

CASE STUDY: Using Field Programmable Gate Arrays in a Beowulf Cluster

Mr. Matthew Krzych

Naval Undersea Warfare Center

Phone: 401-832-8174

Email Address: krzychmj@npt.nuwc.navy.mil

The Robust Passive Sonar (RPS) program has developed a 16-node Beowulf cluster with integrated Field Programmable Gate Arrays (FPGAs) for a computationally intensive signal processing application. The use of FPGAs within the cluster significantly increases the processing capacity of the cluster at low cost. They also have an added benefit of having a relatively small footprint and therefore having minimal impact on space requirements.

The RPS system provides a real-time processing capability that passively localizes an acoustic noise source in three dimensions including bearing, range, and depth. Using the bearing and range estimates, a geographic situation plot is developed recording contact position, course and speed. The application is computationally demanding requiring 500 Gigafllops per second or half a Tera-FLOP of sustained processing in order to evaluate the entire search region. This processing capacity has been achieved through the use of five FPGA boards which allow the system to 'beamform' to 10 million points in space.

The system utilizes a set of desktop computers including AMD 1900 series and Intel Pentium III processors interconnected via Myrinet and Ethernet. The Myrinet network provides a high bandwidth interconnect (1.28 Gbits/sec sustained) which, when used with the Message Passing Interface (MPI) protocol, provides tight coupling of processors between platforms and allows data to flow through the system in a pipeline manner.

Hosted within five desktop computers are PCI based FPGA boards that implement the signal processing kernel. Each board contains two Xilinx FPGA chips; one dedicated for off-board communications and the other for processing the application kernel. Each board provides 50 GFLOPS of compute power and interfaces with the desktop computer via the internal PCI bus of the computer.

The application implemented for this system has a number of characteristics that make it well suited for FPGAs. a) The application is extremely parallel in nature and therefore can be easily partitioned. This is supported by FPGAs that provide multiple memory ports as well as multiple processing elements per chip reducing bottlenecks. b) Data sets are extremely large which also exploit the multiple memory ports and processing units of the chip. c) Resolution of the data is low, generally less than 12 bits, increasing the efficiency of the FPGA hardware. d) The application kernel requires the same, relatively small, set of commands to be continuously executed in a fixed sequence. This greatly simplifies the control logic of the FPGA board and makes it amenable to pipelining.

This presentation will investigate the issues associated with developing and using a Beowulf cluster. A parallel, heterogeneous computing environment with embedded FPGAs provides

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 01 FEB 2005		2. REPORT TYPE N/A		3. DATES COVERED -	
4. TITLE AND SUBTITLE CASE STUDY: Using Field Programmable Gate Arrays in a Beowulf Cluster				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Undersea Warfare Center				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited					
13. SUPPLEMENTARY NOTES See also ADM00001742, HPEC-7 Volume 1, Proceedings of the Eighth Annual High Performance Embedded Computing (HPEC) Workshops, 28-30 September 2004 Volume 1., The original document contains color images.					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 19	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

many unique challenges including run-time configuration, system management, and efficient parallel programming. These critical issues along with performance and lessons learned will be highlighted.

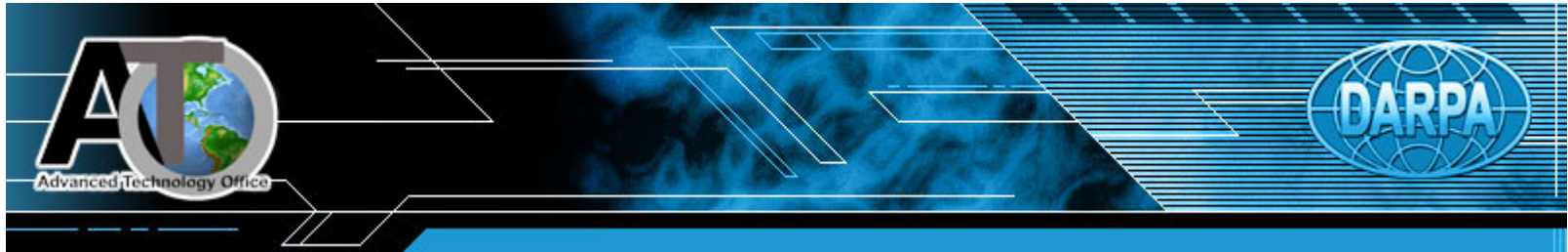
The RPS program is a DARPA funded project.



