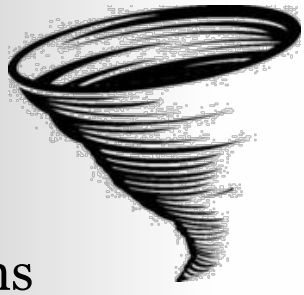


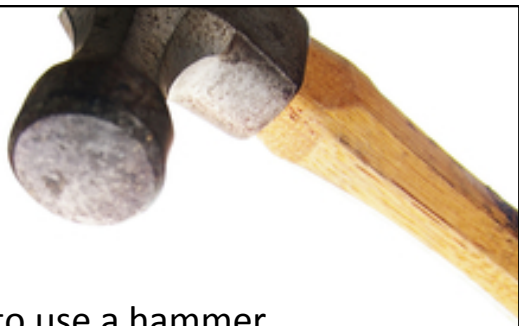
Parallel Paradigms Whirlwind Tour

Dirk Colbry, Research Specialist
Institute for Cyber-Enabled Research

MICHIGAN STATE UNIVERSITY

ICER

Purpose



"If you only know how to use a hammer, everything looks like a nail" – author unknown

- This lecture is to show you there are other tools

Agenda

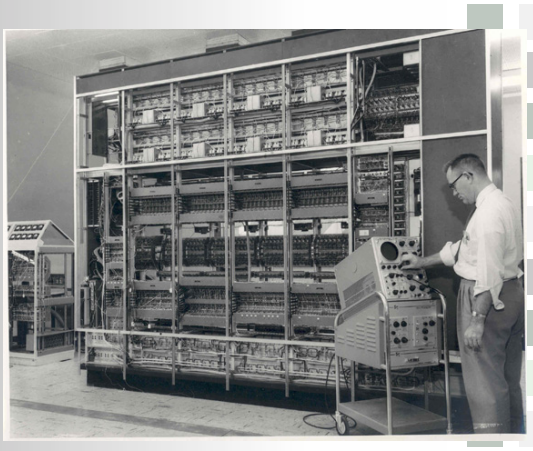
- Intro to me and where I work
- Example Computational Research Problems
- Types of communication

MICHIGAN STATE UNIVERSITY

ICER

1957 MISTIC Mainframe

- MSU's first mainframe
- Hand built by grad students
  - Dick Reid
  - Glen Keeney



MICHIGAN STATE UNIVERSITY

ICER

After MISTIC

- 1957 MISTIC
- 1963-1973 CDC 3600
- 1967 Computer Science Department
- 1968 CDC 6500
- 1971 MERIT
- 1978 Cyber 750
- **2004 HPCC**
- **2009 ICER**

MICHIGAN STATE UNIVERSITY

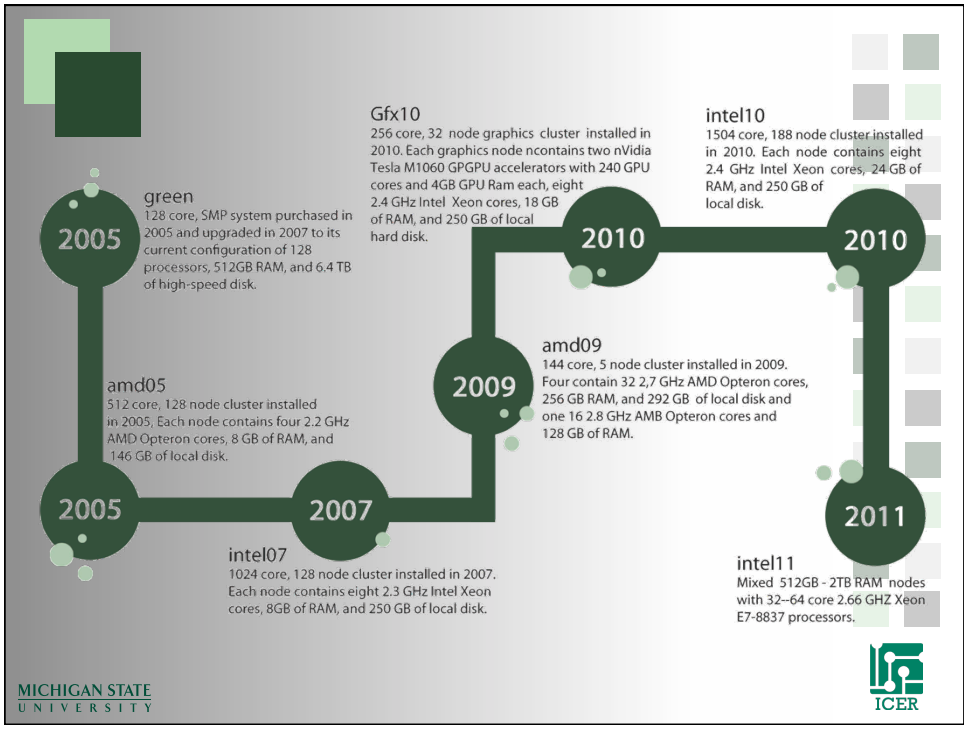
ICER

2004 MSU HPCC

- Provide a level of performance beyond what you could get and reasonably maintain as a small group
- Provide a variety of technology, hardware and software, that would allow for innovation not easily found

MICHIGAN STATE UNIVERSITY

ICER



2009 iCER

The Institute for Cyber Enabled Research(iCER) at Michigan State University (MSU) was established to coordinate and support multidisciplinary resource for computation and computational sciences. The Center's goal is to enhance MSU's national and international presence and competitive edge in disciplines and research thrusts that rely on advanced computing.

