A GPU based solution for distributed FX correlation ?

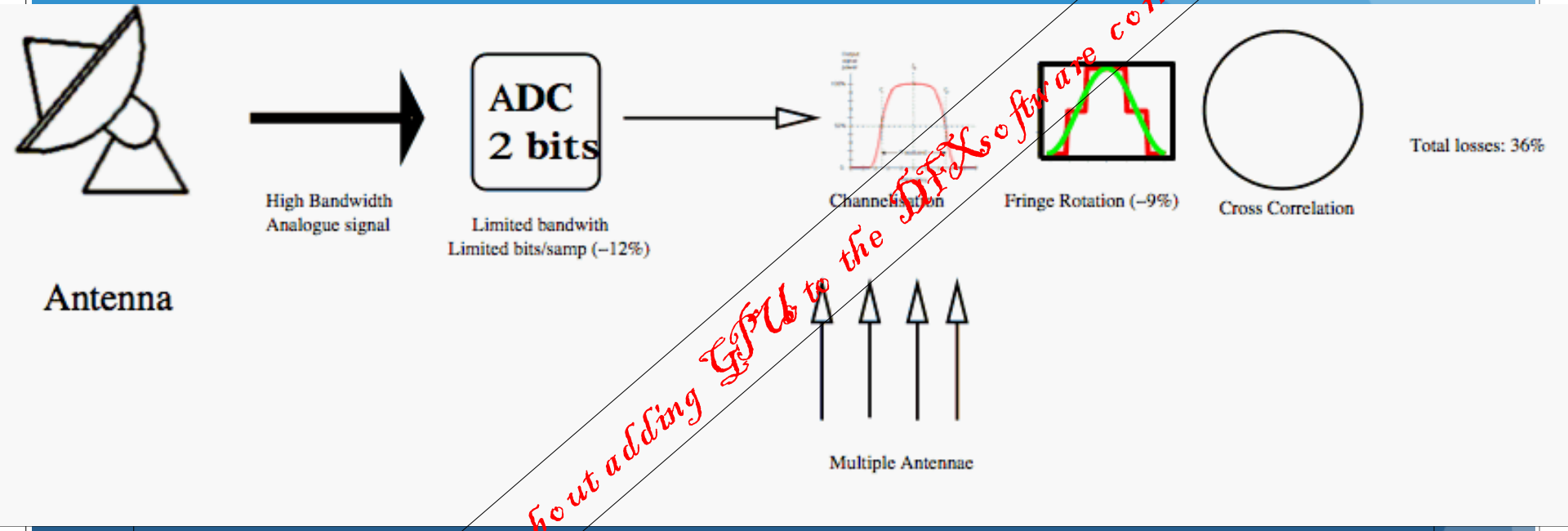
Click to edit Master subtitle style

Richard Dodson, ICRAR
Dick Ferris, CSIRO
Chris Phillips, CSIRO

ICRAR International Centre for Radio Astronomy Research

13/08/09

VLBI Correlation (traditional)



- Fringe rotate (and delay)
- Correlate

This talk is not about adding GPU to the DiFX software correlator!!

