# Program Modeling Concepts:
## Lesson-4: SYNCHRONOUS DATA FLOW GRAPH MODELS

# Data Flow Graph Model and CDFG Assumptions

- A set of the outputs from a circle showing a set of programming steps will be simultaneously available as the inputs to the next circle (next set of programming steps).
- At each circle computations take same time for each output and there is no delay in any of the inputs.

# Synchronous Data Flow Graph (SDFG) Model

- DFG assumptions not true in several cases
- SDFG models the delay in getting an input (s) before the all or part of computation(s) fire at a circle in the DFG
- Models the number of outputs from the program steps at a circle and model the number of inputs

# SDFG Model

- An SDFG model shows the delay(s) [expected due to a previous process may not yet have been completed] as well as shows the number of inputs and outputs.

# SDFG Model

- An arc (directed arrow) represents a buffer in physical memory.
- The arc can contain  one or more initial tokens with the delays.
- A token till received at the vertex (shown by a circle) does not fire the computations (actors) at a vertex (circle).

# SDFG Model

- An initial token may also represent a *delay* that is shown by a dot on the edges of an SDFG
- If there are more than one initial token, the number of initial tokens are mentioned on the dot.

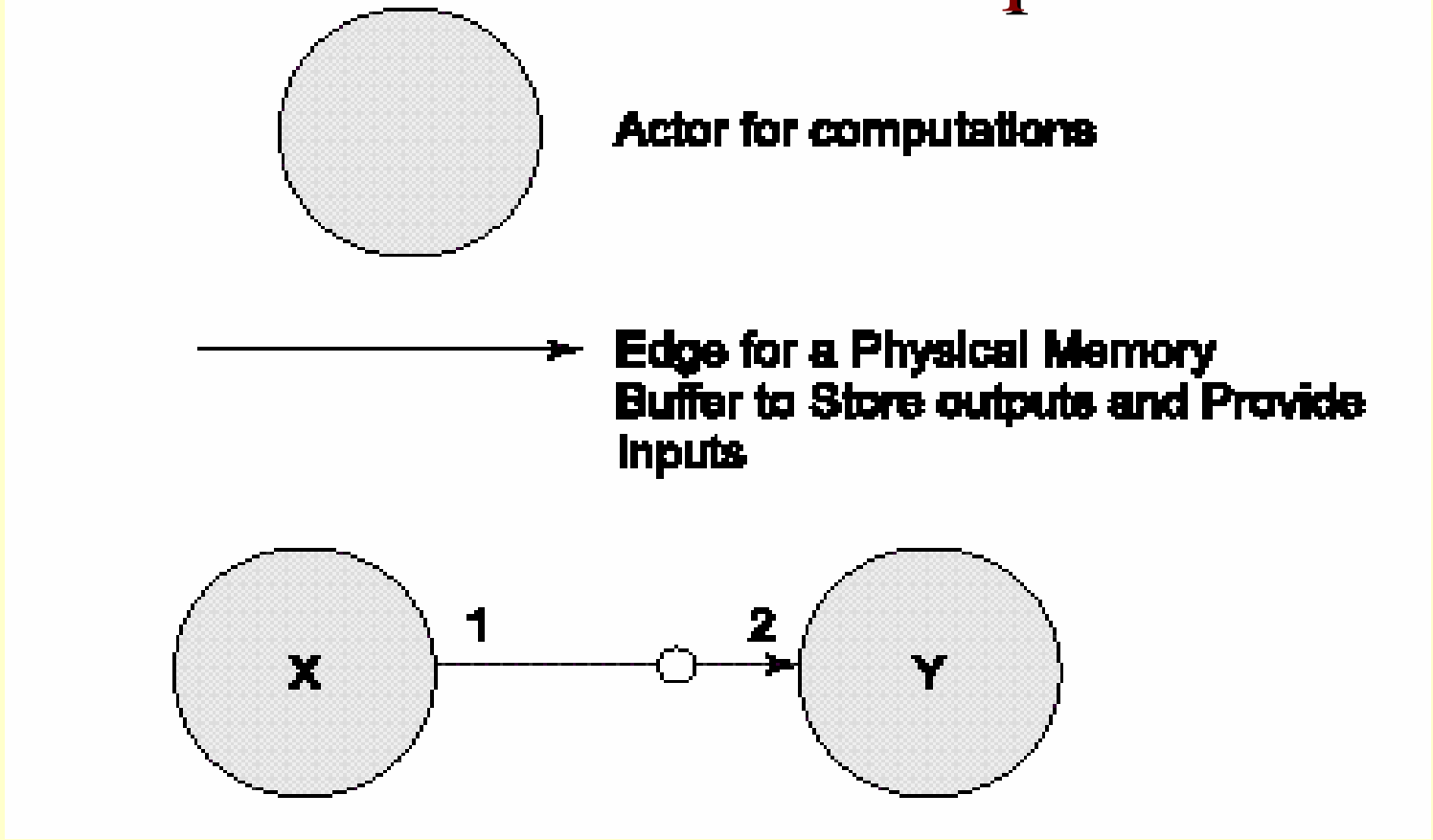
# SDFG Model Example 1

- SDFG model of computations (actors) at the vertices shown by circles) and memory buffer by arcs in a directed graph between X and Y

  —The delay in the input is shown by dot.

  — Number of inputs and outputs also marked

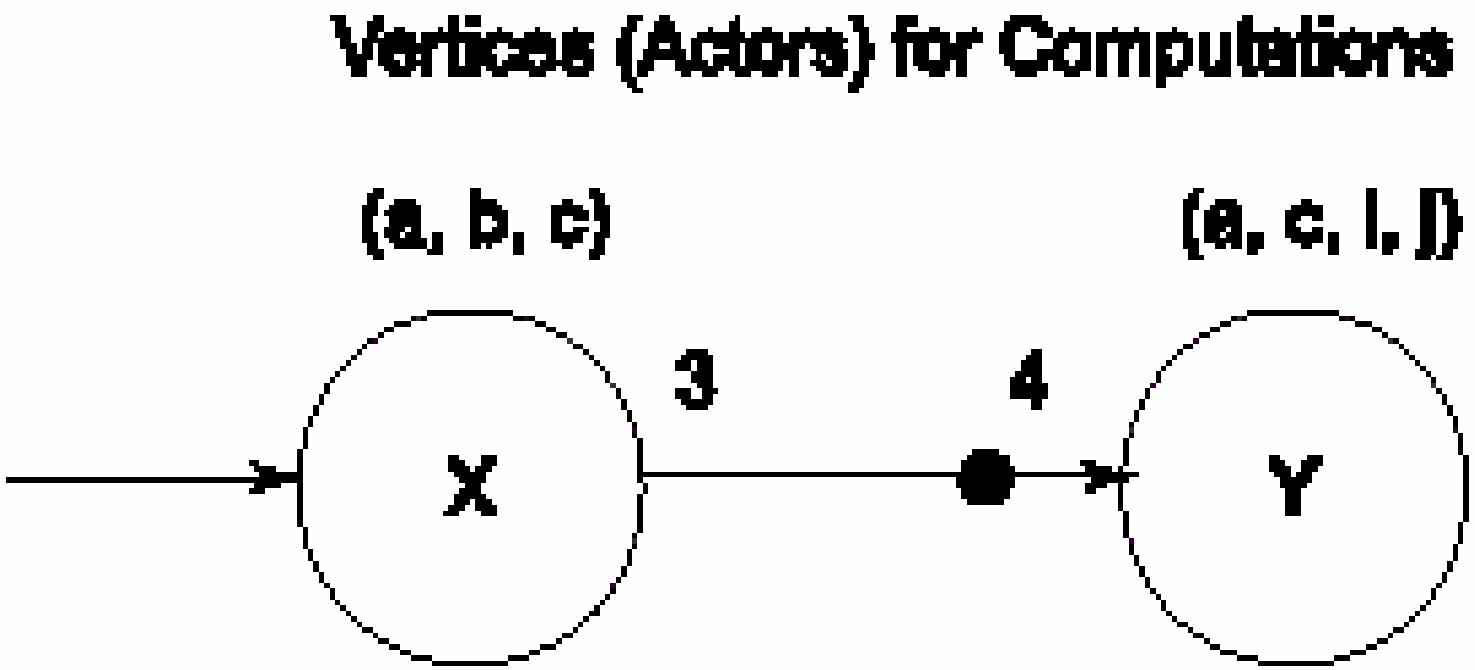
# SDFG Model Example 1



# SDFG Model Example 2

- Actors (Vertices shown by circles) and arcs in a directed graph between X and Y.
  —The outputs are *a*, *b* and *c* and inputs are *a*, *c*, *i* and *j*. The *i* is with a delay (dot).