MICHIGAN STATE UNIVERSITY

ICER

iCER Research Specialist

- Me
  - Research Consulting
  - HPCC Programming
  - Proposal Writing
  - Training and Education
  - Outreach

MICHIGAN STATE UNIVERSITY

ICER

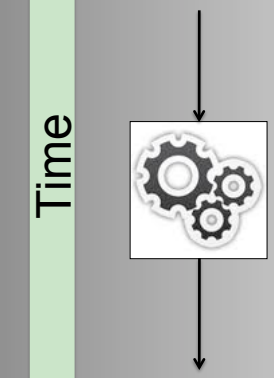
Agenda

- Intro to me and where I work
- Example Computational Research Problems
- Types of communication

MICHIGAN STATE UNIVERSITY

ICER

Scientific Programming



Inputs (Data)

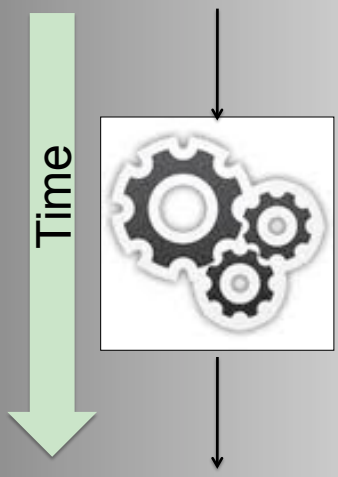
Computation

Outputs (Results)

