The UPC Memory Model

- An introduction
Memory consistency models

- A Language / Programmer contract
- What do we mean by consistency?
- Correctness / cost
- Example:

```
Thread 0

Shared_x = 1;

if (shared_y==0) printf("I win!");

Winner!
```

```
Thread 1

Shared_y = 1;

if (shared_x==0) printf("I win!");

No winner!
```
Specification

• Shared memory behavior
  – Data transformation
  – Control Flow
  – Local Accesses

• Operations
  – Reads (RR, SR)
  – Writes (RW, SW)

• Notation
  – Op(var,value)
  – Example: SW(x,1) and RR(y,0)

• Assumptions
  – All variables are initialized to 0
UPC memory consistency model, v1.0

• Two memory modes
  – Strict (high level of consistency)
  – Relaxed (greater opportunity for optimization)

• Synchronization operations
  – Upc_barrier
  – Upc_notify, upc_wait ("split-phase" barrier)
  – (Upc_fence)
“Strict” memory model

- No re-ordering on thread
- All threads observe all strict ops in same order
- Precise serialization of all memory accesses

Example 1:

- Thread 0
  - SW(x,1)
  - SW(y,1)
- Thread 1
  - SR(x,1)
  - SR(y,1)
“Strict” memory model

- No re-ordering on thread
- All threads observe all strict ops in same order
- Precise serialization of all memory accesses
- Example2:

Thread 0

Thread 1

SW(y, 1) → SR(x, 0)

SW(x, 1) → SR(y, 1)
“Relaxed” memory model

- Operations can usually be re-ordered
- Every thread has their own legal ordering
- Example 1:

```
Thread 0
RR(x,1) → RW(x,2)
RR(x,2) ← RW(x,1)
```

```
Thread 1
RR(x,2) ← RW(x,1)
RR(x,1) → RW(x,2)
```
“Relaxed” memory model

- Operations can usually be re-ordered
- Every thread has their own legal ordering
- Example 2:

 Thread 0

\[\text{RW}(x,1) \rightarrow \text{RW}(y,1)\]

Thread 1

\[\text{RR}(y,1) \rightarrow \text{RR}(x,0)\]
“Strict” and “Relaxed”

• All strict operations are observed in the same order by all threads

• Other operations can be ordered differently on different threads
Other examples
Other examples
Other examples

Thread 0
- SW(x,2) → SR(x,1)

Thread 1
- SW(x,1) → SR(x,2)

Thread 0
- SW(x,2) → SR(x,1)

Thread 1
- SW(x,1) → SR(x,2)
Other examples

Thread 0
- RW(x,1)
- SW(y,1)

Thread 1
- RR(y,1)
- RR(x,1)
- RR(x,0)

Thread 0
- RR(x,0)

Thread 1
- RW(x,1)
- SW(y,1)

- RR(x,1)
- RR(y,1)
Special operations

- \texttt{Upc\_notify} = SW*
- \texttt{Upc\_wait} = SR*
- \texttt{Upc\_barrier} (and \texttt{upc\_fence}) = SW*, SR*
Example

Thread 0

| RW(x,1) | Upc_notify | RR(x,0) |

Thread 1

| RW(x,1) | Upc_notify | RR(x,0) |