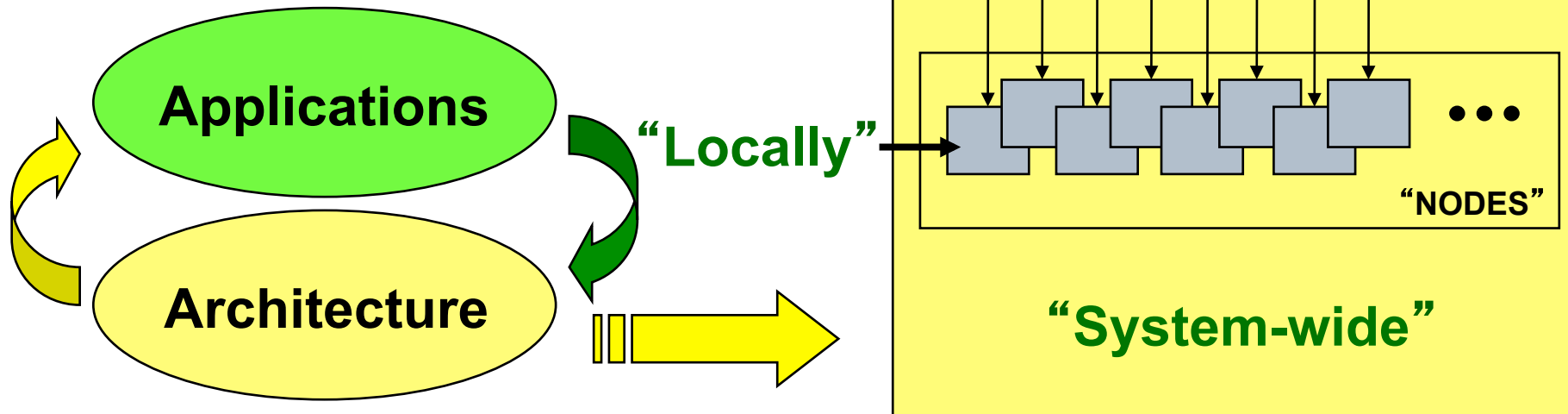


Module 2: Parallel Comp. Setup

- Parallel computing challenge
- Our course focus
- Ways to compute in parallel
- Sources of parallelism
- Technology underpinning
 - Area vs. delay
 - Locality challenge, memory wall

Application/Architecture Challenge

- Performance beyond a single (commodity) processor is only possible as a result of concurrency (parallelism) in applications
- Hierarchical application characteristics
 - “In the small” i.e., “Locally”
 - “In the large” i.e., “System-wide”



This Course: Focus

- Realistic parallel computing exposure
 - Shared-memory multiprocessing, symmetric (SMP)
 - Dual and multiple core general purpose processors
 - Distributed memory machines
 - Message-passing paradigm
 - Some research exposure of lecturer:
 - Non-Uniform Shared Memory (NUMA)
 - Distributed applications: smartphones, cloud, devices
- Programming
 - Concurrent – still on one processor
 - Parallel – using explicitly multiple processors
 - Distributed – using multiple computers

Learning Objectives

- Critical understanding of parallel and distributed systems
 - Performance measures
 - Difficulties and tradeoffs
 - Trends
- Concurrency issues in SW
- Start-to-end parallel/distributed computing project

Quiz: Opposite to Parallel?

- a.) Perpendicular?
- b.) Meandering?
- c.) Serial?
- Answer: think “Turing Machine” (T.M.)
 - T.M. : universal processor/computer model
 - Scans infinite tape with symbols (program or data)
 - On every new symbol, moves and produces output by rules defined via a finite state machine (**sequential**)
- T.M. is as serial as it can be, but is usually referred to as “sequential”

What is Parallel (in Computing)?

- Circuit
 - Analog, digital, quantum; combinational, sequential, ...
- Single processor + specialized circuit (possibly reconfigurable)
 - FPGA computing machine
- Single pipelined processor, single superscalar processor, single multithreaded processor
- Single SIMD processor
 - Also: vector processor/machine
- Multiple processors executing single program
 - Shared- or distributed-memory, ...,
- Multiple computers executing single program
 - Distributed or distributed-shared memory

Parallelism Available

- Bits
- Operations
 - Add, subtract, multiply, ...
 - Instruction-level (ILP)
 - How many processor instructions in parallel?
- Thread-level
 - How many threads at once
- Process-level – as above, less used
- Task-level
- Coarse-level: complete programs (or so)