USING FIELD PROGRAMMABLE GATE ARRAYS IN A BEOWULF CLUSTER

Matthew J. Krzych
Naval Undersea Warfare Center

Sponsor



- ☐ **DARPA - Advanced Technology Office**
 - ☐ **Robust Passive Sonar Program**
 - ☐ **Program Manager – Ms. Khine Latt**

Problem Description

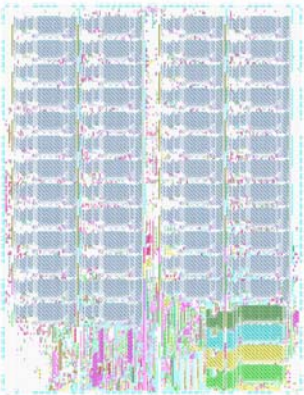
- ❑ Building an embedded tera-flop machine
 - ❑ Low Cost
 - ❑ Small footprint
 - ❑ Low power
 - ❑ High performance
- ❑ Utilize commercially available hardware & software
- ❑ Application:
Beamform a volume of the ocean
 - ❑ Increase the number of beams
from 100 to 10,000,000



On February 9, 2000 IBM formally dedicated [Blue Horizon](#), the teraflops computer. [Blue Horizon](#) has 42 towers holding 1,152 compute processors, and occupying about 1,500 square feet. Blue Horizon entered full production on April 1, 2000.

Approach

- Compile matched field “beamformer” onto a chip



- ← Specialized circuitry
 - 10x over Digital Signal Processors
 - 100x over General Purpose Processors



- DARPA Embedded High Performance Computing Technology

- » Adaptive Computing FPGAs
- » Message Passing Interface (MPI)
- » Myrinet – High Speed Interconnect



Beowulf Cluster



Sustained 65 Gflops with FPGA's

System Hardware

16 Node Cluster

- AMD 1.6 GHz and Intel Pentium 2.2 GHz
- 1 to 4 GBytes memory per node
- 2U & 4U Enclosures w/ 1 processor per enclosure
- \$2,500 per enclosure ¹.

8 Embedded Osiris FPGA Boards

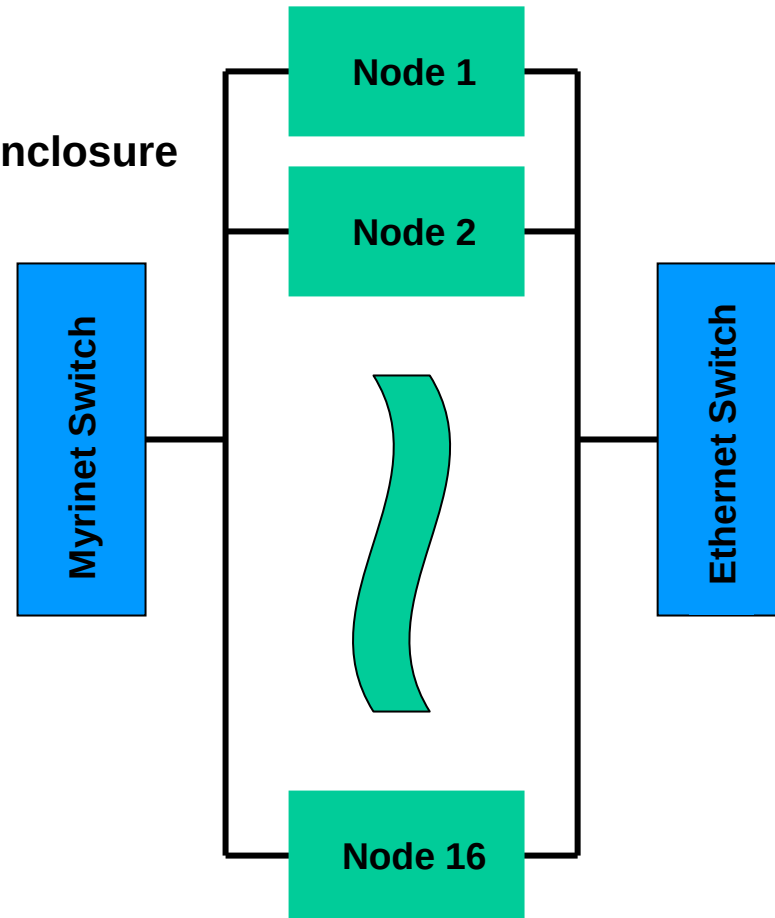
- Xilinx XC2V6000
- \$15,000 per board ¹.

