


Title goes here

CS 594 Spring 2006

594 Class 5: Cluster Computing Part 1
G E Fagg

What Computers Do
& Architectures



INNOVATIVE COMPUTING LABORATORY
COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF TENNESSEE

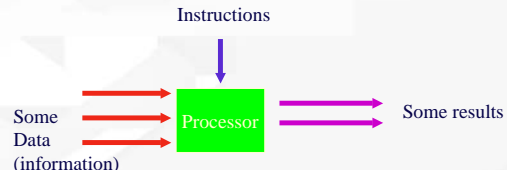
What do computers do?

- » What do computers do?
 - » They do calculations
 - » Floating point
 - » Matrix multiplication
 - » Used to solve PDE (real world modeling)
 - » Risk analysis (online mortgage) broking
 - » Failure modeling of cars (reduces expensive testing)
 - » Risk analysis (statistics)
 - » Fixed point arithmetic (or variable length)
 - » Verifying YOUR ID as you take money out of THEIR ATM
 - » Integer values
 - » Calculating an index into a database table
 - » Yes your car tax is current
 - » Calculating offset into disk cache for faster web page serving
 - » Makes more transactions -> sales -> revenue -> PROFIT
 - » Matching STRINGS
 - » Calculating if you are genetically predisposed to a risk
 - » Reduce medical insurance risks

Introduction

- » What do computers do?
- » How do they do it?
- » Architectures
- » Different work loads

How do they do it?

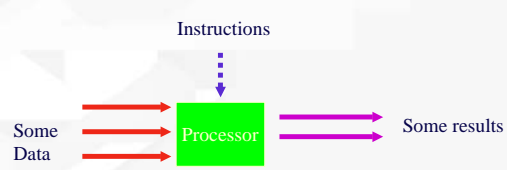


```
graph TD; I[Instructions] --> P[Processor]; D[Some Data (information)] --> P; P --> R[Some results]
```

What do computers do?

- » What do computers do?
 - » They do calculations
 - » Floating Point
 - » Matrix multiplication
 - » Used to solve equations (like in CS/Eng homework)
 - » Fixed point or variable length arithmetic
 - » Taking YOUR money out of the ATM
 - » Rendering graphics on your PS2
 - » Integer Operations
 - » Decrementing integer values and checking for a zero flag
 - » Your lunch is ready in the microwave

