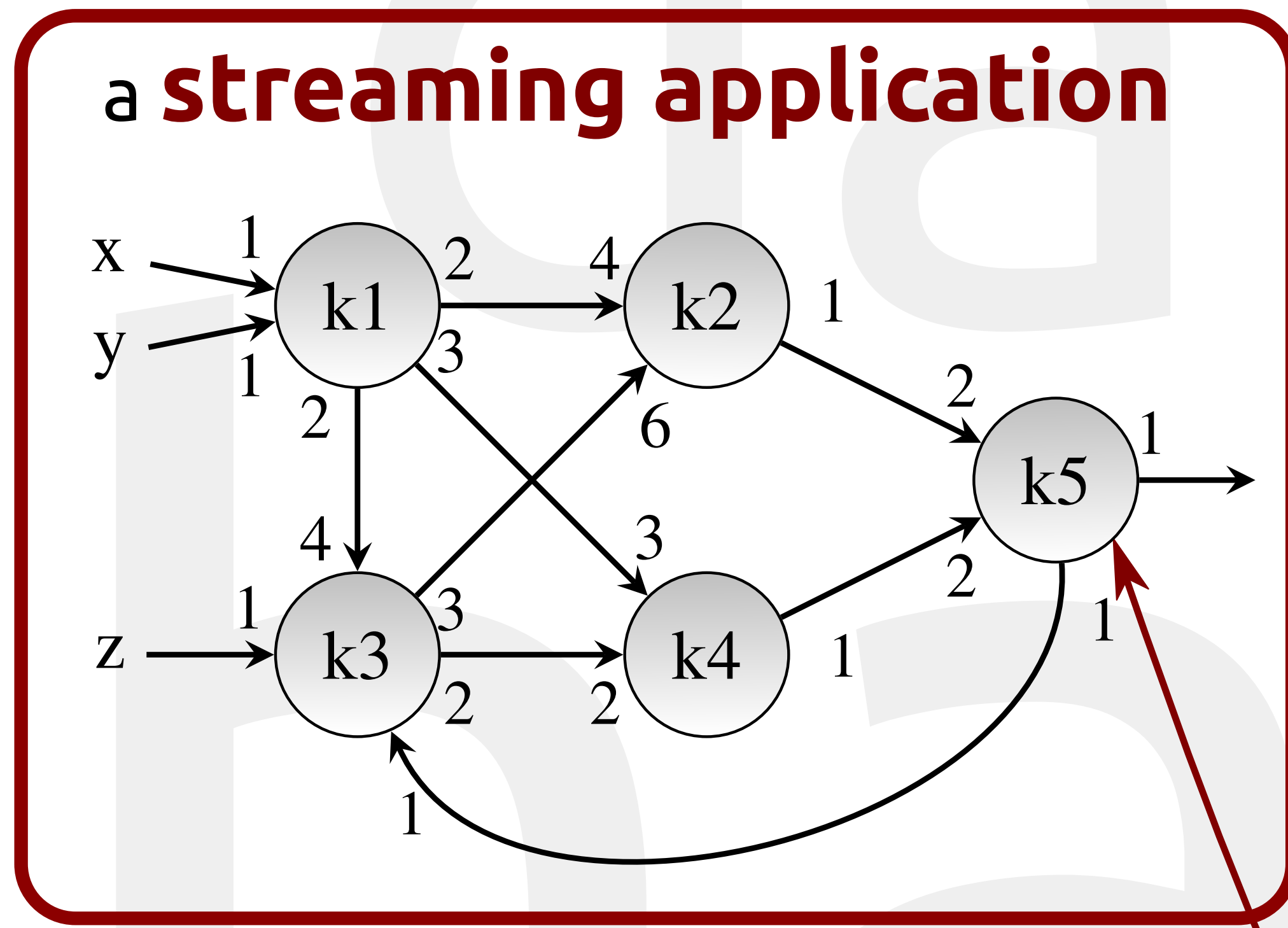


# High Level Structural Description of Streaming Applications

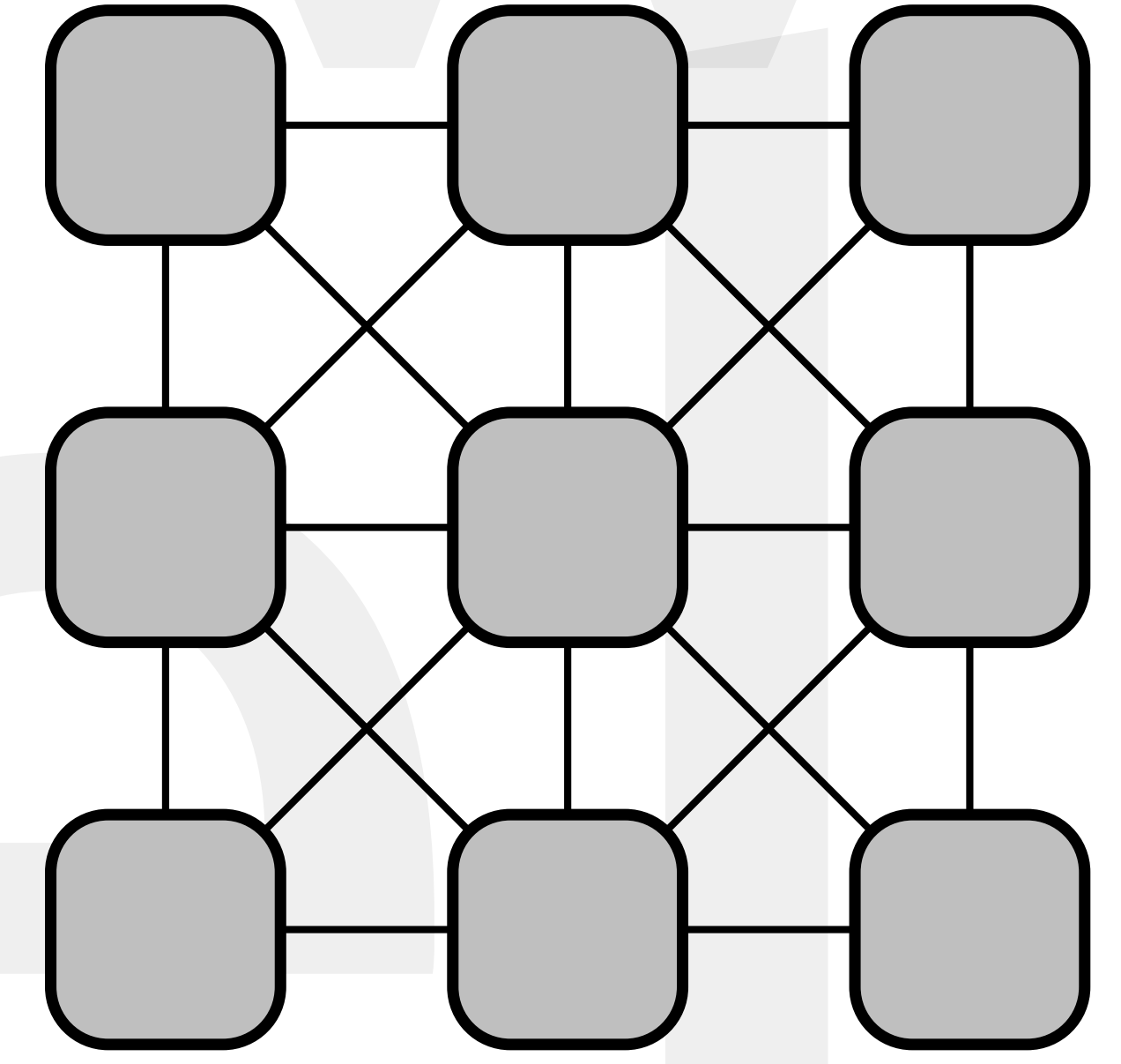
Anja Niedermeier, Jan Kuper, Gerard Smit  