Myrinet High Speed Interconnect

- Data transfer: ~250 MBytes/sec
- Supports MPI
- \$1,200 per node ¹.
- \$10,500 per switch ¹.

100 BASE-T Ethernet

- System control
- File sharing

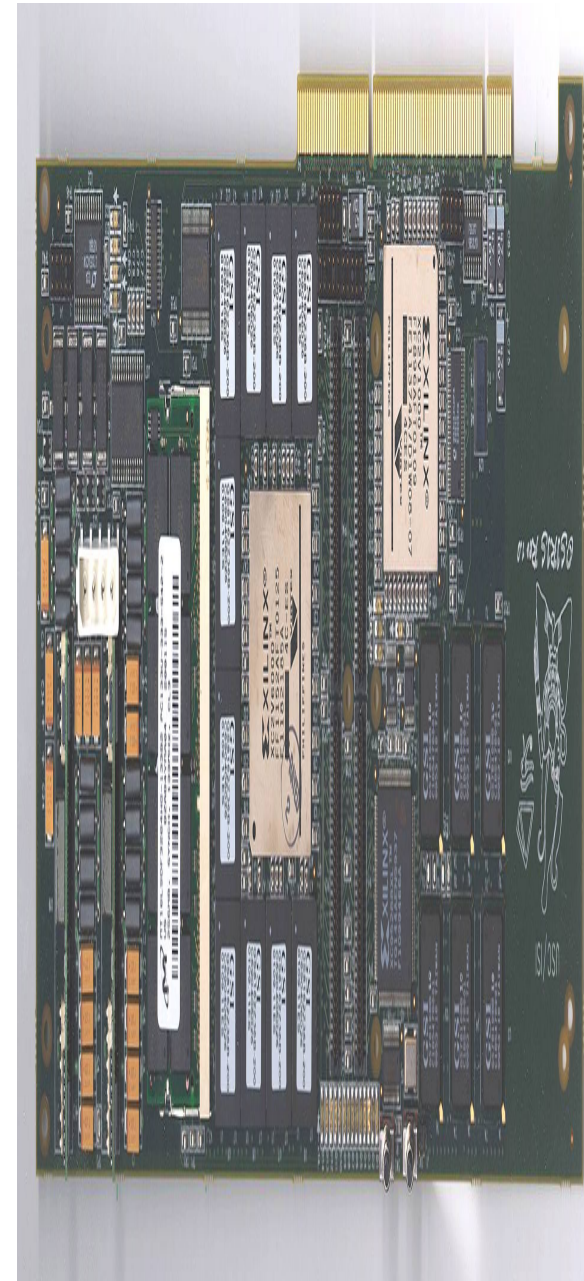


➡ **Total Hardware Cost¹: \$190K** ⬅

¹. Cost based on 2001 dollars. Moore's Law asserts processor speed doubles every 18 months. 2004 dollars will provide more computation or equivalent computation for fewer dollars.