Correlator Losses

From Thompson, Moran, Swenson

- Conversion of analog to digital

  2 bit conversion (4 level) -12%

- Delay and rate correction

  Fringe rotation (3 bit 2 level) -6%

- Discrete Delay Step

  Delay for Centre of band Δv/2 -3%

TOTA -22%

Correlator Losses

- Conversion of analog to digital

  2 bit conversion (4 level) Worse than -12%

- Delay and rate correction

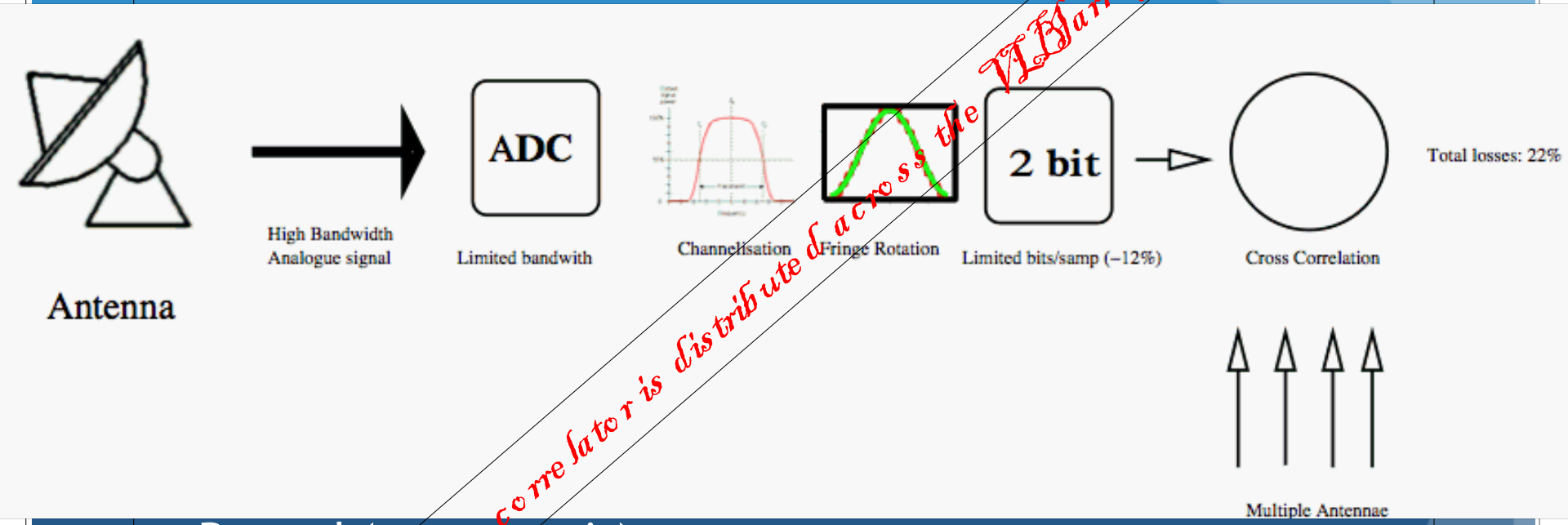
  Fringe rotation (3 bit 2 level) -6%

- Discrete Delay Step

  Delay for Centre of band Δv/2 -3%

TOTA Much Worse than -22%

VLBI Correlation (alternative)



