Outline

• Heterogeneous HPC
• Lifecycle
• Applications
• Future systems
• Recognition
• Summary
Extending the Affordability Trend

**CHALLENGE**
Develop and Incorporate the Most Affordable Embedded Information Technology Available

**APPROACH**
- Leverage Commercial Investments In Computer Architectures
- Develop Portable Embedded DoD Applications using middleware standards
- Leverage DARPA, and Other DoD Efforts In Emerging Architectures

**PAYOFF**
- 3-7 year leap ahead in performance and affordability
- TeraOPS class performance in SWAP that supports embedded HPC experiments

Projection:
- 2007: 60 trillion flops/$M
- 2010: 360 trillion flops/$M
Tech Challenge: HPC to the Field

Software Technologies
- Parallel MATLAB
- Message Passing Interface
- Data Reorganization Interface
- Vector Signal Image Processing Library
- Cluster Integration Toolkit

Hardware Technologies
- Embedded HPC
- Advanced Packaging
- Power Efficient Processing
- Data Intensive Architecture
- Adaptive Computing Systems
- Polymorphic Computing Architecture

Command, Control, and SIP Algorithms
- Adaptive Interference Mitigation
- Real Time Decision Support
- Recognition & Identification
- Detection and Tracking
- Clutter Characterization
- Transform Techniques
- Distributed Information Enterprise Modeling & Simulation
- Compression and Coding
Heterogeneous HPC Vision

- **Thesis:** The combination of high-end server nodes configured in a Beowulf cluster with state-of-the-art FPGA accelerator boards to deliver >10X performance and affordability improvements on key applications and speed transition to the field.

- **Exploitable Technology Trends:**
  - Lower cost
  - Lower size, weight, power
  - Outstanding peak performance
  - Programming environment progress, but challenges remain!

- **Action:** Procured an initial Heterogeneous HPC to quantify advantages, demonstrate new military applications, and work programming issues.
System Specifications

• Proposed
  – 288 GFLOPS Pentium/Athlon + 600 GFLOPS or
  – 34 TOPS FPGAs (demonstrated)
  – 48 GB Memory
  – 3.36 TB Storage

• Delivered
  – 422 GFLOPS Pentium
  – 34+ TOPS FPGAs (demonstrated)
  – 192 GB Memory
  – 3.8 TB Storage
## Heterogeneous HPC

### System Performance

<table>
<thead>
<tr>
<th>Nodes</th>
<th>System Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>48</td>
<td>192 GBytes DRAM</td>
</tr>
<tr>
<td></td>
<td>3.8 TBytes Disk</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FPGA Gates</th>
</tr>
</thead>
<tbody>
<tr>
<td>576M</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FIR GigaOPS</th>
<th>34000 (16 bit)</th>
</tr>
</thead>
</table>

**Peak Floating Point Performance**
- 845 (SP) or 422 (DP) GFLOPS

**Peak Bisectonal Bandwidth**
- 12 GBytes/sec

### Software Development

<table>
<thead>
<tr>
<th>Host Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hat Linux 7.3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compilers</th>
</tr>
</thead>
<tbody>
<tr>
<td>gcc/g++/g77/F90</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Communications Libraries</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPI</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Signal Processing Libraries</th>
</tr>
</thead>
<tbody>
<tr>
<td>VSIPL/ VSIPL++/ATLAS/SPIRAL</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FPGA Development Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Annapolis CoreFire, Synplicity, Xilinx ISE, ModelSim</td>
</tr>
</tbody>
</table>

### HHPC two rack system with two nodes per 2U height density
- **Cost**: $1M
- **Online**: Feb 2003
Compute Node

Commodity Compute Node:

- Dual 2.2 GHz Pentium 4 Xeon (512KB L2 Cache)
  17.6 SP GFLOPS peak per node
- Wildstar II FPGA Board - 2 x 6M gates
- Myrinet network interface
- 4 GB DDR SDRAM
- 80 GB Ultra ATA disk
- Intel E7500 Chipset
Wildstar II FPGA Subsystem

FPGA Subsystem: Annapolis Microsystems Wildstar II

- (2) Xilinx Virtex II 6000-4 (12M gates total) [XC2V6000-4FF1517C]
- (6) 150 MHz DDR SSRAM 1MB banks per FPGA
- (1) 100 MHz DDR SDRAM 64MB bank per FPGA
- Dynamic Message fabric or Data Acquisition: (1) Wildstar LVDS/LVTTL I/O Card per board (dual 50-pin “Champ” external connectors)
Programming Environment