Hardware Accelerator

- ❑ **Osiris FPGA board**
 - ❑ Developed by ISI / USC
 - ❑ Sponsored by DARPA ITO Adaptive Computing Systems Program
 - ❑ 256 Mbyte SDRAM
- ❑ **Xilinx XC2V6000 chip**
 - ❑ ~ 6,000,000 gates
 - ❑ 2.6 Mbits on chip memory
 - ❑ 144 18 by 18 bit multipliers
- ❑ **PCI bus 64 bit / 66MHz Interface**
- ❑ **Sustained 65 Gflops**
- ❑ **Numerous commercial vendors**





System Software

- ☐ **Multiple programming languages used:**
 - ☐ C, C++, Fortran77, Fortran90, Matlab MEX, VHDL

- ☐ **Message Passing Interface (MPI)**

- ☐ **Red Hat Linux v7.3**

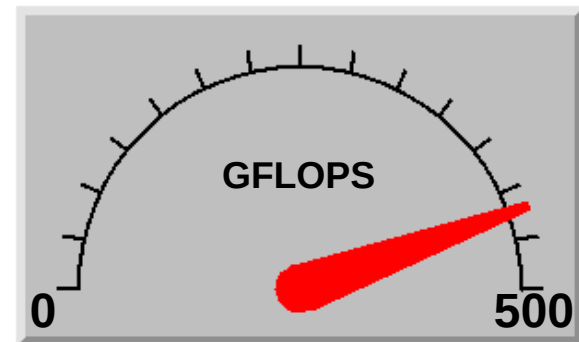
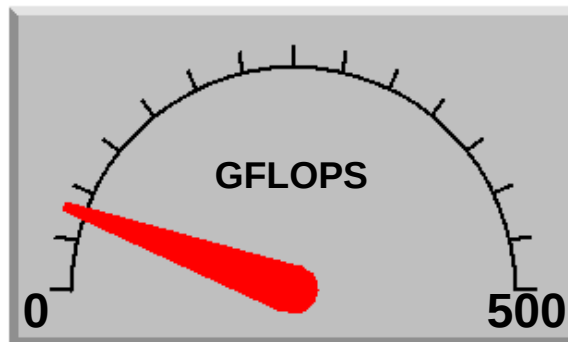
- ☐ **Matlab**
 - ☐ **System displays**
 - Interface to MPI via shared memory
 - ☐ **Post processing analysis**

- ☐ **Run-time cluster configuration**
 - ☐ Supports run-time cluster configuration (hardware & software)

Computational Performance

- ❑ **WITHOUT hardware accelerator**
 - ❑ 16 nodes (2.2 GHz)
 - ❑ 5 GFLOPS sustained
 - Single precision

- ❑ **WITH hardware accelerator**
 - ❑ 8 FPGA boards
 - ❑ 500 GFLOPS
 - Fixed point
 - Pipelining
 - Parallelism