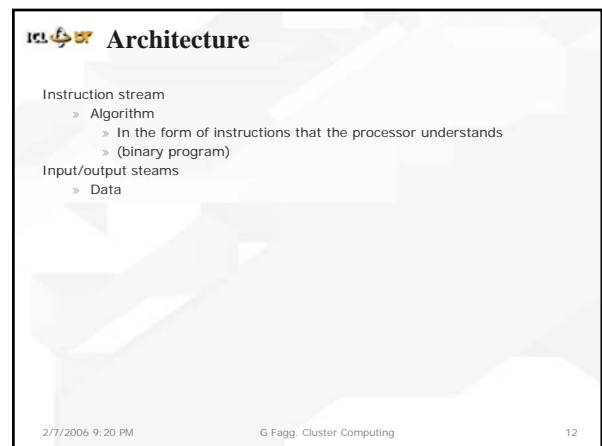
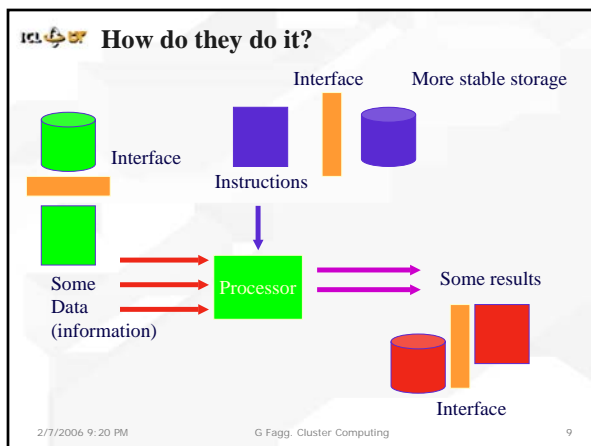
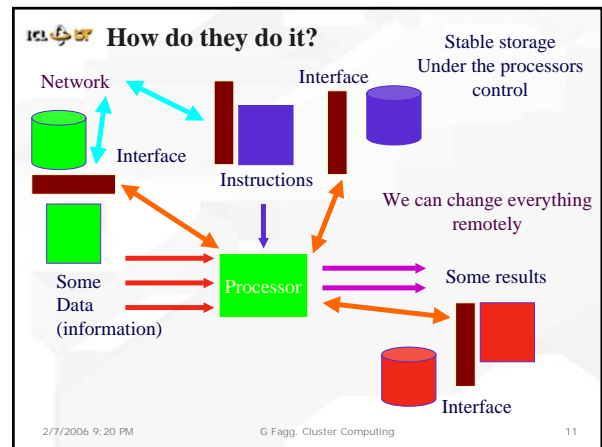
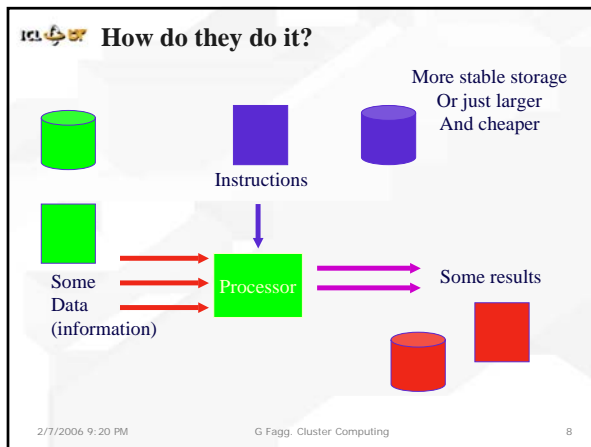
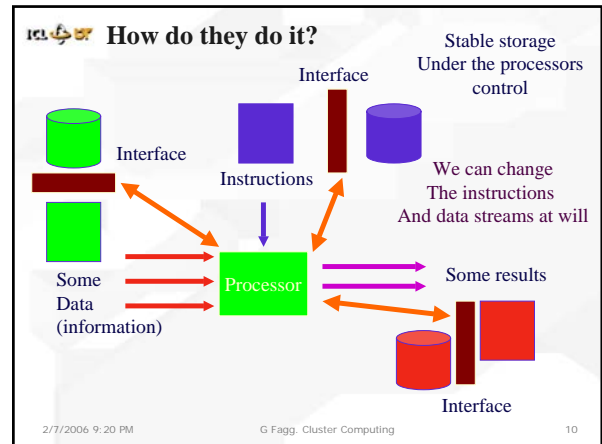
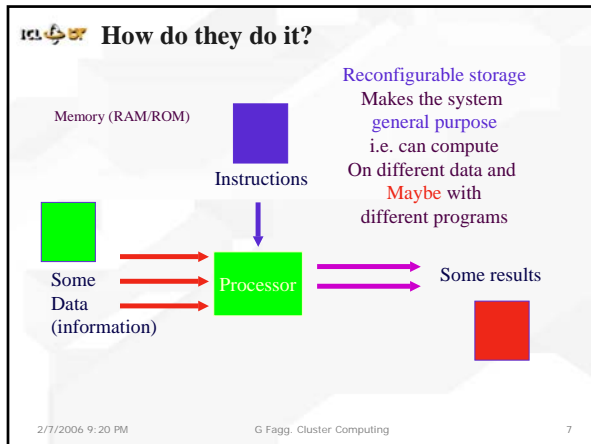
How do they do it?



```
graph TD; I[Instructions] --> P[Processor]; D[Some Data (information)] --> P; P --> R[Some results]
```

Custom hardware: can be cost effective (embedded systems, low power etc)
But can also be more expensive

Title goes here



Title goes here

Architecture

Instruction stream

- » Algorithm
 - » In the form of instructions that the processor understands
 - » (binary program)

Input/output streams

- » Data

» To make a useful system we need (a subset of) the following:

- » Processor (always)
- » External Volatile Memory (almost always (for modern systems))
- » External stable storage (usually)
- » External networking (usually)
- » (+ other misc items, PSU, cooling, a box to put it all in...)