**Operating System:** Red Hat Linux 7.3

**Development Tools**

- GNU Tools (gcc, g++, g77, gdb, etc.)
- Intel FORTRAN 90

**Libraries**

- MPI (MPICH-GM)
- VSIPL (AFRL Cluster Enhanced version, DARPA ACS VSIPL)
- VSIPL++ 0.1 Experimentation, moving to 1.0
FPGA Design Environment

Design Tools

- Annapolis CoreFire
- VHDL
- Synplify
- Xilinx ISE tools

Design Methodologies

- Traditional VHDL->Synthesis->Backend
- CoreFire functional block construction
- VSIPL HW/SW Co-design
Remote Access

- **Network**
  - T3 link to HPCMO Defense Research Engineering Network (DREN)
  - Classified DREN access being added

- **User Accounts**
  - Available to DoD RDT&E Community and their contractors with no fee-for-service
Lifecycle

- # Users: 141
- # Projects: 28
- Downtime: 0-10 system hr. most months, max 55 hr./mo.
Proposed Application Areas

- Imaging
  - Hyperspectral Image Exploitation
  - Infrared Image Processing
  - Bistatic Imaging
  - Multisensor Image Processing
- HPC for C2 Decision Support
- Distributed Information Enterprise Modeling and Simulation
- Polymorphic Computing
- Support of HPC in the Field
- Real-Time Implementation of STAP for Radar and Sonar
Imaging Applications

- Hyperspectral Imaging Portfolio
- Space-Based Infrared System (SBIRS) – High
- Vector Signal Image Processing Library – VSIPL++
- Backprojection for Synthetic Aperture Radar
Objective: To develop a scalable, portable, high performance computing software framework with 8 algorithms for rapidly accessing and processing hyperspectral data.

Impact
- Near Real-Time Access to Hyperspectral Imagery Data for Battlespace Awareness
- Imagery Products for Intelligence Analysts and Battlefield Decision Makers
- Increased Capability to Accurately Locate the Enemy and Precisely Attack Key Enemy Forces or Capabilities
- More Accurately Assess Level of Success

Approach
- Leverage serial DoD hyperspectral imagery codes
- Use DoD HPC's to rapidly process raw hyperspectral data to produce imagery products
- Client for Broadsword and then JBI to provide rapid query, publish and subscribe
- Secure webserver application interface
- Flexible to incorporate future algorithms

Status
- 8 Codes Tested and Integrated with Framework
- Successful BROADSWORD Demo - 4Q FY02
- JBI Core Services Demonstrated - 2Q FY03
- Successful beta review 1Q FY04
- Ported to HPC Platforms (Huinalu, HHPC, Brainerd, Coyote)

POC: Dr. Richard Linderman, 315-330-2208, Email: Richard.Linderman@rl.af.mil
Space-Based Infrared System (SBIRS) - High

• Debugging of complex subassemblies can be time consuming and expensive, especially without instrumentation for controllability and observability

• AFRL/IFTC compiling VHDL Signal Processing Array (SPA) simulation model
  – SBIRS-HI SPA is composed of several complex, redundant, fault-tolerant sub-systems
  – Benefits include system wide documentation, standardization, low and high granularity models

• Near term goal: SPA emulation on the HHPC to allow hardware in-the-loop real-time observability for payload integration
Vector Signal Image Processing Library (VSIPL++)

- Accomplished with a very early* version VSIPL++
  - Ported to VSIPL++ Wide Swath SAR: “Swathbuckler”
    - Generated a Cluster VSIPL++ library Optimizations (FFT)
  - FPASP VSIPL implementation
  - Created a VSIPL++ / FPGA I/O Interface
- Planning to do: (current released version)
  - Complete VSIPL++ cluster optimizations on selected vector functions
    - Collect the measurements
  - Study prospects for creating “open” and optimized VSIPL++ libraries for other processors (IA-64, Opteron, PowerPC, FPASP)
  - Use VSIPL++ in C4ISR Applications of HPEC

*Feb. 2003
• Distributed backprojection algorithm for reconstructing SAR images implemented on HHPC