MICHIGAN STATE UNIVERSITY

ICER

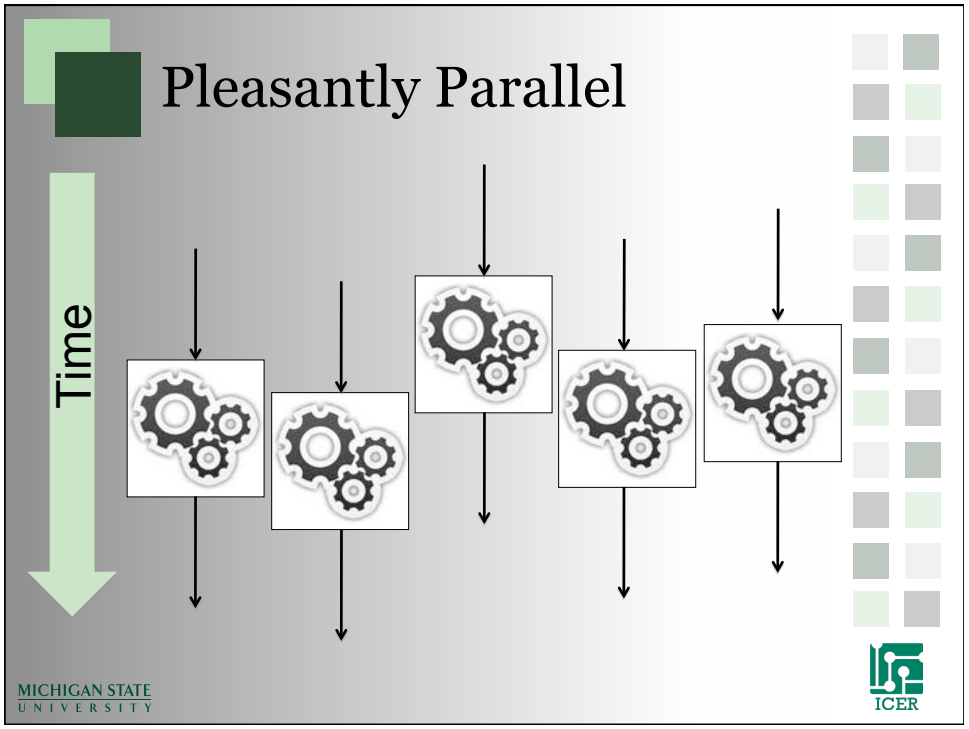
Single Thread Jobs

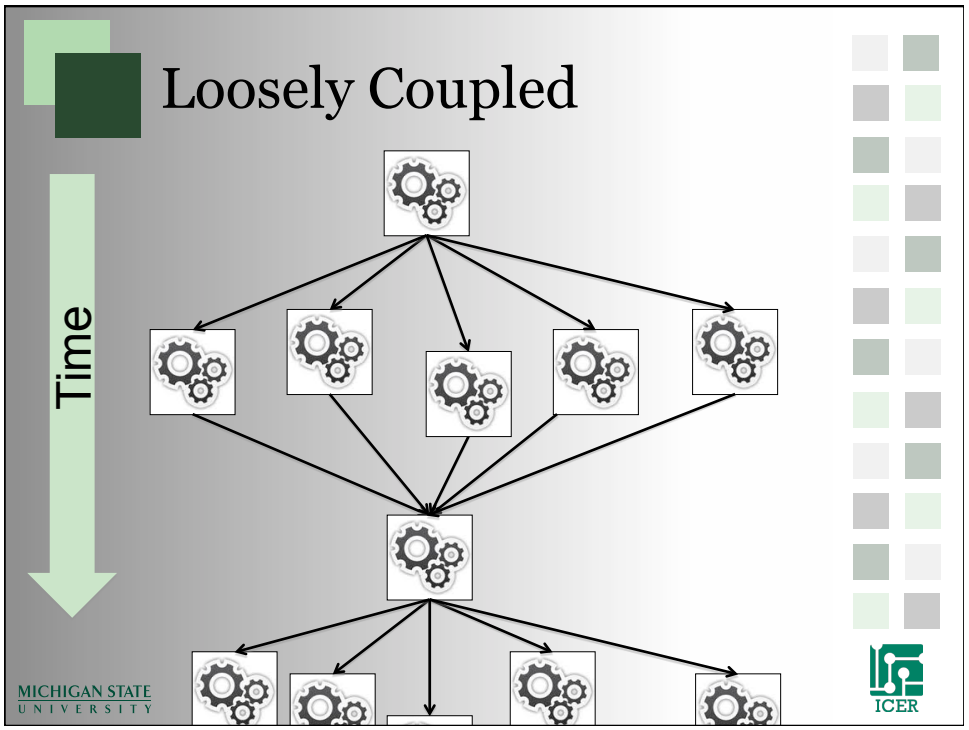


One CPU can only run one thing at a time. (sort of)

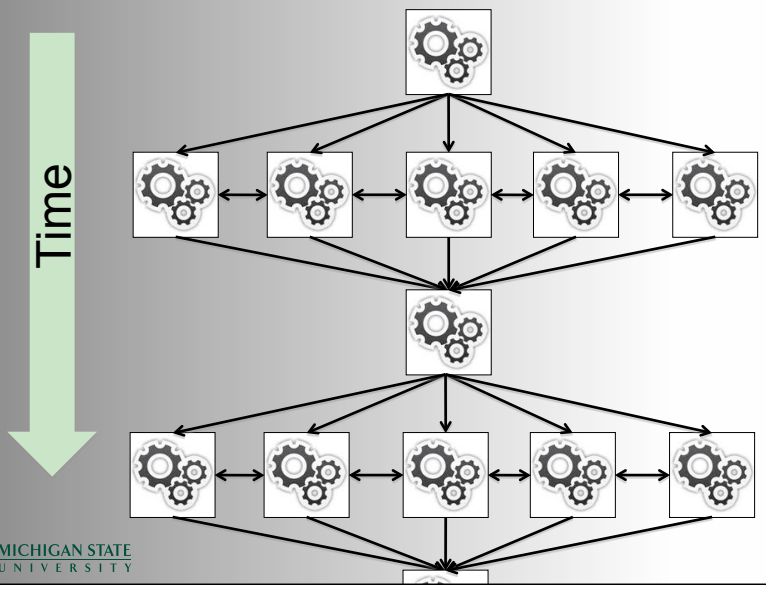
MICHIGAN STATE UNIVERSITY

ICER


Pleasantly Parallel
Time
MICHIGAN STATE UNIVERSITY
ICER


Loosely Coupled
Time
MICHIGAN STATE UNIVERSITY
ICER

Tightly Coupled



MICHIGAN STATE UNIVERSITY

ICER

Example Problems

- Boundary Simulations
- Data Analysis
- Search

MICHIGAN STATE UNIVERSITY

ICER

Example: Boundary simulations

1. Divide a 2D or 3D simulation space into a grid of cells
2. Define information that is transferred at the boundary of the cells
3. Simulate the dynamics of the cell during a time interval
4. Repeat steps 2 and 3

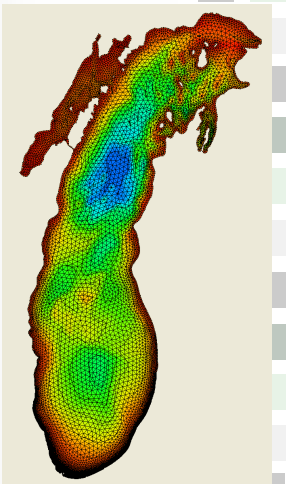


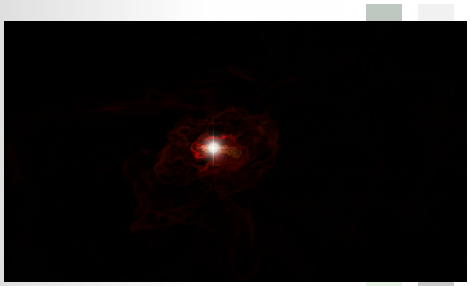
Image Provided by Dr. Mantha Phanikumar, MSU

MICHIGAN STATE UNIVERSITY

ICER

Boundary Simulations

- Fluid dynamics
- Finite element analysis
- Molecular dynamics
- Weather
- Etc.