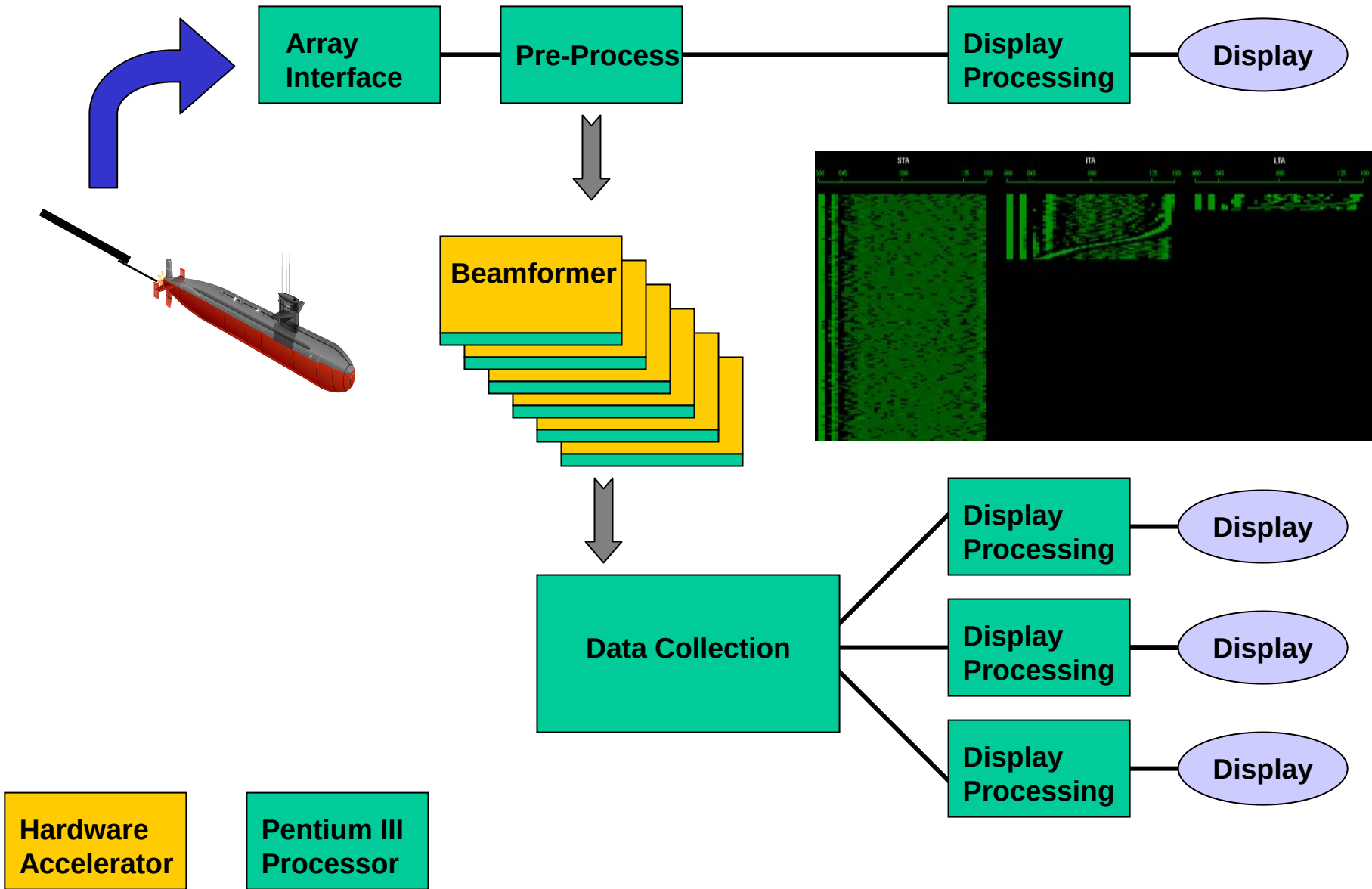
Run-time Cluster Configuration

- ❑ **Developed in-house**
 - ❑ Exploits MPI communication constructs
 - ❑ Uses Linux shell scripts & remote shell command 'rsh'
- ❑ **Based on user specified configuration**
 - ❑ Configuration defined in text file
- ❑ **Allocates system resources at start-up**
 - ❑ Identify hardware availability
 - ❑ Identify which functionality to execute
- ❑ **Map functionality to specific nodes at run-time**

Functional Description File

```
=====
FUNCTION          NUMBER          VALID HOSTS
***
array_if23        1                x0
frontend          1                x0
disp_conv         0                xb
mfp               3                x3, x1, x2, xa
collector         1                xa
disp_mbtr         1                xc, xb
disp_mrtr         1                xb, xc
```

