

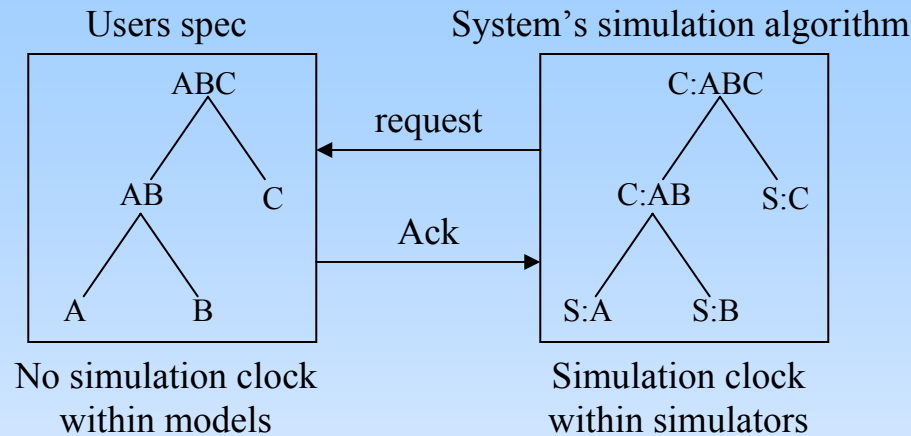
DEVS Model = Specification in Functions/Relations with time constraints

⇒ Simulation = Execution of DEVS Model

⇒ Execution of DEVS Model = Call DEVS Functions/Relations

Simulation Algorithm = Call DEVS functions/relations at the specified time

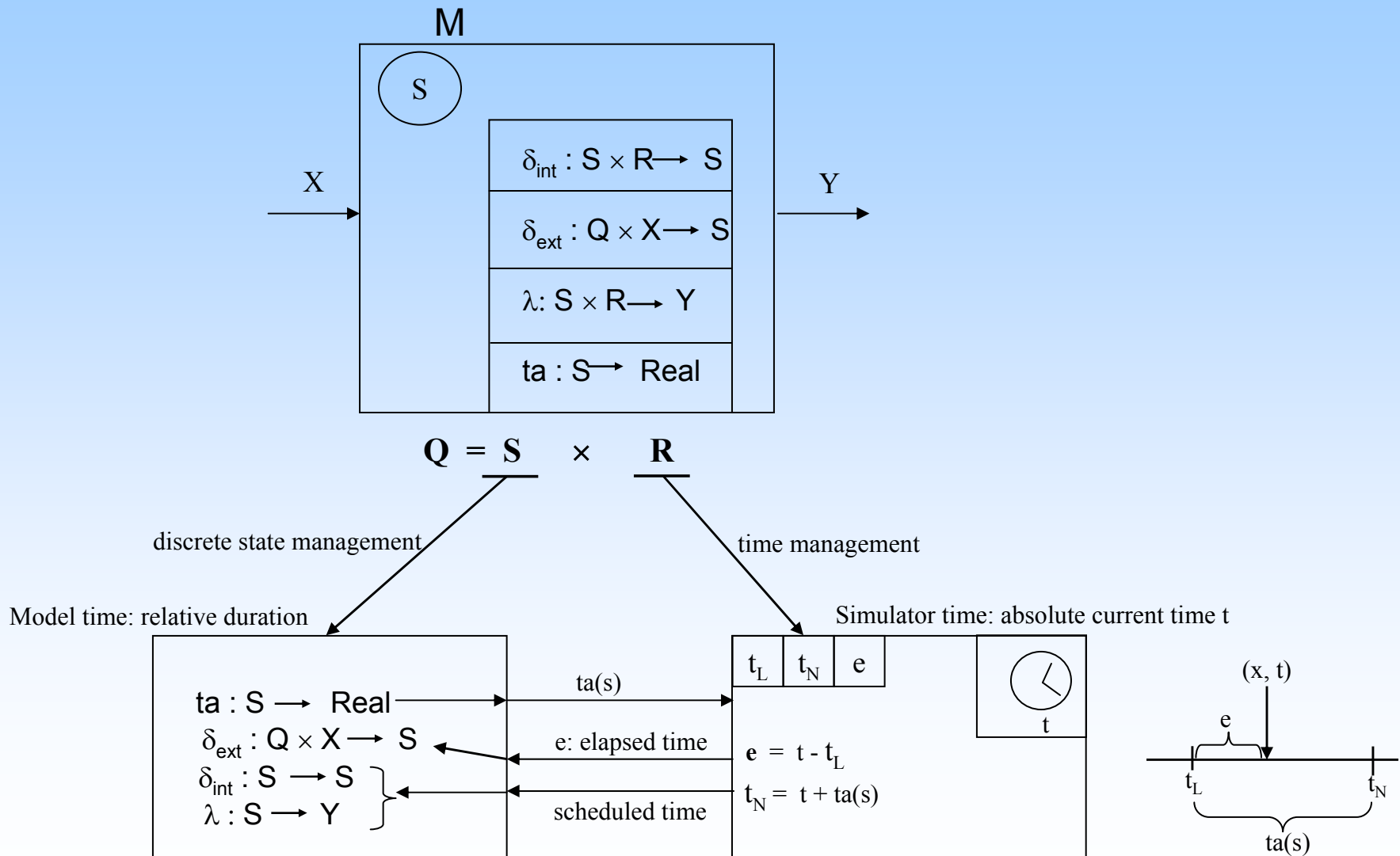
	DEVS Funtions	When Funtions are called
<b>Atomic DEVS</b>	$ta : S \rightarrow R$ (real) $\lambda : S \times R \rightarrow Y$ $\delta_{int} : S \times R \rightarrow Q$ $\delta_{ext} : Q \times X \rightarrow Q$	after each state transition when $r = ta(s)$ at $q = (s, r)$ when $r = ta(s)$ at $q = (s, r)$ when $X$ is received
<b>Coupled DEVS</b>	$EIC \subseteq DN.X \times \cup_i M_i.X_i$ $EOC \subseteq \cup_i M_i.Y_i \times DN.Y$ $IC \subseteq \cup_i M_i.Y_i \times \cup_j M_j.X_j$ $SELECT : 2^M - \emptyset \rightarrow M$	when $X$ is received when $Y_i$ is generated when $Y_i$ is generated when schedule conflict happens <div style="display: inline-block; vertical-align: middle; margin-left: 10px;">                         } Message routing                     </div>

