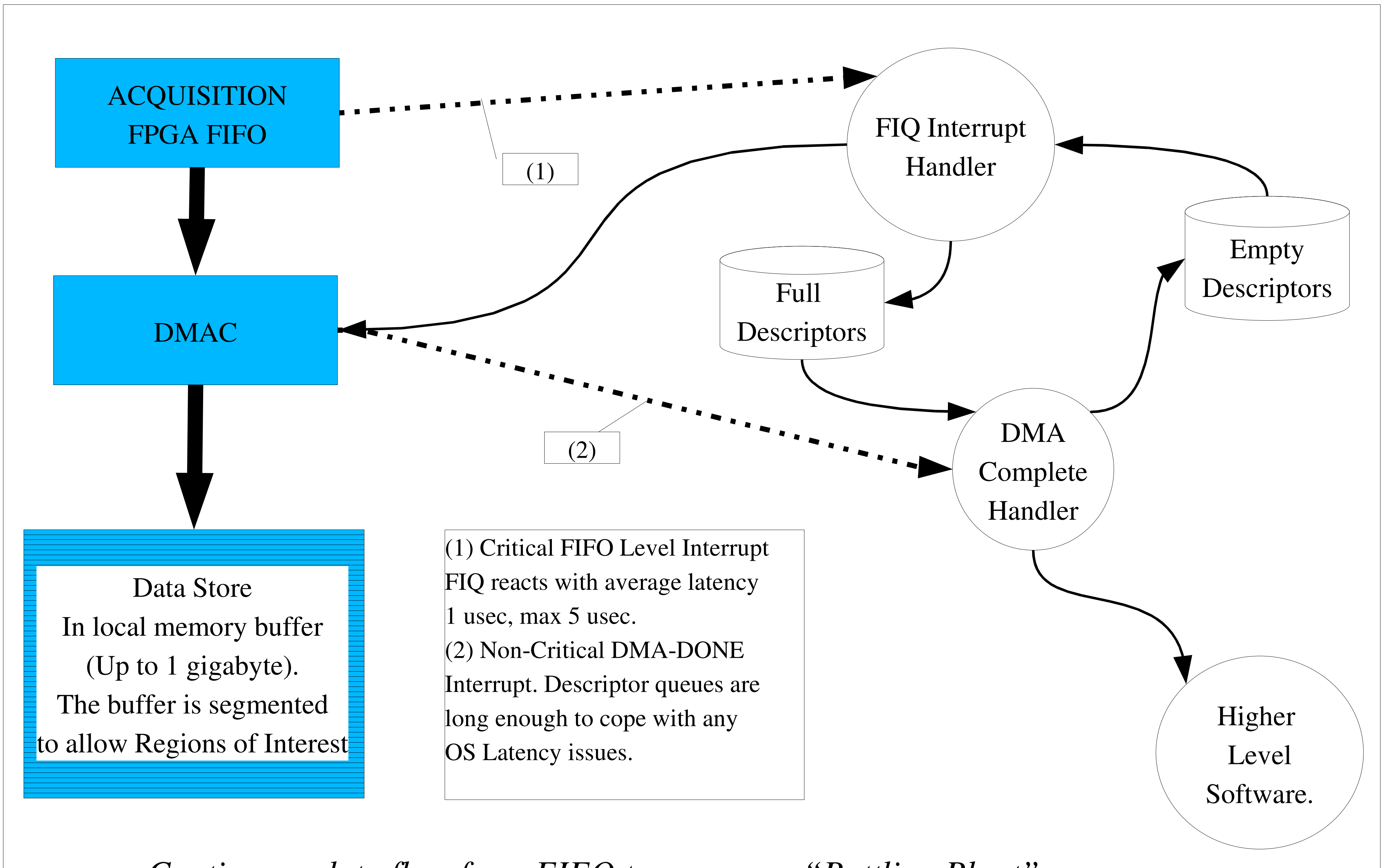


Continuous Streaming Display and Store to MDSplus

Real Time with Linux?

Techniques to achieve Real Time Performance

- Plasma Control firmware is implemented as a custom Kernel Loadable Module. During the shot, interrupts on both Host and Target machines are disabled, and the software executes a single thread, tight, predictable loop
- For diagnostics operations, we have to preserve normal multi tasking operation – for example, data, control and diagnostic information flows via backplane or network interface should continue as normal.
- How to achieve a 400MByte/Sec data flow to memory and have spare CPU cycles for housekeeping?. The embedded processor, designed for RAID controller use, has excellent DMA performance, and coupled with fast and wide local bus and DDR memory, transfer speed is good.
- FIFO size and interrupt latency – the single critical interrupt in the system is triggered by FIFO level – this interrupt is handled by the ARM FIQ interrupt, this is a high priority interrupt invisible to Linux, and never masked, so is fast and highly predictable.
- We ensure that the timing of all other data flows is non-critical by providing large ring buffers, and downstream handling is easily handled by normal Linux managed interrupts and tasklets.