# SDFG Model Example 2



# HSDFG

# Homogenous Synchronous Data Flow Graph (HSDFG) Model

- When there is only one token at the input, and one at the output, an SDFG is called a homogenous SDFG (HSDFG).

# HSDFG Model

- For example, if the outputs from vertex X' (a set of computations) is *a* and input to Y' (another set of computations) is also *a*, the SDFG is equivalent to an HSDFG. An SDFG can always be unfold into the HSDFG(s)

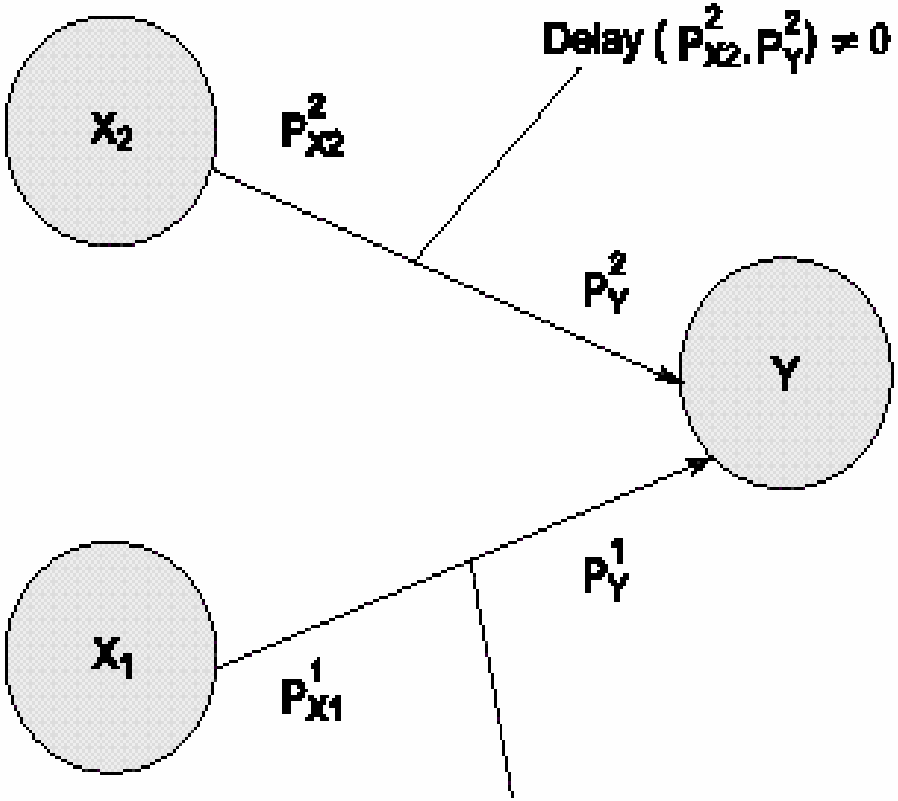
# Example of an SDF graph unfolded into one or more HSDFGs

- Two vertices can be connected by two or more edges in the HSDF graph.
- An HSDF graph will thus naturally have more vertices and edges than an SDFG because only one token is permitted at a vertex.

# HSDFG Model Example

- HSDFG model of actors (Vertices shown by circles) and memory buffer after unfolding an SDFG in which a vertex gives (a, b, c) outputs

# HSDFG Model Example




Token i is from $P_{X1}$ and
Token j is from $P_{X2}$

Delay ($P^1_{X1}$, $P^1_y$ ) = 0 between the output Port $P^1_{X1}$

And Input Port $P^1_Y$

Superscript 1 means first output Token and 2 means Second Output token that firew after a delay with respect to the first

# APEFG

# Acrylic Precedence Expansion Graph (APEG) Model

- Acrylic precedence is a precedence of vertices in a directed graph such that there are no delays at the arcs.
- If initial tokens (delays) are taken off from an HSDF graph, an acrylic precedence expansion graph (APEG) is obtained.