• Makes use of both a scalable number of HHPC nodes and FPGA Hardware (Annapolis Wildstar II) on each node

• Global target image is partitioned spatially

• Partitions are processed in parallel by HHPC nodes

• Target image partitions are reconstructed on the FPGA hardware with a processing pipeline designed to match the PCI data transfer rate

• Implementation can process data as fast as it can be read from/written to file system. Implementation uses:
  • 33 HHPC nodes
  • Wildstar II boards on each node
  • Myrinet communication channels

• 26 times speed up compared to an efficient serial implementation run on a single HHPC node for 33 node implementation
Backprojection for SAR - Results

Single Target – High Reflectivity Coefficient

Three Targets – Ranging Reflectivity Coefficients

Eight Targets – Low Reflectivity Coefficients

- Images are 16km x 1km
  - 1m resolution in range and azimuth
  - reconstructed from synthetic data
- 33 nodes take 19 seconds to reconstruct an image of this size
- Lack of parallel file I/O capabilities of HHPC is bottleneck
  - Run-time is not linear function of number of nodes due lack of parallel file I/O

Run Time as a function of Parallelism

- Run Time as a function of Number of Processing Nodes
  - 0, 10, 20, 30, 40 nodes
  - Run Time in seconds: 0, 100, 200, 300, 400, 500, 600
HPC for C2 Decision Support

- Field Programmable Gate Array (FPGA) Acceleration of Joint Battlespace Infosphere (JBI) Pub-Sub Brokering
- Intrusion Database Modeling for Anomaly Detection
- Development of an FDTD Accelerator on the HHPC Cluster
- US Army Research, Engineering, and Development Command – Electronic Battlefield Environment Portfolio
- Distributed and Embedded Methods for Speed-Up of a Genetic Algorithm Approach to Solving the DNA Code Word Library Generation Problem
• Pub-Sub brokering is a core service of the Joint Battlespace Infosphere (JBI).
• The problem is to find matches between subscribers and an incoming publication.
• FPGA acceleration tested on one node of the HHPC.
  – For document with 700 characters and 14 string leaves, using FPGAs,
    • Parsing takes 14 µs at 50 MHz
    • Actual processing has been measured to be 45 µs
    • Data transfer dominates the overall processing time. Parallel predicate matching provides further acceleration.

  – In comparison, on the microprocessor, without using the FPGAs, the parse time alone is around 2 ms.

by Dr. Chun-Shin Lin, University of Missouri, Columbia
Intrusion Database Modeling for Anomaly Detection
AFRL: Dr. Keesook J. Han, IFGB

Objectives

- Develop an intrusion database model for interacting with the intrusion detection systems.
- Develop reliable anomaly detection algorithms.

Scientific/Technical Approach

- Structure the database for improving the retrieval rate of a huge amount of data.
- Implement an anomaly detection system that interacts automatically with the compact intrusion database in order to update the profiles.

Accomplishment

- Developed the New Intrusion Detection Compression Technology

Challenges

- Efficient Intrusion data capturing, modeling and management will be a solution to improving the performance of anomaly detection systems.
• Accelerate parallel Finite-Difference Time Domain (FDTD) code

• Full-wave computational electromagnetics solver
  • Applications: RF communication modeling, antennae patterns and interference

• Such an FDTD code was used as the basis for the MRTD code in former CHSSI project CEN5
  • with Dr. Linda Katehi, Dr. Costas Sarris, Dr. Donghoon Chun, Dr. Barry Perlman, Dr. Timothy Flemming,
Develop an HPC based RF Propagation Server with the performance and functionality to support SIGINT and communications Mission Planning.

Integrate data from EBE-01 and/or EBE-03 to support extension of the RFPS to Directional Antennas.

Provide infrastructure for integrating higher fidelity models in an operational framework.

Provide an interactive mechanism and HCI for users to reliably determine the relationship between expected run time, problem parameters and the number of processors employed on a given HPC system to provide predictable and consistent response time.