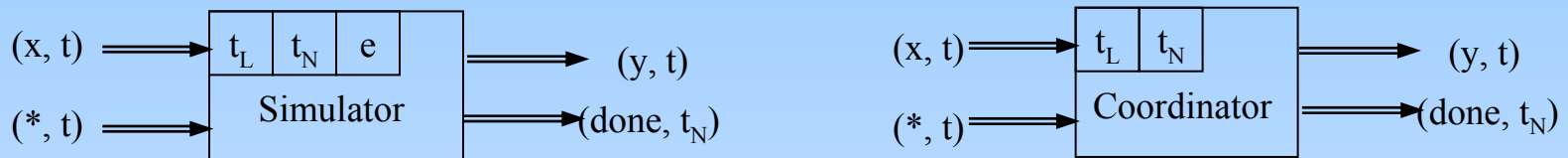


S : AM = Simulator for Atomic model AM (simulation algorithm for DEVS AM)  
C : CM = Coordinator for Coupled model CM (simulation algorithm for DEVS CM)

- Simulator S:AM  $\Leftrightarrow$  atomic AM
  - (i) simulation clock update
  - (ii) call 4 Atomic DEVS functions timely for transition, output, and scheduling
- Coordinator C:CM  $\Leftrightarrow$  coupled Model CM
  - (i) simulation clock management
  - (ii) call 4 Coupled DEVS functions timely for message routing and conflict resolution
- Root-coordinator  $\Leftrightarrow$  Outmost coordinator  $\Rightarrow$  Overall time management

# Time Specification in Model and Simulator





## Timing Variables

$t$  : current time  
 $t_L$  : time of last event  
 $t_N$  : time of next event  
 $e$  : elapsed time

## Messages

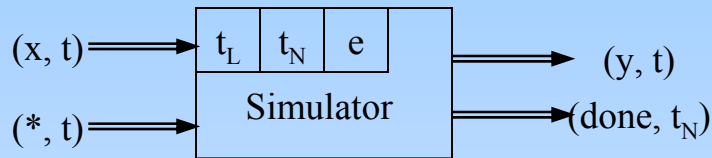
$(x, t)$  : external input event arriving at time  $t$   
 $(*, t)$  : internal event notifying that schedule time comes  
 $(y, t)$  : output event generated at time  $t$   
 $(done, t_N)$  : synchronization event notifying next event time is  $t_N$