The correlator is distributed across the VLBI array

- Record (or transmit)
- Load into correlator

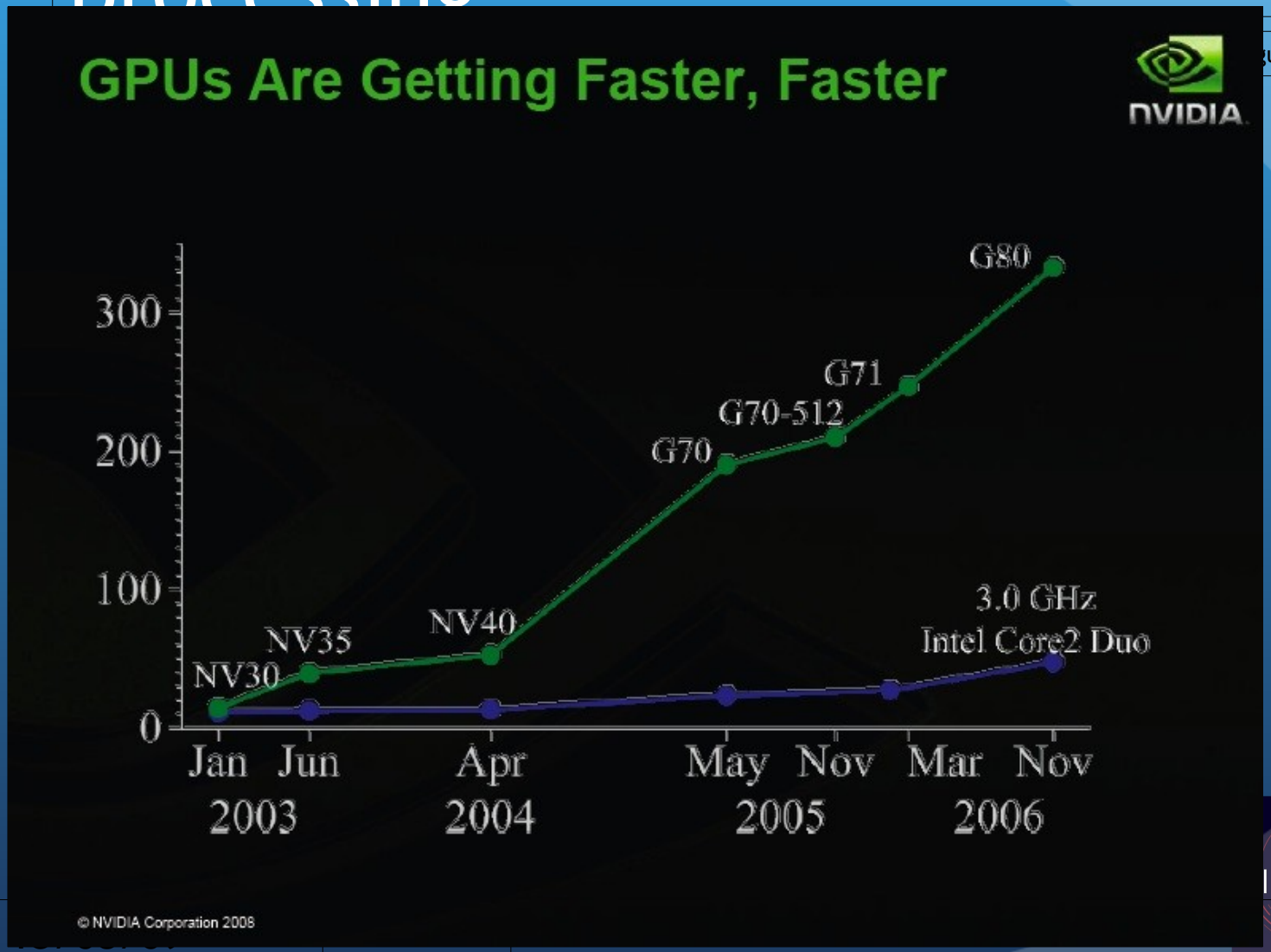
Gains from pre-encoding

- Fringe rotation and channelisation at high bits per sample: Near zero losses.
- RFI Handling: Channel excision at high bit levels
- Optimum 2-bit encode: Better compression.
- Transfer of processing power to antenna hardware: Better use of processing power.
- Reduction in correlator hardware requirements: Support >GBps without upgrading correlator hardware.

GPUs – massively parallel processing

- Graphical Processing Units are designed to process the screen pixels, in parallel
- The same processing power is now being turned to `similar' tasks
- New languages (Cuda & OpenCL) are hiding the Graphics Primitives
- Works well for suitable tasks