ENZO Simulation, Drs. O'Shea and Smith

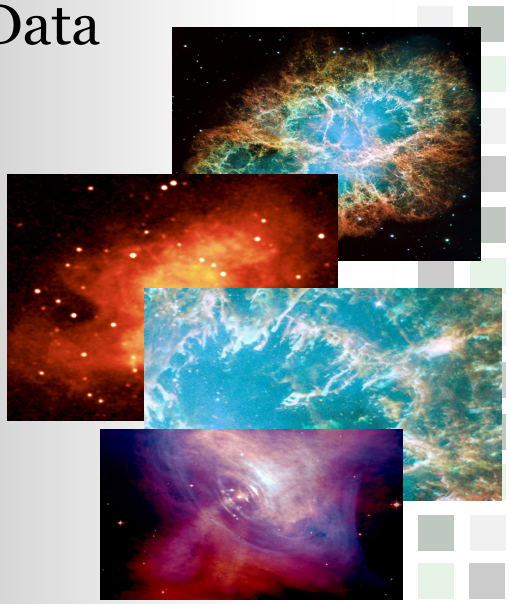
- System of PDE (Partial Differential equations)
- Mathematically equivalent to inverse of a matrix

MICHIGAN STATE UNIVERSITY

ICER

Example: Data Analysis

1. Input data file
2. Find features, search or filter data in some way
3. Output Results



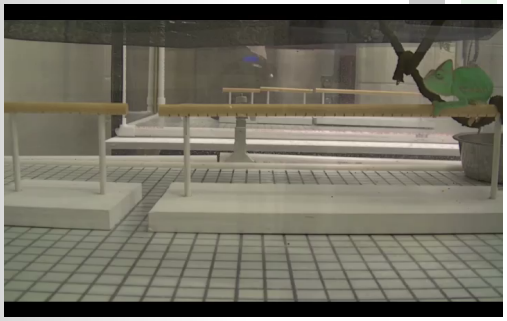
Images from, "Understanding the H_2 Emission from the Crab Nebula", C.T. Richardson, J.A. Baldwin, G.J. Ferland, E.D. Loh, Charles A. Huehn, A.C. Fabian, P.Salomé

MICHIGAN STATE UNIVERSITY

ICER

Data Analysis

- Loosely coupled
- Bulk of computation is typically pleasantly parallel
- Can be I/O bound



Video Provided by Dr. Fred Dyer

MICHIGAN STATE UNIVERSITY

ICER

Example: Search

- Randomly generate test candidates
- Evaluate the quality of solution
- Repeat until found

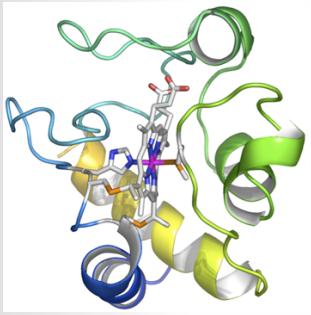


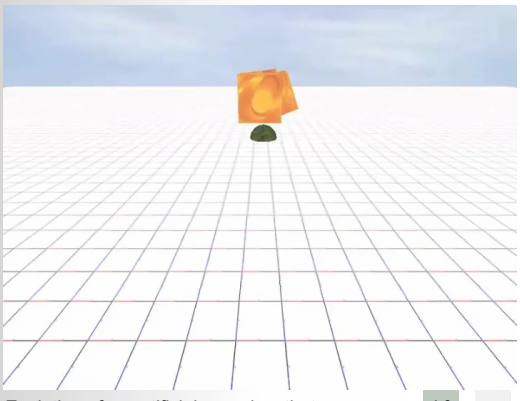
Image Provided by Dr. Warren F. Beck, MSU

MICHIGAN STATE UNIVERSITY

ICER

Search

- Pleasantly parallel
- The more the better
- Typically not I/O bound
- Typically not memory bound



Evolution of an artificial organism that can move and forage for food, Dr. Nicolas Chaumont

MICHIGAN STATE UNIVERSITY

ICER

Agenda

- Intro to me and where I work
- Example Computational Research Problems
- Types of communication

MICHIGAN STATE UNIVERSITY

ICER

Examples

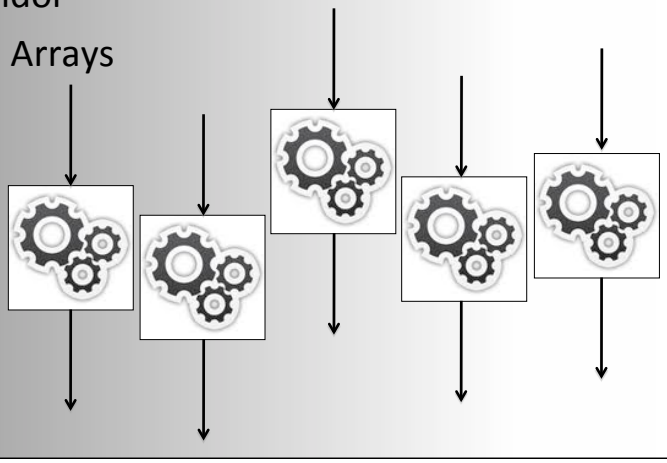
- Examples available on HPCC using the following commands:
  - module load powertools
  - getexample
  - getexample examplename
- Lots of other examples available on the web.