## Hierarchical Simulation Algorithm: Hierarchical Simulator 6 of 11

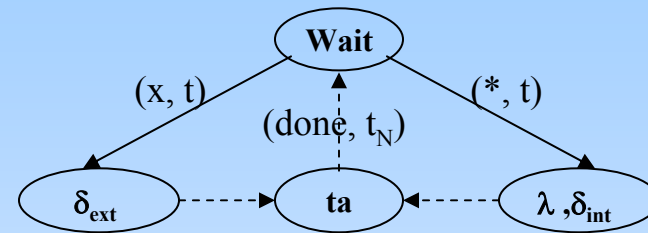
---

- ◆ Architecture of Hierarchical Simulator = Architecture of DEVS Model
- ◆ Co-operation of Simulator (Atomic DEVS) and Coordinator (Coupled DEVS)
- ◆ Hierarchical Algorithm
  - ❖ Computation of Next Schedule Time (No global event list)
  - ❖ Synchronization
  - ❖ Message routing
- ◆ Distributed Algorithm
  - ❖ No Global Information Shared
  - ❖ Implementation in Sequential Machine
  - ❖ Implementation in Distributed/Parallel Machine
- ◆ Conversion of  $t_a$ (Duration) to Simulation Clock(Absolute Value)

# Hierarchical Simulation Algorithm: Simulator



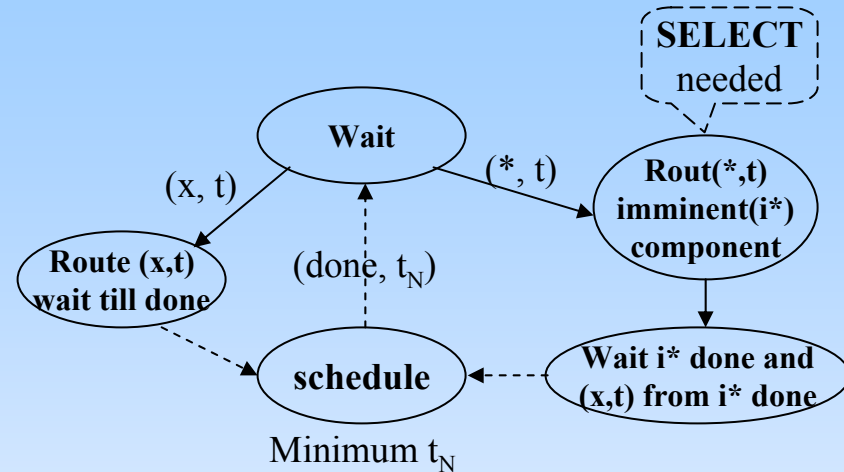
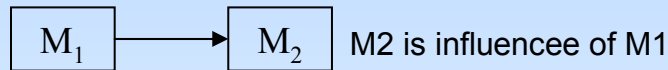
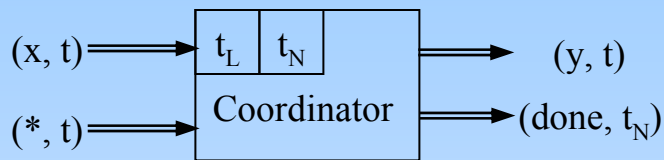
$t_L$  : time of last event  
 $t_N$  : time of next event  
 $e$  : elapsed time



**when receive (x,t)**  
 $done := false$   
 if  $t_L \leq t \leq t_N$  then  
 $e := t - t_L$   
 $s := \delta_{ext}((s,e),x)$   
 $t_L := t$   
 $t_N := t + ta(s)$   
 else error  
 $done := true$

**when receive (\*,t)**  
 $done := false$   
 if  $t = t_N$  then  
 $y := \lambda(s)$   
 $s := \delta_{int}(s)$   
 $t_L := t$   
 $t_N := t + ta(s)$   
 else error  
 $done := true$

# Hierarchical Simulation Algorithm: Coordinator



**when receive (\*,t)**

```

done := false
if t = t_N then
    find component(s) with t_N
    select one i*
    send (*,t) to i*
    translate y_{i*} to x
    send x to its influences
    wait i* and its influences done
    t_L := t
    t_N := min{t_{Ni} | i: i* + its influences }
else error
done := true
    
```