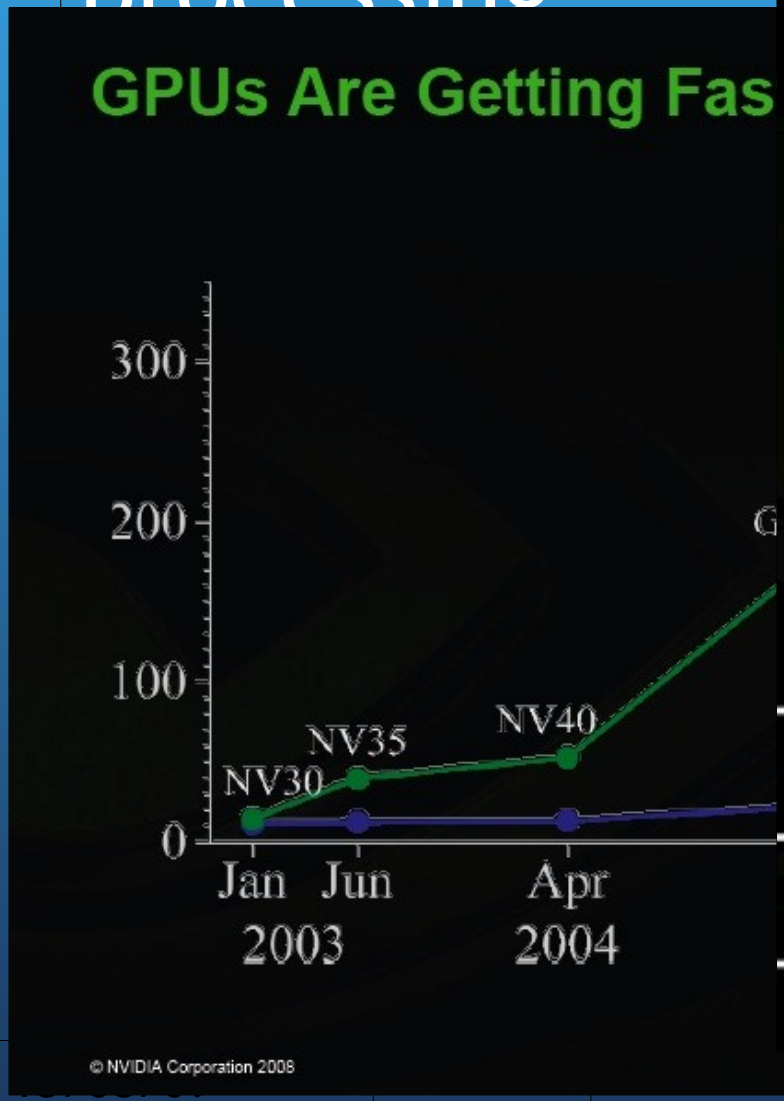
GPUs – massively parallel processing

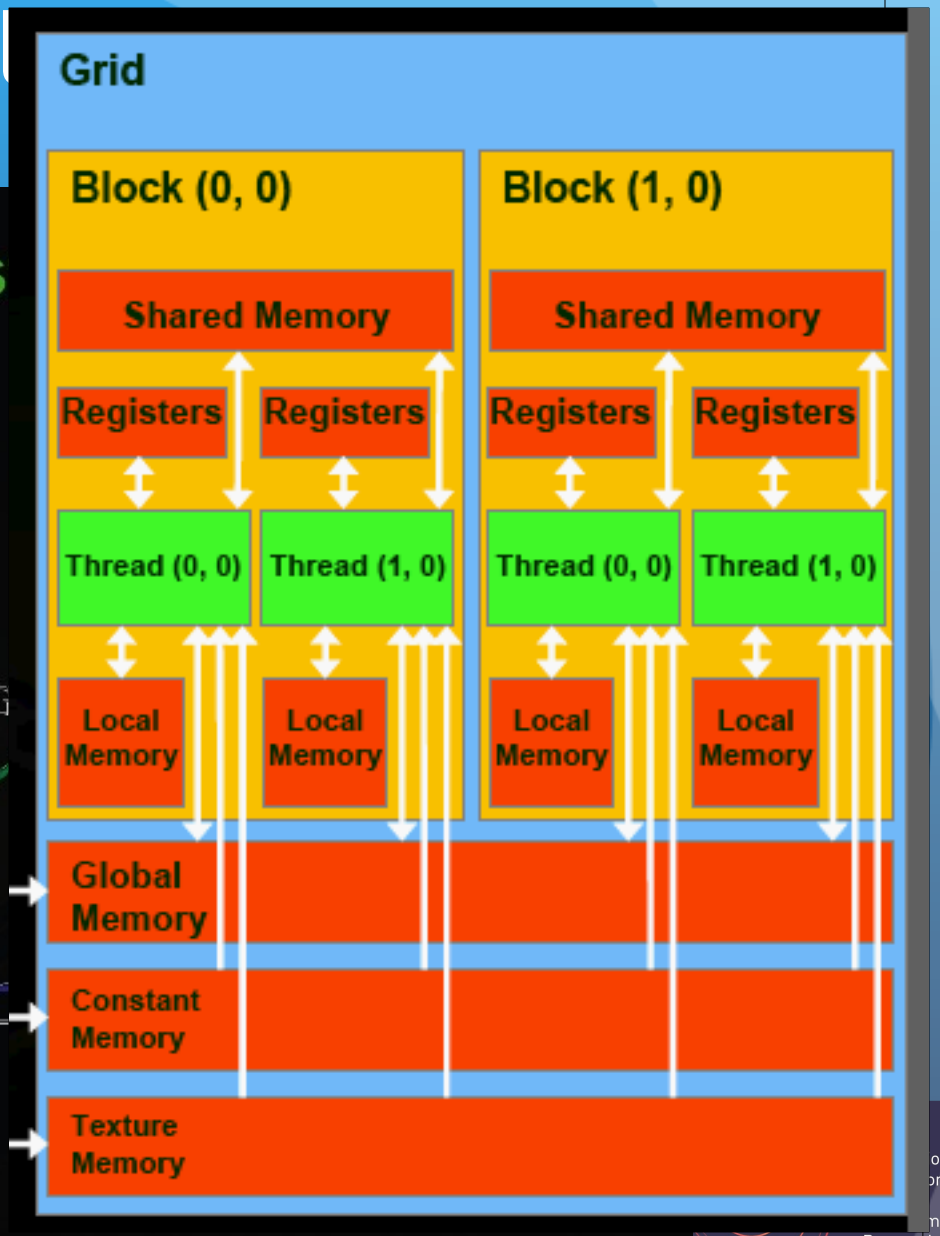


gures from NVIDIA

ICRAR International Centre for Radio Astronomy Research

GPUs – massivel
processing





onal
or
my
Research

GPUs – limitations

- GPUs sit on the PCI bus

  ∴ The data I/O bandwidth is limited

- GPUs have (had) limited processing power