MICHIGAN STATE UNIVERSITY

ICER

Pleasantly Parallel

- Map without the reduce
- Condor
- Job Arrays



MICHIGAN STATE UNIVERSITY

ICER

Condor
High Throughput Computing

- Job submission system
- Runs like a screen saver
- Steals CPU Cycles

MICHIGAN STATE UNIVERSITY

ICER

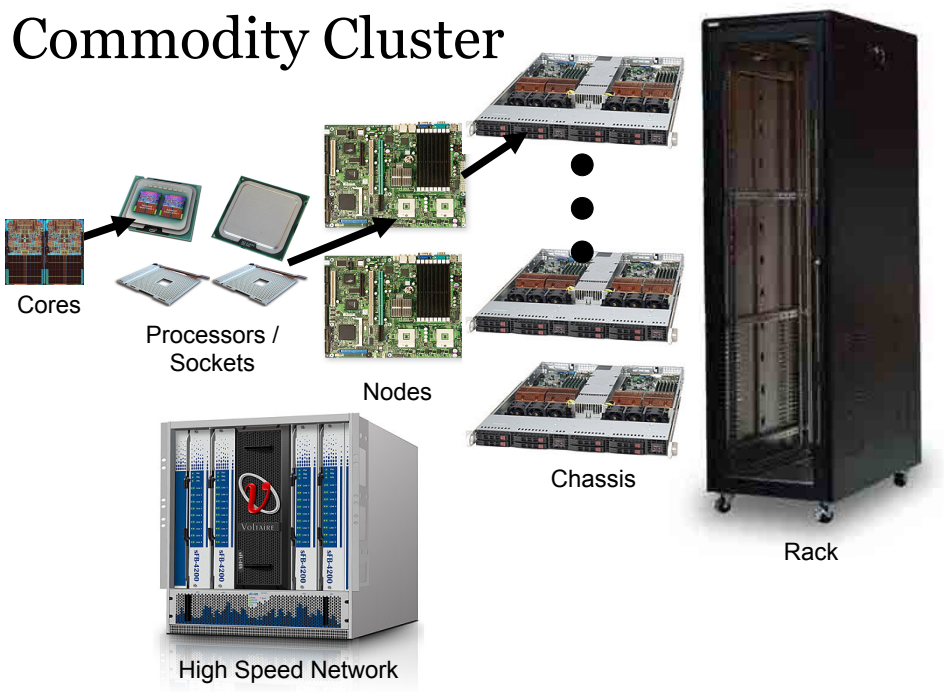
Commodity Cluster

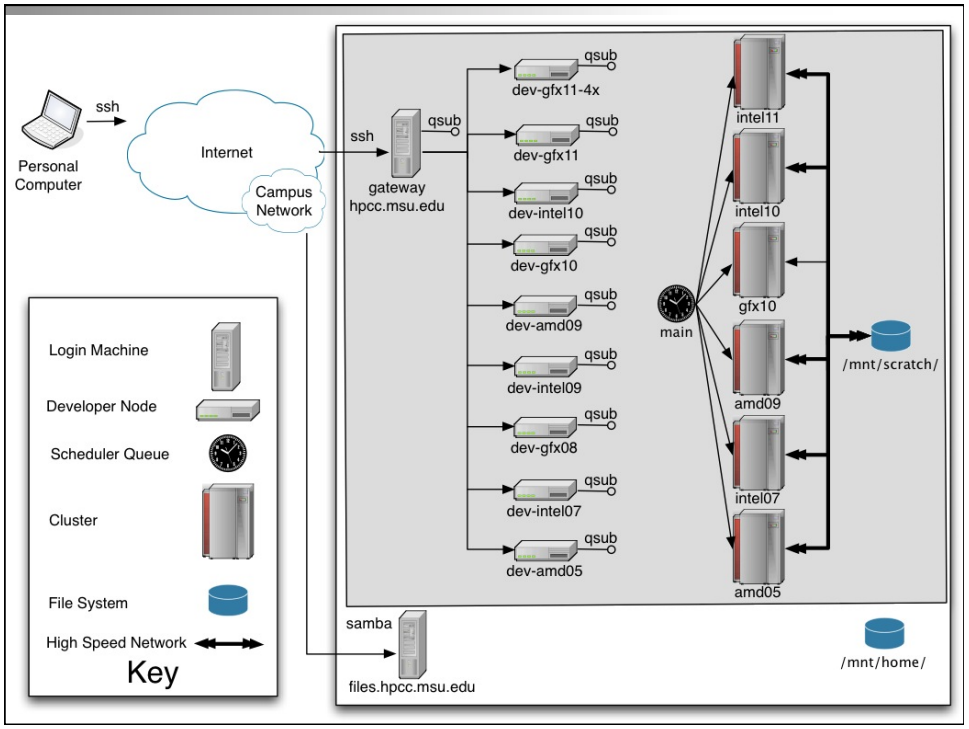
- Most computers at HPCC fall into this category:
  - Racks of commodity Nodes
  - Connected with network
  - Uses a scheduler to run jobs.