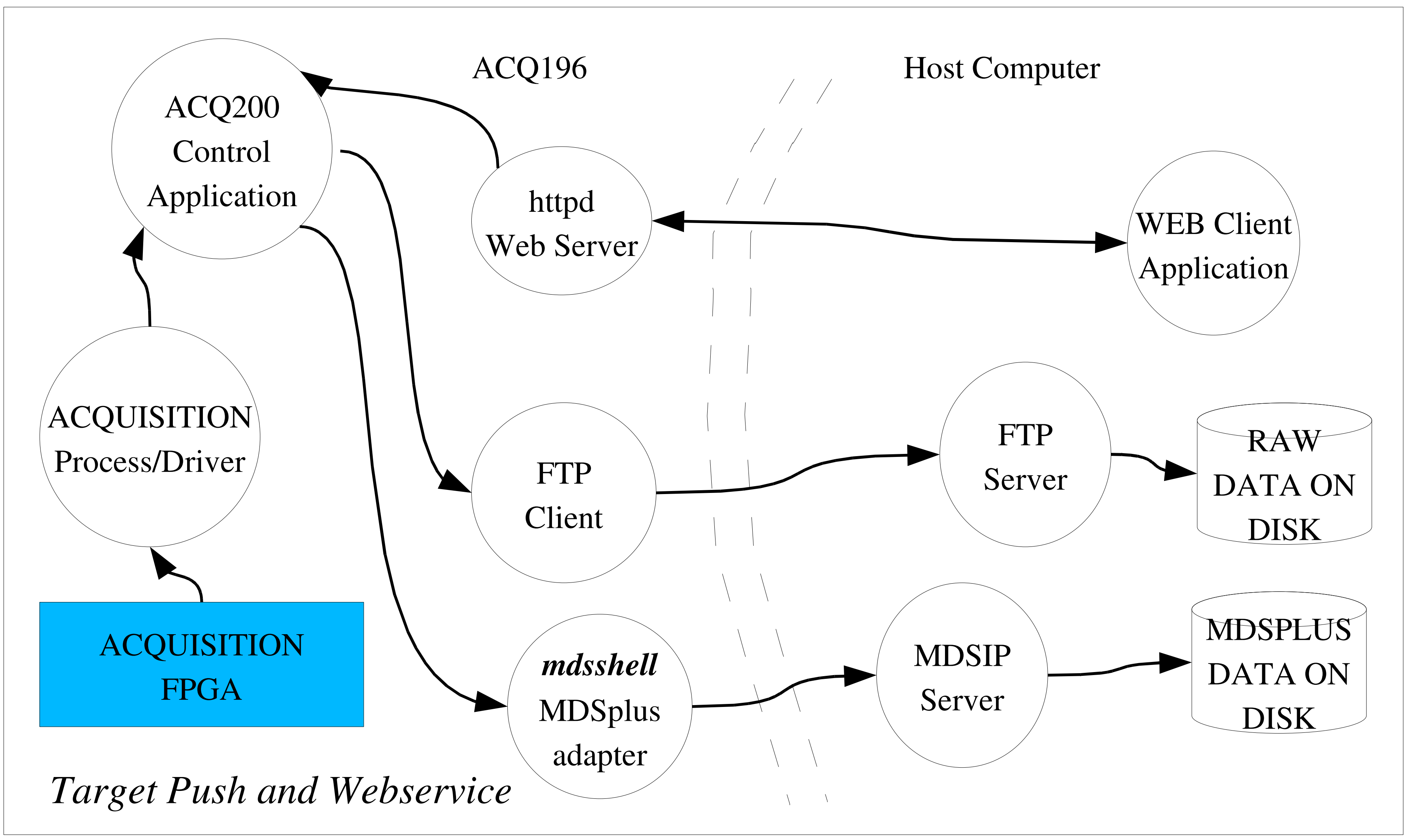


Continuous data flow from FIFO to memory - "Bottling Plant".