2/7/2006 9:20 PM G Fagg, Cluster Computing 13

Architecture

If we just consider instruction/data streams

Instructions

Data ↔ Processor

2/7/2006 9:20 PM G Fagg, Cluster Computing 16

Architecture

Instruction stream

- » Algorithm
 - » In the form of instructions that the processor understands
 - » (binary program)

Input/output streams

- » Data

» To make a useful system we need (a subset of) the following:

- » Processor (always)
- » External Volatile Memory (almost always (for modern systems))
- » External stable storage (usually)
- » External networking (usually)
- » (+ other misc items, PSU, cooling, a box to put it all in...)

» Something to handle all the interfaces between the processor and external memory, storage and networking

- » Operating system (may be very small)
- » Something to create the binary programs with? (and debug them)

2/7/2006 9:20 PM G Fagg, Cluster Computing 14

Architecture

Single Instructions

Single Data ↔ Processor

SISD
Classic single processor

2/7/2006 9:20 PM G Fagg, Cluster Computing 17

Architecture

If we just consider instruction/data streams

Flynn 1966

Instructions

Data → Processor → Data

2/7/2006 9:20 PM G Fagg, Cluster Computing 15

Architecture

Single Instructions

Multiple Data ↔ Processor

Processor

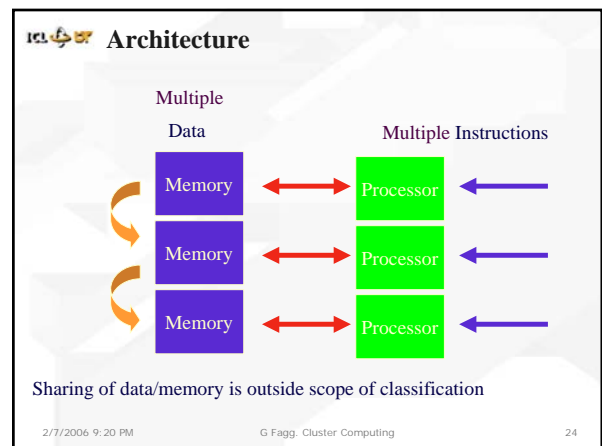
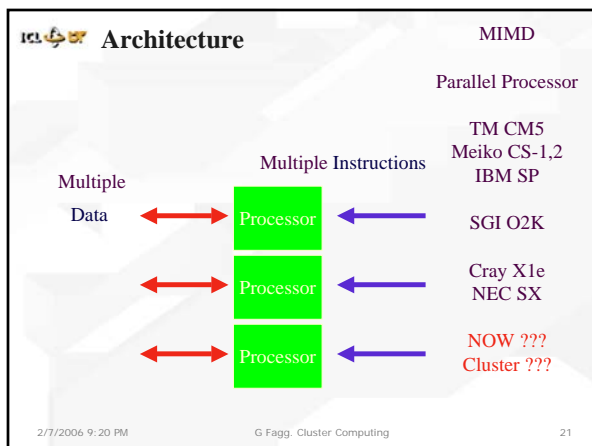
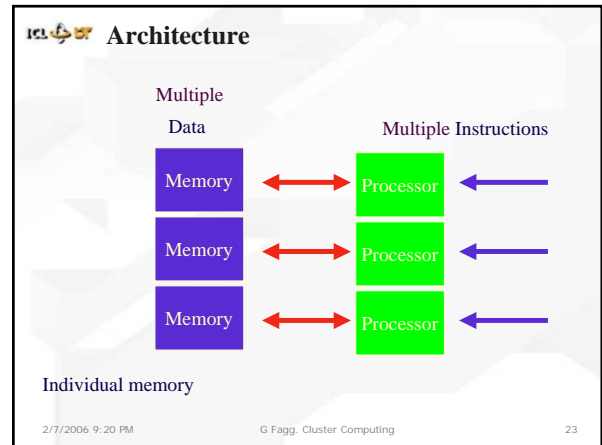
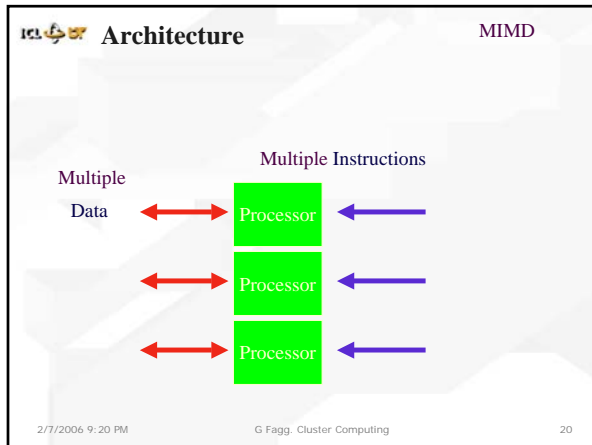
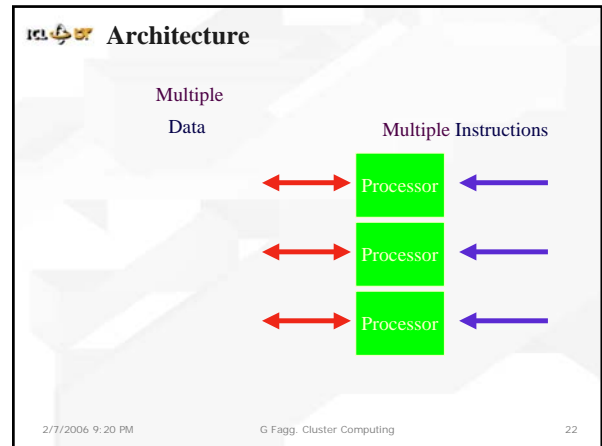
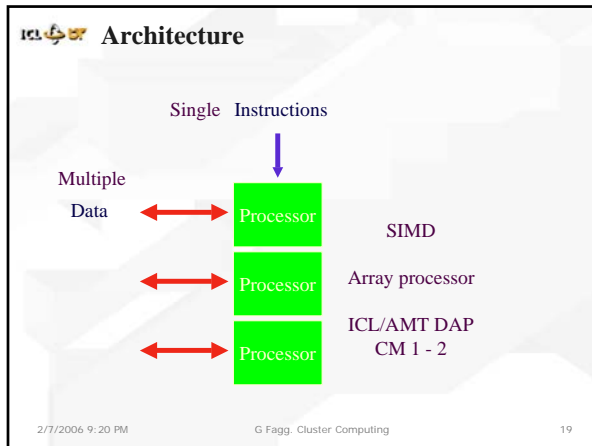
Processor