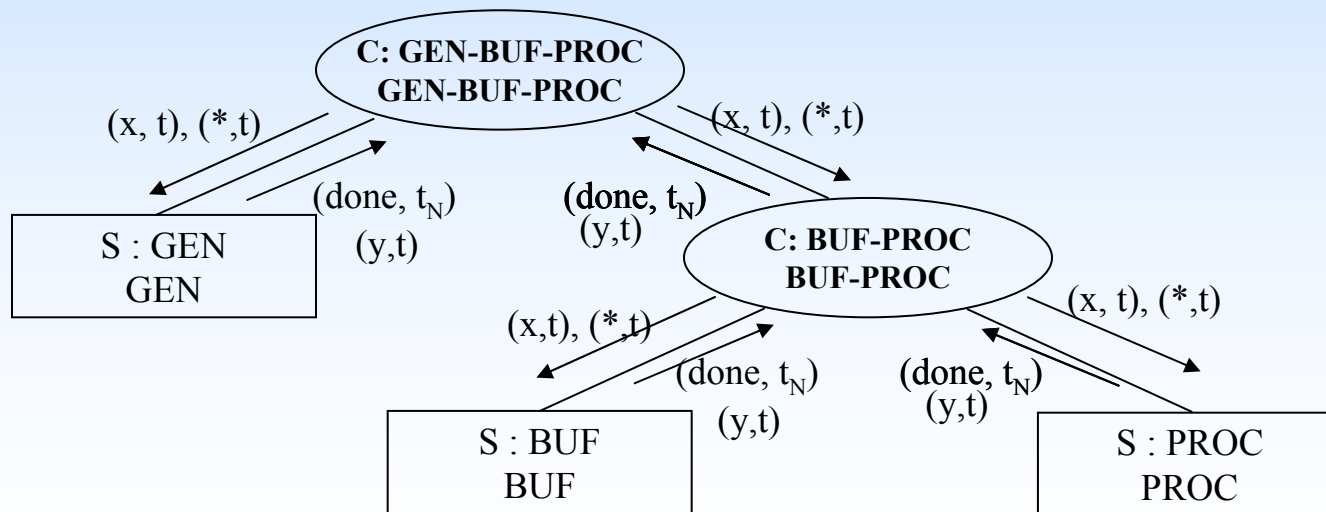
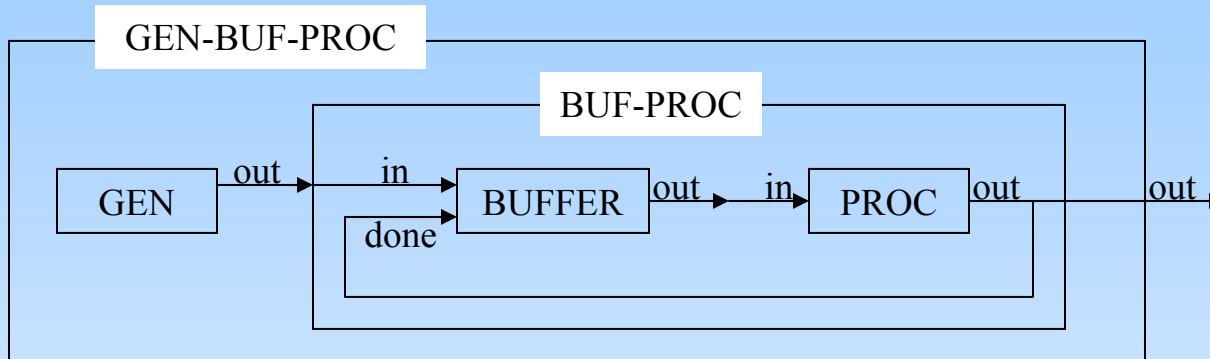
**when receive (x,t)**

```

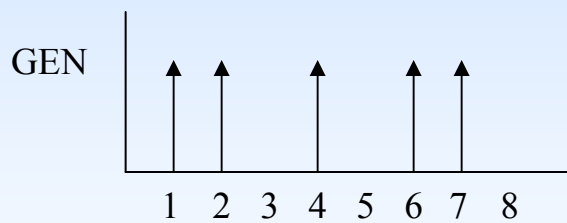
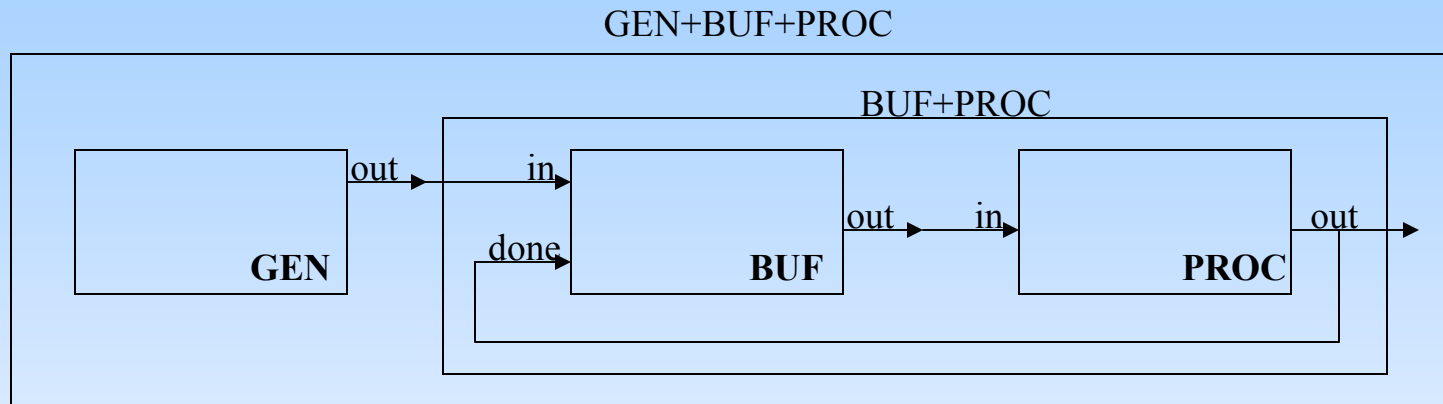
done := false
if t_L ≤ t ≤ t_N then
    send (x,t) to connected component(s)
    wait all component(s) done
    t_L := t
    t_N := min{t_{Ni} | i: component}
else error
done := true
    
```