Sonar Application



Benefits

- ❑ High performance (500 GFLOPS), low cost solution (<200K)
- ❑ FPGAs
 - ❑ Performance (100x increase)
 - ❑ Small footprint (PCI board)
 - ❑ Power
- ❑ Beowulf Cluster
 - ❑ Flexibility /robustness
 - Supports heterogeneous hardware
 - Run-time selection of processors
 - Run-time selection of functions to instantiate
 - Run-time selection of system parameters
 - ❑ Scalability
 - Add / remove hardware assets
 - Add / remove functionality
- ❑ MPI
 - ❑ Facilitates flexibility & scalability
 - ❑ Runs on multiple hardware platforms & operating systems
 - ❑ Supports multiple communication schemes
(point-to-point, broadcast, etc.)



Issues

❑ FPGAs

- ❑ Lengthy development time
- ❑ Difficult to debug
- ❑ Bit file tuning: sizing, placement, & timing
- ❑ Bit files are NOT easily modified
- ❑ Bit files are NOT portable

❑ Beowulf Cluster

- ❑ Functional mapping
 - Flexibility must be programmed in
- ❑ Performance optimization
 - Identifying bottlenecks
 - Load balancing
- ❑ Configuration Control
 - System maintenance
 - Keeping track of assets
 - Asset compatibility
- ❑ Tool availability



Summary

- ❑ **Computationally demanding sonar application successfully implemented**
 - ❑ Could NOT have been implemented using traditional methods
- ❑ **16 node Beowulf cluster developed using 8 embedded FPGAs**
 - ❑ Fits in 1 ½ standard 19" racks
 - ❑ Hardware costs < \$200k
 - ❑ FPGA software tools < \$40k
- ❑ **500 GFLOPS sustained processing achieved**





USING FIELD PROGRAMMABLE GATE ARRAYS IN A BEOWULF CLUSTER

Matthew J. Krzych
Naval Undersea Warfare Center

Problem Description

- ❑ Building an embedded tera-flop machine

- ❑ Low Cost
- ❑ Small footprint
- ❑ Low power
- ❑ High performance

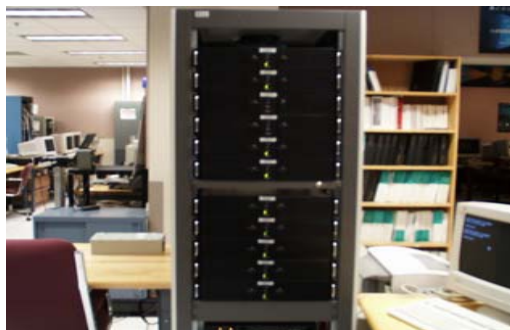
- ❑ Utilize commercially available hardware & software

- ❑ Application:
Beamform a volume of the ocean

- ❑ Increase the number of beams
from 100 to 10,000,000



On February 9, 2000 IBM formally dedicated [Blue Horizon](#), the teraflops computer. [Blue Horizon](#) has 42 towers holding 1,152 compute processors, and occupying about 1,500 square feet. Blue Horizon entered full production on April 1, 2000.



**Beowulf
Cluster**

System Hardware

16 Node Cluster

- AMD 1.6 GHz and Intel Pentium 2.2 GHz
- 1 to 4 GBytes memory per node
- 2U & 4U Enclosures w/ 1 processor per enclosure
- \$2,500 per enclosure ¹.