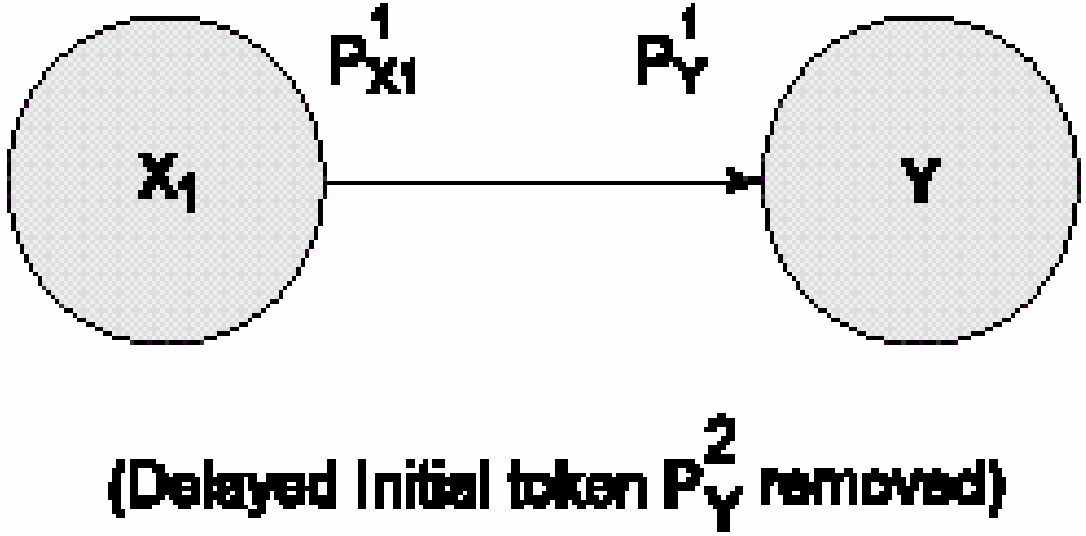
# An APEG

- An initial token does not represent *delay* and there is no dot on the edges of an SDFG or HSDFG

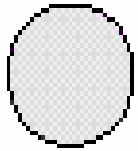
# Importance of an APEG

- APEG not only has along the arc, starting inputs identical to the output from a previous vertex, but also no delaying tokens.

# APEG model of actors (Vertices shown by circles) and memory buffer after unfolding an SDFG after removing the delayed path(s)

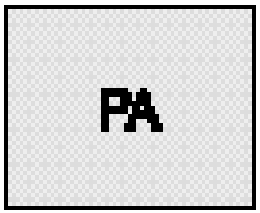


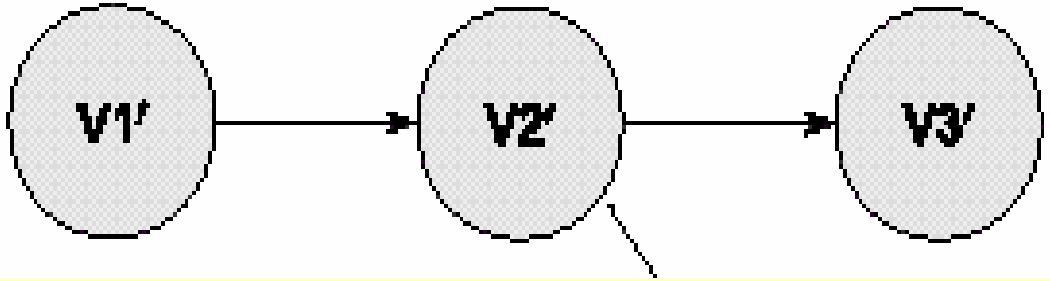
# APEG model of computations on a processor PA



Circle Vertex for Computations at PA or PB.

Directed arrow for the Outputs carried to next Vertex



# Summary

We learnt

- SDFG modeling for the processes
- HSDFG modeling for the processes
- APEG modeling for the processes

We learnt

- A Modeling method in a multiprocessor system is the use of the SDFG and HSDFG representations in which there is unfolding of the SDFG so that there is only one token which delays along its edge and /or an APEG in which there are no delays.

# End of Lesson 4 of Chapter 6