Mr. Richard Pei
EBE-04 PI
US Army CERDEC I2WD
Fort Monmouth, NJ
732-427-4820
Richard.Pei@us.army.mil

Beta testing- 4Q FY05- HHPC
Distributed Information Enterprise Modeling & Simulation (SOS-1)

Project Funded by
Dr. Andrew Mark
DOD High Performance Computing Modernization Office (HPCMO)

CHSSI Portfolio
System-of-Systems

Portfolio Management
Dr. J. Michael Barton
U. S. Army Developmental Test Command

• Modeling Framework to Simulate Performance of a Conceptual Enterprise
• Simulate System Enterprise at The “Information” Level
• Identify, Quantify, And Evaluate Protocols, Processes, and Core Functions Comprising an Operational Enterprise
• A Generalized Information Architecture Simulator
• Project Successfully Completed
Developed a single PC version of Genetic Algorithm (GA) based Optimizer code and applied it to the NP-complete problem of parameterizing non-linear ODE models of Ag-Ab binding on surfaces.

Developed a distributed (C, MPI, cluster) version of GA Optimizer and applied it to the problem of DNA Code Word Library generation, another NP-complete problem relevant to in-house and AFOSR sponsored work in IFTC Bio-molecular Computing.

Evaluating FPGA version for extreme speed-up of LvM calculation during constraint checking in the DNA Code Word Generation problem.

– Speed Profiling: 98%+ time evaluating LvM
  – LvM all integer, can calculate in FPGA
  – GA island model is also a natural for FPGA implementation
  – ‘back of the envelope’ FPGA speed estimate 0.1 us vs 10 us (1000x)
  – writing synthesizable VHDL version for testing (poster at MAPLD ’05)
Polymorphic Computing

• Parallel Discrete Event Simulation
Parallel Discrete Event Simulation (PDES):
- JBI speedup by accelerating SPEEDES
- Designed an I/O backplane to provide direct connectivity between the 24 FPGA boards (Each board has 2 FPGAs)
- Identified important simulation functionality that can benefit from execution on the FPGAs and the utilization of the I/O backplane
  - Several promising opportunities identified
  - Global Virtual Time implementation on FPGA underway

- Important capability at core of several DoD projects such as:
  - Wargaming (e.g., ONR’s Naval Simulator)
  - Complex system analysis (e.g., JBI, VHDL simulation)

- Parallel simulation involves the execution of a *single* simulation program on a collection of processors (e.g., a cluster).

Dr. Nael Abu-Ghazaleh SUNY Binghamton
1. Using an FPGA enabled cluster to accelerate simulation
   - Migrating functionality to the FPGA
     • Currently testing custom all-to-all board to have the FPGAs talk to each other without host interference
     • Global Virtual Time on FPGAs implemented
     • Plan to test this summer

2. Exploring Alternative Communication Subsystem Support
   • Peer-to-peer instead of centralized server
     • Up to 20% speedup on JBI benchmark
   • MPI instead of sockets
     • Up to 75% speedup on JBI benchmark
   • Using Myrinet instead of Gigabit Ethernet
     • Up to 100% speedup on JBI benchmark
   • Total of 3x speedup on JBI benchmark
PDES – Algorithmic/ Simulator Improvement

• Developed a time-bounded (instead of event-bounded) synchronization algorithm
  – Preliminary results: up to 3x speedup on JBI model
  – Need to adaptively derive appropriate time bound
    • Work in progress

• Adaptive simulator configuration (in progress)
  – Many parameters and configuration options have major influence on performance. Currently statically set.
  – Load balancing has major effect on simulator performance
    • Verified the presence of large imbalances in our models
Support of HPC in the Field

• Support to International TTCP Wide-Swath High Resolution SAR Image Formation Field Experiments

• Deployment of Seafarer and Mach2 Clusters to SPAWAR and ASC, respectively, in support of the Distributed Interactive HPC Testbed (DIHT) Effort

• Deployment of HHPC nodes for demonstrations at SC 2004

• On-line support for field demonstrations of HIE testbed at OOPSLA and of Space-Based Radar at Boeing and JPL
Goal:

• Demonstrate affordable, continuous real-time, wide-swath, high-resolution SAR image formation, compression, and storage in flight tests.