Processor

SIMD

2/7/2006 9:20 PM G Fagg, Cluster Computing 18

Title goes here



Title goes here

Architecture

Multiple Data

Multiple Instructions

Networking

Memory

Processor

Sharing of data/memory is outside scope of classification

2/7/2006 9:20 PM G Fagg, Cluster Computing 25

Detailed description: This diagram illustrates Flynn's Architecture. On the left, three purple boxes labeled 'Memory' are stacked vertically under the heading 'Multiple Data'. On the right, three green boxes labeled 'Processor' are stacked vertically under the heading 'Multiple Instructions'. Red double-headed arrows connect each memory box to its corresponding processor box. Blue arrows point from the right towards each processor box, representing instruction streams. On the far left, two orange curved arrows labeled 'Networking' point towards the memory stack. Below the diagram, the text states 'Sharing of data/memory is outside scope of classification'. The footer includes the date '2/7/2006 9:20 PM', the author 'G Fagg, Cluster Computing', and the slide number '25'.

Architecture

- > Flynn's Classification indicates how many instruction and data streams there are and how they relate to each other
 - > SISD
 - > SIMD
 - > MIMD
- > DOES NOT indicate where they come from, or if data or instructions are shared in storage
 - > Distributed Memory
 - > Shared Memory
 - > Vector
- > Flynn's classification controls the programming method to some extent

2/7/2006 9:20 PM G Fagg, Cluster Computing 28

Detailed description: This slide contains a bulleted list under the heading 'Architecture'. The first bullet point states that Flynn's Classification indicates the number and relationship of instruction and data streams, with sub-points for SISD, SIMD, and MIMD. The second bullet point states that it does not indicate where data or instructions come from or if they are shared, with sub-points for Distributed Memory, Shared Memory, and Vector. The third bullet point states that Flynn's classification controls the programming method to some extent. The footer includes the date '2/7/2006 9:20 PM', the author 'G Fagg, Cluster Computing', and the slide number '28'.