MICHIGAN STATE UNIVERSITY

ICER

Commodity Cluster





Simple Job Array

```
#!/bin/bash -login
#PBS -l walltime=00:05:00,mem=2gb
#PBS -l nodes=1:ppn=1,feature=gbe
#PBS -t 1-200

cd ${PBS_O_WORKDIR}

./myprogram ${PBS_ARRAYID}.in > ${PBS_ARRAYID}.out

qstat -f ${PBS_JOBID}
```

MICHIGAN STATE UNIVERSITY

ICER

Communication

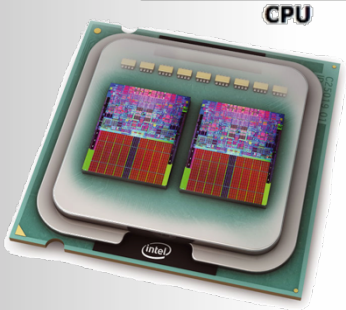
- Shared Memory
- Shared Network
- Distributed Network
- Dedicated Accelerators
- Hybrid Systems

MICHIGAN STATE UNIVERSITY

ICER

Shared Memory Communication

- Cores on a processor share the same memory
- OpenMP
- Fat nodes
  - 64 cores
  - 2TB of memory

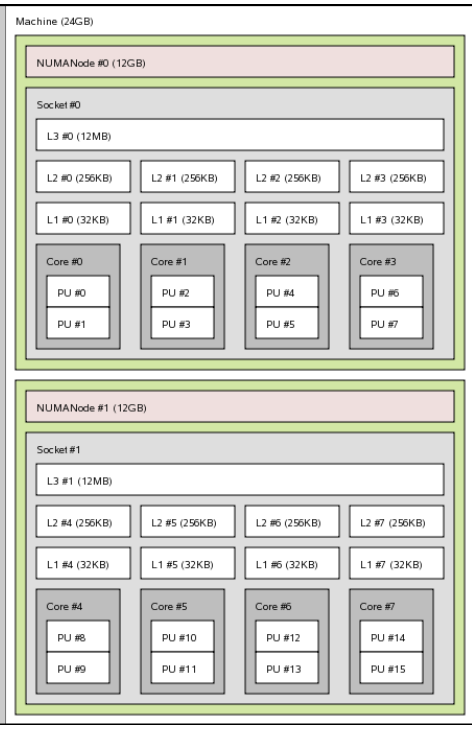


MICHIGAN STATE UNIVERSITY

ICER

Intel10

- 8 cores
- 24 GB memory

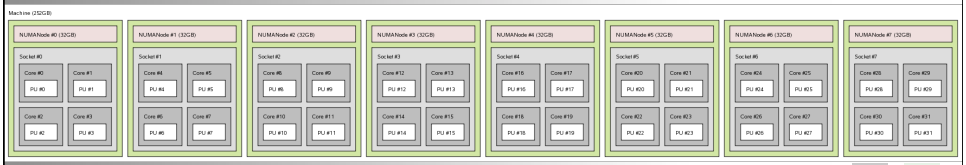


MICHIGAN STATE UNIVERSITY

ICER

amd09

- 32 cores
- 256 GB memory

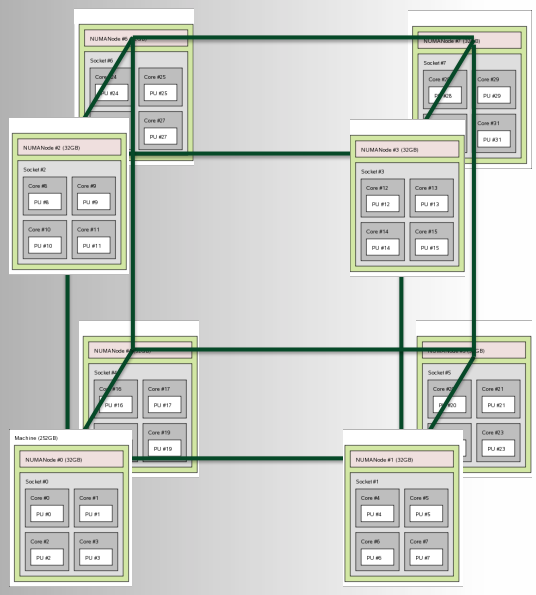


- We have nodes with 64 cores and 2TB of memory

MICHIGAN STATE UNIVERSITY

ICER

NUMA



MICHIGAN STATE UNIVERSITY

ICER

OpenMP

- Common Shared Memory parallelization
- C/C++/FORTRAN
- Single program runs in many cores
- Really easy to pick loops that are parallel and split them into multi threads
- Minor modifications to code that can be written not to affect single cpu version of code.