Approach:

• Canadian test aircraft with SAR radar and algorithms, test flights
• US HPC with parallel software, Frontend Hardware, subsystem integration
• UK FPGA programming, Compression Algorithms
• AUS conops/system/data analysis

Military Impact:

• Affordable, High-resolution SAR area coverage
AFRL/IF DC: Distributed Interactive HPC Testbed (DIHT)

Unclassified Interactive HPC Network

- Distributed HPC’s
- Accessed by Authorized Users Anywhere on the DREN
- Time Critical Problems
- Co-Habitation of Interactive and Batch Processing
- Classified and Unclassified Network
-Multiplicity of Applications (100x JBI, Parallel Matlab, …)
• Space-Based Radar
**Radiation Tolerant Open Source Processor**

(VHDL and Software Development Environment is Open Source Available to Government Contractors)

**Impact**
- Reduced system complexity vs non-programmable approaches
- Reduced power consumption 30X
- Reduced system integration time
  - Open source facilitates debugging
  - Predictable execution times

**Processor Description**
- Fully programmable (C++, C) processor with Floating Point unit
- Optimized for power efficiency
- Complex instructions for common DSP functions like FFT
- Writeable control store for new instructions and debugging
- SEU mitigation features
  - EDAC on memory
  - triple voting or three processors option
  - Hardware CRC checks on messages

**Design Flow:**
- **Simulation**
  - Execution Speed: ~ 10 Hz
  - Time to Change: Minutes
  - Ideal for Initial Processor Simulations, Slow but Very Detailed
- **ISA (C) Simulation**
  - Execution Speed: ~ 1 MHz
  - Time to Change: N/A
  - Debugging Software, Very Portable, Rough Model of Hardware
- **Emulation**
  - Execution Speed: ~ 20 MHz
  - Time to Change: ~ 200 MHz
  - Finding Complex Bugs, Early Demonstrations, Optimization
- **ASIC**
  - Execution Speed: ~ 250 MHz
  - Time to Change: Months
  - Rad-Tolerant Highest Speed Lowest Power Costly to Fix

**Program Description**

<table>
<thead>
<tr>
<th>Program</th>
<th>AESOP 0.18 μm</th>
<th>Trusted Foundry 0.13 μm</th>
</tr>
</thead>
<tbody>
<tr>
<td>On-Chip Mem Type</td>
<td>SRAM</td>
<td>DRAM</td>
</tr>
<tr>
<td>Independent Mode Size</td>
<td>1 MByte</td>
<td>4 MBytes</td>
</tr>
<tr>
<td>Voted Mode Size</td>
<td>3 MBytes</td>
<td>12 MBytes</td>
</tr>
</tbody>
</table>

AESOP chips  Sept 2005, Trusted Foundry chips Dec 2005

POC: John Rooks AFRL/IFTC 315-330-2618
Emulation of a SBR On-Board processor

- Required an early testbed to reflect Technology update for processor optimization and allow closing the debugging loop without multi-million dollar chip fabrications
- Heterogeneous HPC (HHPC)
- Emulate the Processor using the HHPC
  - One Dual processor per FPGA
- Same VHDL for ASIC Simulation, Synthesis for FPGA Emulation, and Synthesis for the ASIC

Run your code early at 10% of full speed on up to 96 emulated processors
Future Systems

- SRC Mapstation Development Environment System
  - Dual µprocessor board
  - MAP with 2 Xilinx XC2VP100 FPGAs
  - Linux OS, SRC Carte Runtime & Development
  - C, C++, and Xilinx ISE
  - Due September 2005

- Need for Larger Reconfigurable System Identified
Recognition

• HPCMP Hero Awards - 2 of 4 in 2004 to AFRL/IF
  – Dr. Scott Spetka, Technical Excellence: Key technical contributor to success of HIE Portfolio
  – Virginia Ross, AFRL/IF DC Manager, for Innovative Management in running DC

• MVEEC Organization Productivity Award: George Ramseyer, HIE Portfolio

• IEEE Fellow, AFRL Fellow: Richard Linderman

• SAB review AFRL/IF rated world class in Advanced Computing Architecture
Summary

• **Unique Heterogeneous HPC** system funded by HPCMP has been available to the community since Feb 2003

• **New R&D facilitated** by this unique resource is leading to high impact success stories

• **Available** to entire DoD RDT&E community for support of local applications as well as remote access over the internet

• **Broad Applicability** – Signal/Image Processing, Test and Evaluation, Command and Control, and other applications

• **Challenges**, such as programming environment productivity, have been confronted—some R&D involved

• **Excellent prospects** for future, larger Heterogeneous HPCs as payoffs continue to be demonstrated