



**CS 8625**

# System Simulation with Threads

CS 8625 Dr. Ken Hoganson, Copyright © 2009

## A simple model of a computer system.

### Contains thread types:

- Main thread – has clock and shared memory
- (8) Processor threads – each have cache
- The threads communicate through shared memory, by accessing the shared array in the main thread.
- The processor has a cache that can hold 8 pages (each “page” will be contain a single data item – an integer, initially empty)
- The memory stores 32 integers. The initial contents of memory will be the same as its “address”.

- The processor thread will generate a request for data (an “address” – a random number between 0 and 31)
- The processor will check the cache for the requested data (will be a simple linear search through the storage locations in the cache to see if the needed page is in the cache)
- If the cache has the data, then the processor will record a tally of cache hits.
- 50% of addresses are for data to read, 50% are for data to read/increment/write-back (writes are increments + 1).

- If the cache does not have the data, then the processor will record a tally of cache misses, and will then go to the shared memory.
- The processor will print the data (an integer) on the screen
- The processor will iterate a fixed number of times (10 to begin with), then terminate and terminate the program. (iterate 1000 times when all is working).

- The cache is private memory to the processor thread.
- It is fully associative (data can go anywhere)
- 8 storage locations, each with an address
- Address is the memory page that the cache duplicates
  
- What about cache replacement algorithm? – just use a random number from 1-8 to chose a cache page to replace with a new page on page faults.

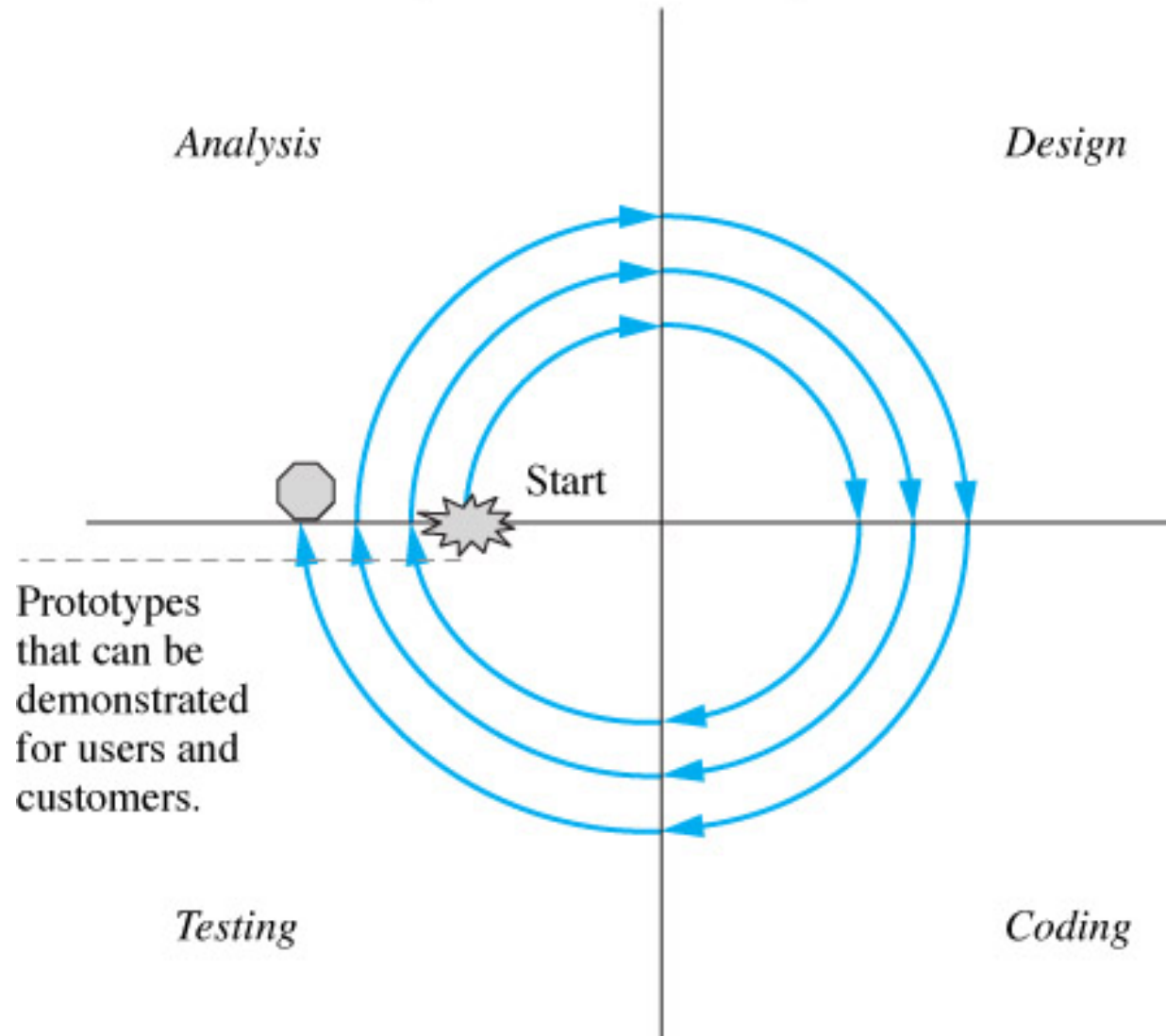
- **Data Assigned: Tuesday, June 16**
- **Turn in by email by Midnight on TBD:**
  - documented java code

- Build and test in phases
- More fun, builds confidence, less frustrating
- Small phases are easier to debug
- Often called the Spiral method or spiral design method

- A simple way to design software.
- Particularly effective for students new to programming, OR
- When learning a new programming language
- Also, when building an application in
  - new area,
  - or with a new technology,
  - or new techniques.
- Build and test in parts, that slowly evolve
- Build a prototype, and then evolve and grow the prototype.

- Build a prototype, and then evolve and grow the prototype.
- Prototype Demonstrations: and excellent way to communicate with
  - clients,
  - customers,
  - focus groups,
  - managers.
- Prototype demonstrations can **validate** the:
  - objectives,
  - goals,
  - and user interface

## Spiral or Evolutionary Model



**End**

**Of**

**Today's**

**Lecture.**



This slide intentionally left blank