  Previously no floats, doubles just now available

- Memory access needs careful handling

  Memory between different threads (i.e. pixels) has slow access

LBA plans towards GBps VLBI

- For ATCA it will be CABB based up to 2GHz (10bit)
- PKS/Mopra will use the DFB3 1GHz (8bit)
- Aiming to achieve 8-16 Gbps
- Correlation on DiFX or CABB

Chris will speak on this after lunch

GPU plans towards GBps VLBI

- Develop a auto-correlator (needed) for the current system

  Hardware must sit within current LBADR

- Develop fringe rotation & encoding on-card
- Replace integration-on-card with streaming to ethernet
- Massively increase processing power for next-gen eVLBI

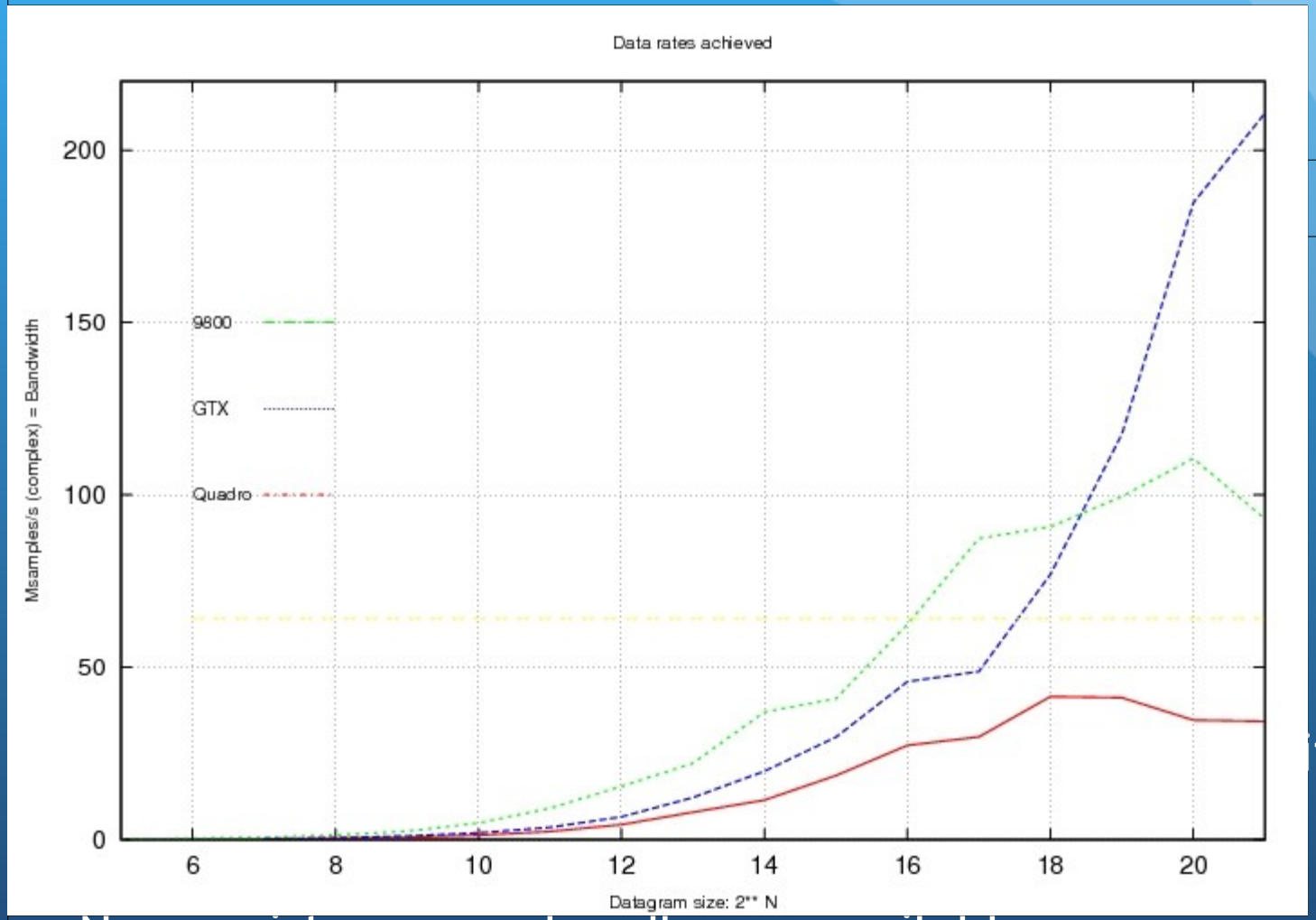
VLBI Correlation (alternative)