Transient Capture 2

Multiple ways to communicate

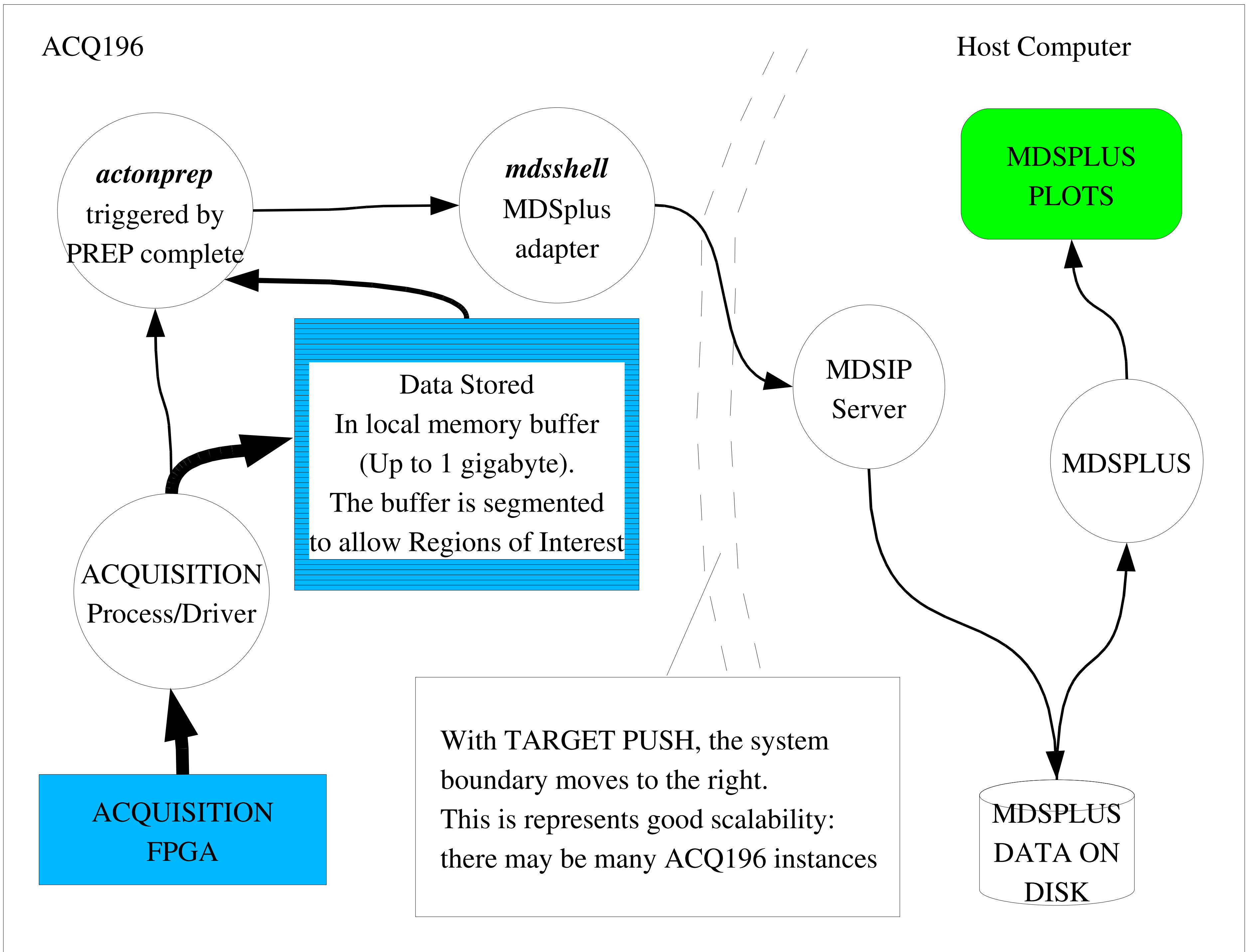
- Hub operation above – Host Pulls data off the Target. The Hub device acts as a proxy for the networked appliance, designed for back compatibility with existing PCI interfaces.
- A solution with great scalability is to have the Target Push data back to the host when it is available. Two techniques that have been demonstrated include MDSplus Thin Client - "mdshell" process runs on ARM embedded target and communicates with a remote MDSIP server on the network. Second, upload has been implemented by automated ftp script. In principle, any other published, preferably Open Source protocol may be implemented – the embedded system has full glibc and C++ support.
- For control, we may have a stored procedure on the target i.e. a "boot script", in a depply embedded system this can be stored in non-volatile Flash File system. Or, a host controller may deliver a configuration before the start of shot, and there are many ways to do this – using the HUB connection, via ssh, or via the embedded webserver as a webservice. A webservice implementation has been demonstrated that will upload a comb of the sampled data (data set reduced in time by decimation), and then upload regions of interest at the full sample rate. MDS thick client software has been ported to the ARM Linux processor opening possibility for running TDI scripts local to the acquisition unit.
- Modern data acquisition hardware is capable of generating huge amounts of data. By incorporating intelligence close to the point of capture, we aim to provide the subset of "interesting data" to higher level software, reducing required network bandwidth, time to upload, storage space, but perhaps most importantly, reducing the need to archive and search redundant data.



Continuous Operation Diagnostics 2

Long Pulse example: Gyrotron Test System

- High speed data – 96x250kSPS and 16x12MSPS – too fast to stream on Ethernet, and it makes no sense to store 400MB/sec for 3600 seconds anyway
- Setup "Regions of Interest" before the shot as a list of Pre Programmed Triggers (PREPs)
- Capture proceeds a full speed to onboard buffer.
- When a trigger condition is reached, the corresponding data set is reserved. Capture continues, but the reserved areas will not be overwritten.
- PREPs appear to application software as a tree of device files. Application software is triggered at the end of the shot, and data is posted to MDSplus using "Thin Client" software ported to the embedded ARM Processor.
- At all times, the FPGA hardware is ready to detect and act on an external hardware Fault Trigger.
- At all times, capture proceeds at full speed, and trigger detection is continuously available, data sets are uploaded, all with ***Zero Dead Time***.



Storing PRE Programmed Triggers to MDSplus via Target Push: