Memory Consistency Models

Anupama Potluri

School of Computer and Information Sciences
University of Hyderabad

Outline

Sequential Consistency I

Definition

The result of any execution is the same as if the read and write operations by all processes are executed in some sequential order and the operations of each individual process appear in this sequence in the order specified by its program.

Sequential Consistency II

Table 1 : Example of Correct Sequential Consistency

P1:	W(x)a				
P2:		W(x)b			
P3:			R(x)b		R(x)a
P4:				R(x)b	R(x)a

Table 2 : Example of Incorrect Sequential Consistency

P1:	W(x)a				
P2:		W(x)b			
P3:			R(x)b		R(x)a
P4:				R(x)a	R(x)b

Sequential Consistency III

Table 3 : An Example of Three Concurrently Executing Processes

Process P1	Process P2	Process P3
x = 1;	y = 1;	z = 1;
print(y, z);	print(x, z);	print(x, y);

Table 4 : Some valid execution sequences with the vertical axis representing time

x= 1;	x = 1;	y = 1;	y = 1;
print(y, z);	y = 1;	z = 1;	x = 1;
y = 1;	print(x, z);	print(x, y);	z = 1;
print(x, z);	print(y, z);	print(x, z);	print(y, z);
z = 1;	z = 1;	x = 1;	print(y, z);
print(x, y);	print(x, y);	print(y, z);	print(x, y);
Prints: 001011	Prints: 101011	Prints: 010111	Prints: 111111

Invalid String: 001001

FIFO Consistency or Processor Consistency or PRAM Consistency I

Definition

Writes done by a single process are seen by all other processes in the order in which they were issued, but writes from different processes may be seen in a different order by different processes.

Table 5 : Example of Correct Processor Consistency

P1:	W(x)a						
P2:		R(x)a	W(x)b	W(x)c			
P3:					R(x)b	R(x)a	R(x)c
P4:					R(x)a	R(x)b	R(x)c

FIFO Consistency or Processor Consistency or PRAM Consistency II

Table 6 : Execution sequences seen by Processes P1, P2 and P3

Process P1	Process P2	Process P3
x= 1;	x = 1;	y = 1;
print(y, z);	y = 1;	print(x, z);
y = 1;	**print(x, z);**	z = 1;
print(x, z);	print(y, z);	**print(x, y);**
z = 1;	z = 1;	x = 1;
print(x, y);	print(x, y);	print(y, z);
Prints: 00	Prints: 10	Prints: 01

Weak Consistency I

When a process is in a critical section and modifying shared data, there is no need to propagate every access to all other processes. Instead, the final outcome after the critical section is exited can be written **once** to all other processes, thus reducing the traffic and latency of operations.

Weak Consistency II

Properties

1. Accesses to syncrhonization variables associated with a data item are sequentially consistent.
2. No operation on a synchronization variable is allowed to be performed until all previous writes have completed everywhere.
3. No read or write operation is allowed to be performed until all earlier operations on synchronization variables are performed.

Weak Consistency III

Table 7 : A valid sequence of events for Weak Consistency

P1:	W(x)a	W(x)b	S			
P2:				R(x)a	R(x)b	S
P3:				R(x)b	R(x)a	S

Table 8 : An invalid sequence of events for Weak Consistency

P1:	W(x)a	W(x)b	S		
P2:				S	R(x)a

Release Consistency I

In weak consistency, there is only one variable for synchronization whether a process is entering or exiting from a critical section. So, on every synchronization, all locally completed writes are synchronized in remote copies and all writes in other copies are synchronized with the local copy. However, this can be avoided with two primitives:

1. **Acquire:** used to tell that a critical section is being entered. The local copy is brought upto date by bringing in all writes in all remote copies. Locally made changes are not guaranteed to be immediately propagated to other copies.
2. **Release:** used to tell that a critical section is being exited. All remote copies are brought in line with the writes in the local copy. Does not necessarily import remote changes.

Release Consistency II

Table 9 : A valid sequence of events for Release Consistency

P1:	Acq(L)	W(x)a	W(x)b	Rel(L)					
P2:					Acq(L)	R(x)b	Rel(L)		
P3:									R(x)a

References I

[1] Andrew S. Tanenbaum and Maarten van Steen:
Distributed Systems: Principles and Paradigms
Pearson Education (2002)