MICHIGAN STATE UNIVERSITY

ICER

OpenMP is easy

```
#include <omp.h>
...
  #pragma omp parallel for
  for (i=0;i<100;++i) {
    A(I) = A(I) + B
  }
...
```

MICHIGAN STATE UNIVERSITY

ICER

Compile OpenMP Jobs

- Use compiler option openmpi.
  - –fopenmp
- Example:

gcc –fopenmp mycode.cc –o mycode

MICHIGAN STATE UNIVERSITY

ICER

Running OpenMP code

```
export OMP_NUM_THREADS=2

./myCode
```

MICHIGAN STATE UNIVERSITY

ICER

Network parallelization

- Message Passing Interface (MPI)
- C/FORTRAN library that allows programs to pass “messages” between computers over the internet.
- Generally requires a high speed network such as 10gigE or Infiniband

MICHIGAN STATE UNIVERSITY

ICER

MPI program (1 of 4)

```
/* Needed for printf'ing */
#include <stdio.h>
#include <stdlib.h>

/* Get the MPI header file */
#include <mpi.h>

/* Max number of nodes to test */
#define max_nodes 264

/* Largest hostname string hostnames */
#define str_length 50
```


MPI program (2 of 4)

```
int main(int argc, char **argv)
{
   /* Declare variables */
   int   proc, rank, size, namelen;
   int   ids[max_nodes];
   char  hostname[str_length][max_nodes];
   char  p_name[str_length];

   MPI_Status status;
   MPI_Init(&argc, &argv);
   MPI_Comm_rank(MPI_COMM_WORLD, &rank);
   MPI_Comm_size(MPI_COMM_WORLD, &size);
   MPI_Get_processor_name(p_name,&namelen);
```

MPI program (3 of 4)

```
if (rank==0) {
   printf("Hello From: %s I am the receiving processor
%d of %d\n",p_name, rank+1, size);
   for (proc=1;proc<size;proc++) {
      MPI_Recv(&hostname[0][proc], \\
               str_length,MPI_INT,proc, \\
               1,MPI_COMM_WORLD,&status);
      MPI_Recv(&ids[proc], \\
               str_length,MPI_INT,proc, \\
               2,MPI_COMM_WORLD,&status);
      printf("Hello From: %-20s I am processor %d of
%d\n",&hostname[0][proc], ids[proc]+1, size);
   }
```

MICHIGAN STATE UNIVERSITY

ICER

MPI program (4 of 4)

```
} else { // NOT Rank 0
      srand(rank);
      int t = rand()%10+1;
      sleep(t);
      MPI_Send(&p_name,str_length, \\
               MPI_INT,0,1,MPI_COMM_WORLD);
      MPI_Send(&rank,str_length, \\
               MPI_INT,0,2,MPI_COMM_WORLD);
   }
   MPI_Finalize();

   return(0);
}
```

MICHIGAN STATE UNIVERSITY

ICER

Compile MPI Jobs

- To compile an mpi program you need to use the mpi compiler wrappers:
  - mpicc
  - mpif90

MICHIGAN STATE UNIVERSITY

ICER

Using MPI

- MPI programs are run using the "mpirun" command:

  mpirun –np 10 –hostfile ./hosts ./myprogram
  - Number of processor cores
  - Hostefile:
    - Ipaddress
    - Computer names
- In a job script on a cluster using a scheduler:

  mpirun ./myprogram

MICHIGAN STATE UNIVERSITY

ICER

Distributed Network Parallelization

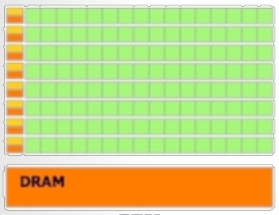
- Map-Reduce (HADOOP)
- Fault tolerant
- Does not require high speed network
- Scales very well.
- Not all problems map well to map-reduce

Dedicated Accelerators

- Small shared memory/network systems.
- HPC on a card
  - GPGPU (CUDA)
  - Phi Cards
  - FPGA

GPUs

- Cards used to render graphics on a computer
- Hundreds of cores
- Not very smart cores
- But, if you can make your research look like graphics rendering you may be able to run really fast!