Architecture

Multiple Data

Multiple Instructions

Memory

Processor

Shared memory as an architecture class / as well as programming model

2/7/2006 9:20 PM G Fagg, Cluster Computing 26

Detailed description: This diagram illustrates Shared Memory Architecture. On the left, a single large purple box labeled 'Memory' is under the heading 'Multiple Data'. On the right, three green boxes labeled 'Processor' are stacked vertically under the heading 'Multiple Instructions'. Red double-headed arrows connect the single memory box to each of the three processor boxes. Blue arrows point from the right towards each processor box, representing instruction streams. Below the diagram, the text states 'Shared memory as an architecture class / as well as programming model'. The footer includes the date '2/7/2006 9:20 PM', the author 'G Fagg, Cluster Computing', and the slide number '26'.

Architecture

- > Processors
 - > Moore's law (1975) circuit complexity doubles every 18 month.
 - > Doubling the transistor count does not always lead to a doubling of FLOPs.
 - > Neither does doubling the clock speed.
 - > But it does allow for implicit parallelism within the processors that increases the number of Ops per cycle.

2/7/2006 9:20 PM G Fagg, Cluster Computing 29

Detailed description: This slide contains a bulleted list under the heading 'Architecture'. The main bullet point is 'Processors', which has four sub-points: Moore's law (1975) circuit complexity doubles every 18 month, Doubling the transistor count does not always lead to a doubling of FLOPs, Neither does doubling the clock speed, and But it does allow for implicit parallelism within the processors that increases the number of Ops per cycle. The footer includes the date '2/7/2006 9:20 PM', the author 'G Fagg, Cluster Computing', and the slide number '29'.

Architecture

Multiple Data

Multiple Instructions

Memory

Processor

These links are usually Very Fast

2/7/2006 9:20 PM G Fagg, Cluster Computing 27

Detailed description: This diagram illustrates Shared Memory Architecture with a focus on the data links. On the left, a single large purple box labeled 'Memory' is under the heading 'Multiple Data'. On the right, three green boxes labeled 'Processor' are stacked vertically under the heading 'Multiple Instructions'. Red double-headed arrows connect the single memory box to each of the three processor boxes. Blue arrows point from the right towards each processor box, representing instruction streams. A red curved arrow points from the text 'These links are usually Very Fast' to the data links between the memory and processors. The footer includes the date '2/7/2006 9:20 PM', the author 'G Fagg, Cluster Computing', and the slide number '27'.

Architecture

- > Memory
 - > Densities and speed have not always kept up with CPUs.

CPU speed

Memory access speed

2/7/2006 9:20 PM G Fagg, Cluster Computing 30

Detailed description: This slide contains a bulleted list under the heading 'Architecture'. The main bullet point is 'Memory', with a sub-point: Densities and speed have not always kept up with CPUs. Below the text is a graph with two lines. The red line, labeled 'CPU speed', shows an exponential increase. The blue line, labeled 'Memory access speed', shows a much slower, nearly linear increase. The footer includes the date '2/7/2006 9:20 PM', the author 'G Fagg, Cluster Computing', and the slide number '30'.