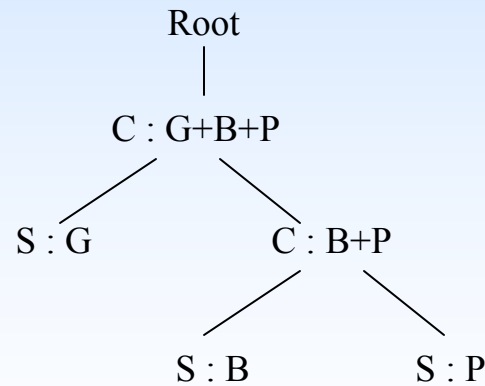
# Hierarchical Simulation: Example



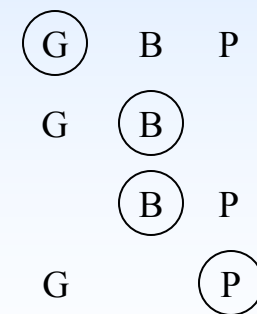
# Example of Hierarchical Simulation



ta(BUF) : 2 1 3 2 1  
 ta(PROC) : 1 2 1 3 2



**SELECT**



# Simulation Procedure of G-B-P Model

	S : GEN	S : BUF	S : PROC	C : B+P	C : G+B+P	ROOT	t
(done, $t_N=1$ )	①				→ ②	→ ③	0
(* , 1)	③ ←				← ②	← ①	t = 1
	$\lambda(s)$						
	$\delta_{int}(s)$	→	$\delta_{ext}(s)$				
	$t_a = 1$		$t_a = 2$				
schedule = min{2,3}	$t_N=t+t_a=2$	$t_N=t+t_a=3$					
(done, 2)	①				→ ②	→ ③	
(* , 2)	③ ←				← ②	← ①	t = 2
	$\lambda(s)$						
	$\delta_{int}(s)$	→	$\delta_{ext}(s)$				
	$t_a = 2$		$t_a = \{1\}$				
schedule	$t_N=4$	$t_N=3$					
(done, 3)		①			→ ②	→ ③	
(* , 3)		③ ←			← ②	← ①	t = 3
		$\lambda(S)$					
		$\delta_{int}(s)$	→	$\delta_{ext}(s)$			
		$t_a = 1$		$t_a = 1$			
schedule	$t_N=4$	$t_N=4$	$t_N=4$				
(done, 4)	①				→ ②	→ ③	
(* , 4)	③ ←				← ②	← ①	t = 4
	$\lambda(s)$						
	$\delta_{int}(s)$	→	$\delta_{ext}(s)$				
	$t_a = 2$		$t_a = 2$				
schedule	$t_N=6$	$t_N=4$	$t_N=4$				
(done, 4)		①			→ ②	→ ③	