8 Embedded Osiris FPGA Boards

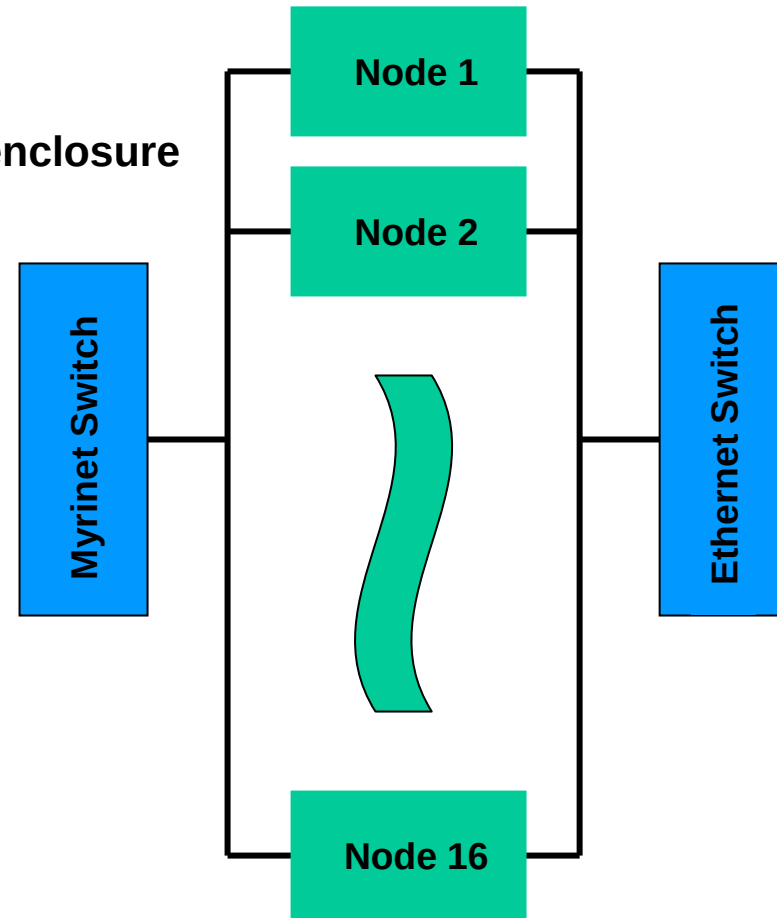
- Xilinx XC2V6000
- \$15,000 per board ¹.

Myrinet High Speed Interconnect

- Data transfer: ~250 MBytes/sec
- Supports MPI
- \$1,200 per node ¹.
- \$10,500 per switch ¹.

100 BASE-T Ethernet

- System control
- File sharing



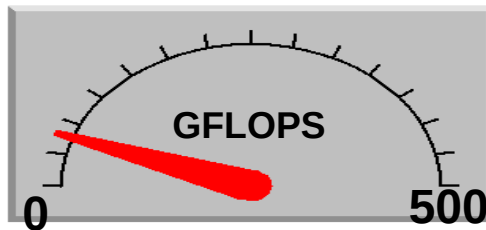
➡ Total Hardware Cost¹: \$190K ⬅

¹. Cost based on 2001 dollars. Moore's Law asserts processor speed doubles every 18 months. 2004 dollars will provide more computation or equivalent computation for fewer dollars.

Lessons Learned

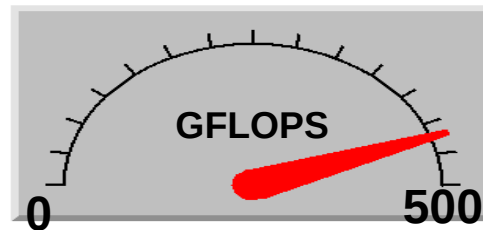
❑ WITHOUT hardware accelerator

- ❑ 16 nodes (2.2 GHz)
- ❑ 5 GFLOPS sustained
 - Single precision



❑ WITH hardware accelerator

- ❑ 8 FPGA boards
- ❑ 500 GFLOPS
 - Fixed point
 - Pipelining
 - Parallelism



❑ Beowulf Cluster

- ❑ Flexibility / robustness
 - Supports heterogeneous hardware
 - Run-time selection of processors, functions, & system parameters
- ❑ Scalability
 - Add / remove hardware assets
 - Add / remove functionality

❑ MPI

- ❑ Facilitates flexibility & scalability
- ❑ Runs on multiple hardware platforms & operating systems
- ❑ Supports multiple communication schemes (point-to-point, broadcast, etc.)