MICHIGAN STATE UNIVERSITY

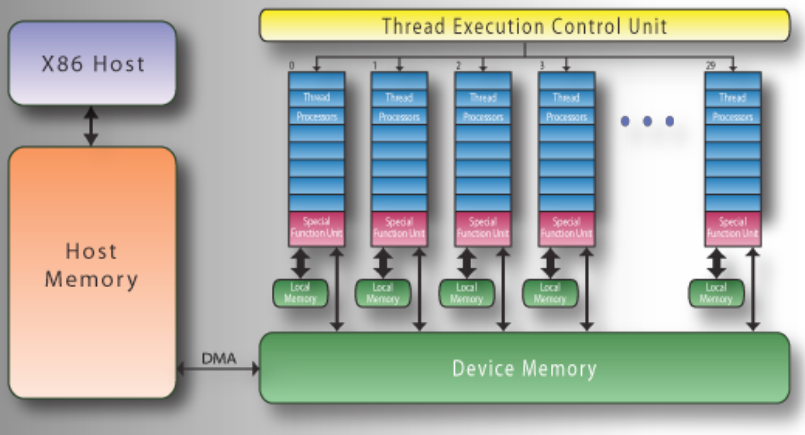
ICER

GPU vs CPU

MICHIGAN STATE UNIVERSITY

ICER

CPU and GPU working together



MICHIGAN STATE UNIVERSITY

ICER

Running on the GPU

- Program Starts on the CPU
  - Copy data to GPU (slow-ish)
  - Run kernel threads on GPU (very fast)
  - Copy results back to CPU (slow-ish)
- There are a lot of clever ways to fully utilize both the GPU and CPU.

MICHIGAN STATE UNIVERSITY

ICER

Pros and Cons

- Benefits
  - Lots of processing cores.
  - Works with the CPU as a co-processor
  - Very fast local memory bandwidth
  - Large online community of developers
- Drawbacks
  - Can be difficult to program.
  - Memory Transfers between GPU and CPU are costly (time).
  - Cores typically run the same code.
  - Errors are not detected (on older cards)
  - Double precision calculations are slow (On older cards)

MICHIGAN STATE UNIVERSITY

ICER

CUDA program (1 of 5)

```
#include "cuda.h"
#include <iostream>

using namespace std;

void printGrid(float an_array[16][16]) {
  for (int i = 0; i < 16; i++){
      for (int j = 0; j < 16; j++) {
          cout << an_array[i][j];
      }
      cout << endl;
    }
}
```

MICHIGAN STATE UNIVERSITY

ICER

CUDA program (2 of 5)

```
__global__ void theKernel(float * our_array)
{
  // This is array flattening,
  //(Array Width * Y Index + X Index)
  our_array[(gridDim.x * blockDim.x) * \\
            (blockIdx.y * blockDim.y + threadIdx.y) + \\
            (blockIdx.x * blockDim.x + threadIdx.x)] = \\
           = 5;
}
```


CUDA program (3 of 5)

```
int main()
{
  float our_array[16][16];

  for (int i = 0; i < 16; i++) {
    for (int j = 0; j < 16; j++) {
      our_array[i][j] = 0;
    }
  }
```

CUDA program (4 of 5)

```
//STEP 1: ALLOCATE
float * our_array_d;
int size = sizeof(float)*256;
cudaMalloc((void **) &our_array_d, size);

//STEP 2: TRANSFER
cudaMemcpy(our_array_d, our_array, size, \\
            cudaMemcpyHostToDevice);
```

MICHIGAN STATE UNIVERSITY

ICER

CUDA program (5 of 5)

```
  //STEP 3: SET UP
  dim3 blockSize(8,8,1);
  dim3 gridSize(2,2,1);

  //STEP 4: RUN
  theKernel<<<gridSize, blockSize>>>(our_array_d);

  //STEP 5: TRANSFER
  printGrid(our_array);
  cudaMemcpy(our_array, our_array_d, size, \\
             cudaMemcpyDeviceToHost);
  cout << "--------------------" << endl;
  printGrid(our_array);


}
```

MICHIGAN STATE UNIVERSITY

ICER

Compile CUDA Jobs

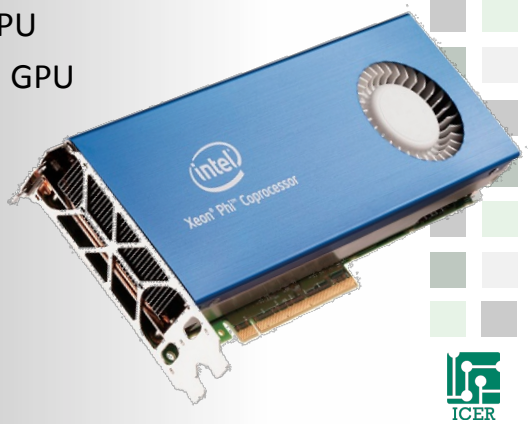
- Just like MPI, to compile an cuda program you need to use the cuda compiler wrappers:
  - nvcc simple.cu -o simple_cuda

MICHIGAN STATE UNIVERSITY

ICER

Intel Xeon Phi

- Cross between CPU and GPU
- About 60 Pentium I cores
  - Less cores than GPU
  - Easier to use than GPU
    - OpenMP
    - MPI
- Very new
  - January 2013



MICHIGAN STATE UNIVERSITY

ICER

Which approach is the best?

- Depends on what you are doing?
- Depends on how much communication you need.
- Depends on what hardware you have.
- Depends on how much time you have.

Terms Test

- Condor
- Job Array
- OpenMP
- MPI
- Hadoop
- Cuda
- GPU
- Phi
- Communication
- Scaling
- Accelerator Cards


QUESTIONS?
MICHIGAN STATE
UNIVERSITY
ICER