Title goes here

Architecture

- » Memory
 - » CPU speeds have increased at approx 40%/year
 - » Memory (DRAM) speeds 10%/year
 - » Created a deep memory hierarchy

2/7/2006 9:20 PM G Fagg, Cluster Computing 31

Architecture

10 GB/Sec
On-die
Bandwidth
100MB/Sec

CPU
registers
Level 1
Level 2
Main memory
SS / Disk

Fast nSec
16-128+KBytes
128KBytes – 2MBytes
60-100nSec
128KBytes-2GB
Slow mSec
2GBytes – 2TB+

2/7/2006 9:20 PM G Fagg, Cluster Computing 34

Architecture

On-die
CPU
registers
Level 1
Level 2
Main memory
SS / Disk

Fast nSec
60-100nSec
Slow mSec

2/7/2006 9:20 PM G Fagg, Cluster Computing 32

Architecture

- » Modern Single CPU systems need small 'chunks' to get better utilization / efficiency
 - » Smaller problems mean better infinity of data to the caches
 - » In-core verses out-of-core
 - » Leave it to the compilers to handle large data
- » Case is much better for specialized / custom CPUs
 - » Except they cost a lot.
 - » Price / performance compromise to be made

2/7/2006 9:20 PM G Fagg, Cluster Computing 35

Architecture

10 GB/Sec
On-die
Bandwidth
100MB/Sec

CPU
registers
Level 1
Level 2
Main memory
SS / Disk

Fast nSec
60-100nSec
Slow mSec

2/7/2006 9:20 PM G Fagg, Cluster Computing 33

Architecture

TOP 500
Chip Technologies

ECL
CMOS off the shelf
CMOS on-chip

2/7/2006 9:20 PM G Fagg, Cluster Computing 36

Title goes here

Architecture

- » Single processor systems do not have the memory bandwidth to sustain modern CPUs.
- » But using multiple CPUs and expanding the memory paths [may] solve this problem.
 - » It can also make it worse.
 - » Look at a Cray T3e versus a modern IBM SP
 - » Or a NEC SX-6 vs Cray X-1
- » Cost of using lots of commodity processors is now much cheaper than building a custom CPU.

2/7/2006 9:20 PM G Fagg, Cluster Computing 37

Work loads

- » Going back to what organizations use computers for:
 - » High Throughput
 - » High Availability
 - » High Performance

2/7/2006 9:20 PM G Fagg, Cluster Computing 40

Architecture

2/7/2006 9:20 PM G Fagg, Cluster Computing 38

Work loads

- » High Throughput
- » Handling as many operations as possible per unit of time
- » Computational jobs
 - » Transactions
 - » Banking operations
 - » Database updates
 - » Web pages served
 - » They are usually independent operations (but can have dependencies and be nested)

2/7/2006 9:20 PM G Fagg, Cluster Computing 41

Architecture


2/7/2006 9:20 PM G Fagg, Cluster Computing 39

Work loads

- » High Throughput
- » They are usually independent operations (but can have dependencies and be nested) -> sequential codes?
- » Shared memory, Distributed systems and maybe parallel systems


2/7/2006 9:20 PM G Fagg, Cluster Computing 42

Title goes here

 **Work loads**


- » High Availability
- » Almost always the same as High Throughput BUT
- » Handling some minimum level of operations per unit of time
(and the rest of the time, maybe high throughput)
- » Computational jobs
 - » Weather forecasting, wave height analysis
- » Transactions
 - » Banking operations
 - » Database updates
 - » Web pages served
- » They are usually independent operations (but can have dependencies and be nested)

2/7/2006 9:20 PM G Fagg, Cluster Computing 43

 **Work loads**


- » High Performance
- » Parallel Programming
 - » Depending on the data interactions might be shared memory or distributed memory (with interconnect)

2/7/2006 9:20 PM G Fagg, Cluster Computing 46

 **Work loads**

- » High Availability
- » They are usually independent operations (but can have dependencies and be nested) -> sequential codes?
- » Shared memory, Distributed systems and maybe parallel systems

2/7/2006 9:20 PM G Fagg, Cluster Computing 44

 **Work loads**

- » High Performance
- » Solving problems that are bigger, in a shorter time
- » $O(n)^3$ cpu operations verses $O(n)^2$ memory
- » Scientists want to solve bigger and bigger problems
(More accurate models of the real world)
- » CPUs limited by speed of light, internal register widths and data paths
 - » More performance = more processors
 - » In cache performance verse out of cache performance
- » Due to limits in system components might use multiple systems just to
 - » Have enough memory
 - » Addressable memory limit vs memory needed

2/7/2006 9:20 PM G Fagg, Cluster Computing 45