- IF chain (down-convert and filter)
- A2D conversion at 8-bit
- Channelise
- Fringe rotate
- Mitigate RFI
- Compress to 2-bit
- Record (or transmit)

On GPU

GPU steps so far

- Bare Bones System: Dual streams, transfer one channelise other. Main delay is in FFTs
- Adaption of LBA program **fauto** to **gauto**:
- Ugly, but working, single stream. Main delay is conversion of bits and memory coalescing
- Next steps: improve coding, improve coalescing, shift demangling onto card,
- New assistance, cuda-gdb now available

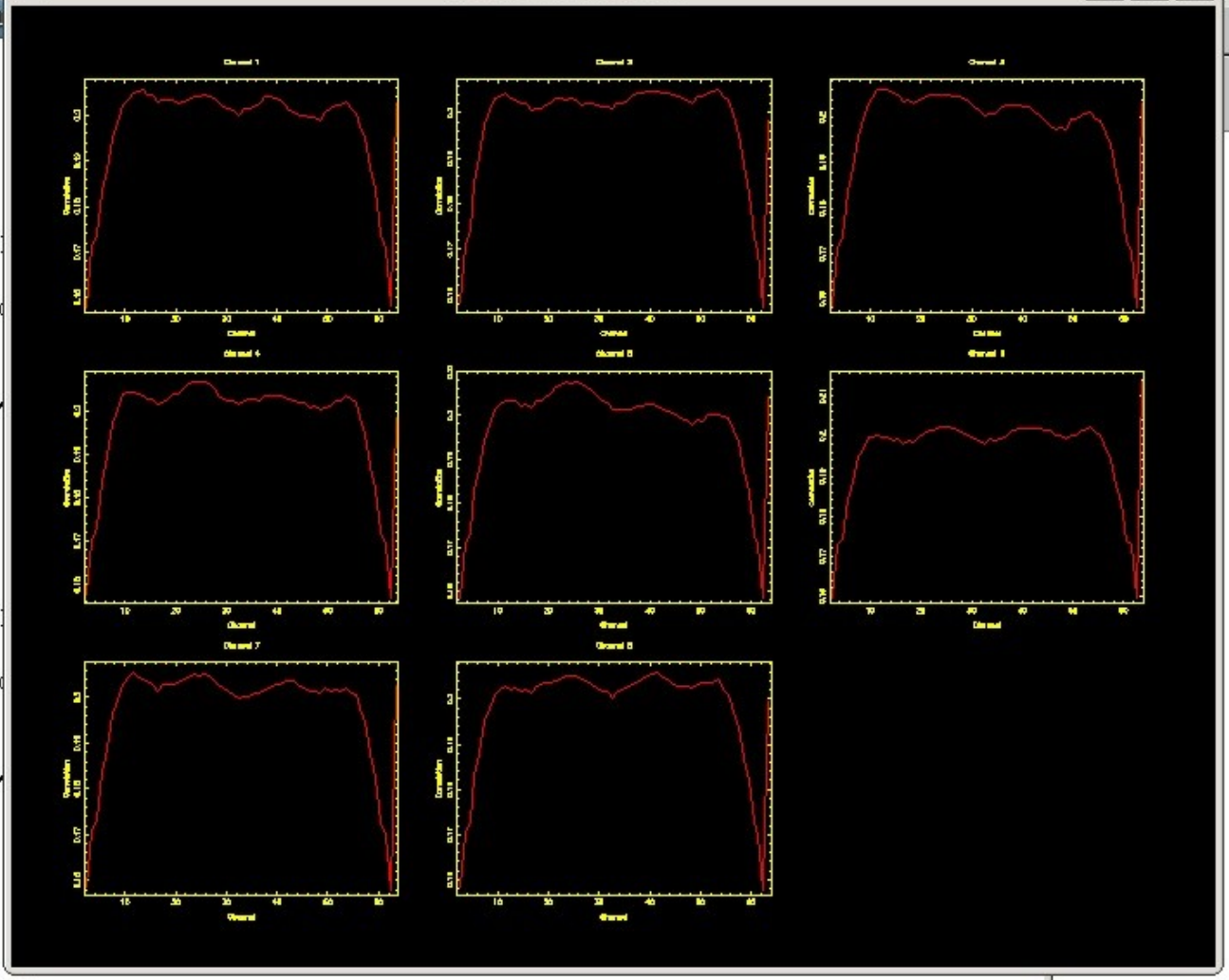


- New assistance, cuda-gdb now available

Applications Places System 2.00 17 °C Mon May 25, 5:40 PM US 105-ke

PGPLOT Serv

PGPLOT Window 2



```
.1ba
v293a_Pk_060_084220.1ba
Reading header
Plotting all channels
Bytes per integration = 256
Memory on Card 0.5
Allocated 33280 complex (8) to D_
as 8 * 64 * 64 +1
Total computational time= 9.8 sec
Total counts 1280.0e6
  17.3  32.7  32.7  17.3
astro-556:/usr/local/cuda/rdodson/
0.1ba
v293a_Pk_060_084220.1ba
Reading header
Plotting all channels
Bytes per integration = 256
Memory on Card 0.5
Allocated 33280 complex (8) to D_
as 8 * 64 * 64 +1
Total computational time= 9.8 sec
Total counts 1280.0e6
  17.3  32.7  32.7  17.3
astro-556:/usr/local/cuda/rdodson/
0.1ba
v293a_Pk_060_084220.1ba
Reading header
Plotting all channels
Bytes per integration = 256
Memory on Card 0.5
Allocated 33280 complex (8) to D_IN
as 8 * 64 * 64 +1
Total computational time= 10.4 seconds for 2500032 ffts
Total counts 1280.0e6
  17.3  32.7  32.7  17.3
astro-556:/usr/local/cuda/rdodson/vsib_util>
```

xterm [gpu_int... pgxwin [Terminal] gauto.c [NVIDIA ... [CudaRe... PGPLOT2

GPU steps in near term

- Develop our skills for Antenna based channelization: Now
- Providing a real-time auto-correlator: Now, but needs improvement
- Roll out 9880GT onto all LBA antennae: Costs small

  (<$2000)
- Outcomes from ARC grant to do this: November
- Outcomes from NVIDIA support for the same: Soon

In conclusions

One can improve the efficiency of the VLBI correlators

Main gain will be in achieving the ideal A2D conversion

This can be done with GPUs

ICRAR is developing their expertise in this area &

Converting existing software auto-correlators

Developing the code for data encoding/decoding

fringe-rotation