 Computer Architecture for Embedded Systems  
 University of Twente, The Netherlands  
 a.niedermeier@utwente.nl



our goal is to

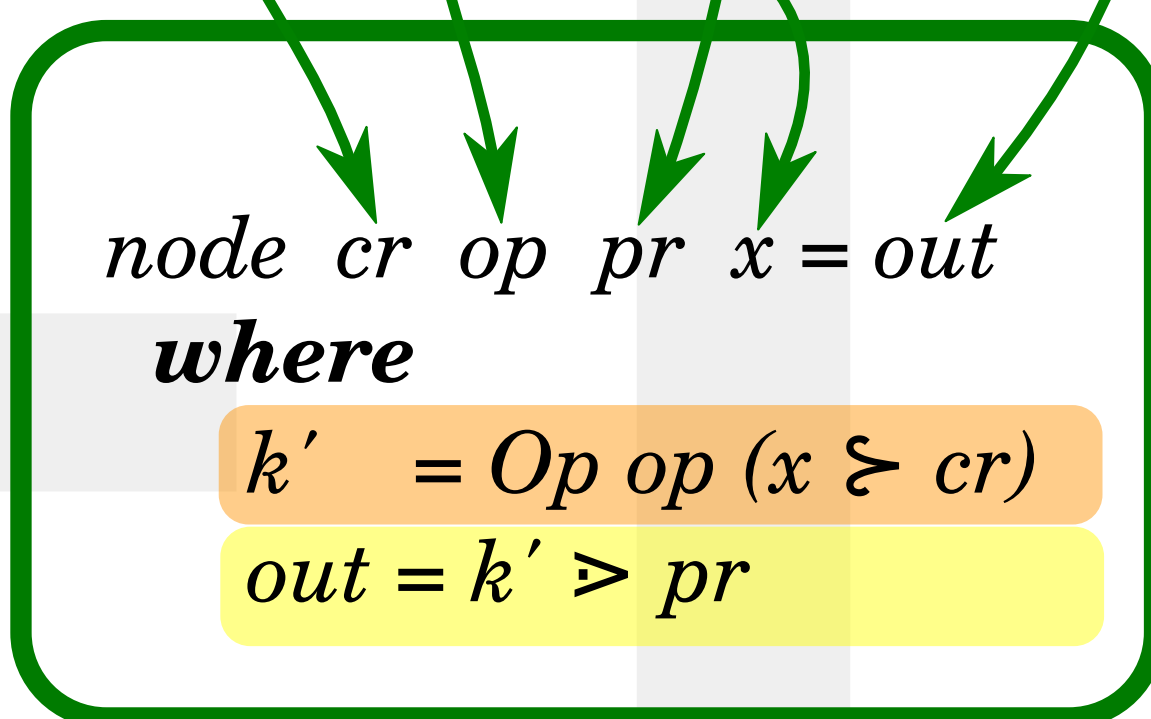
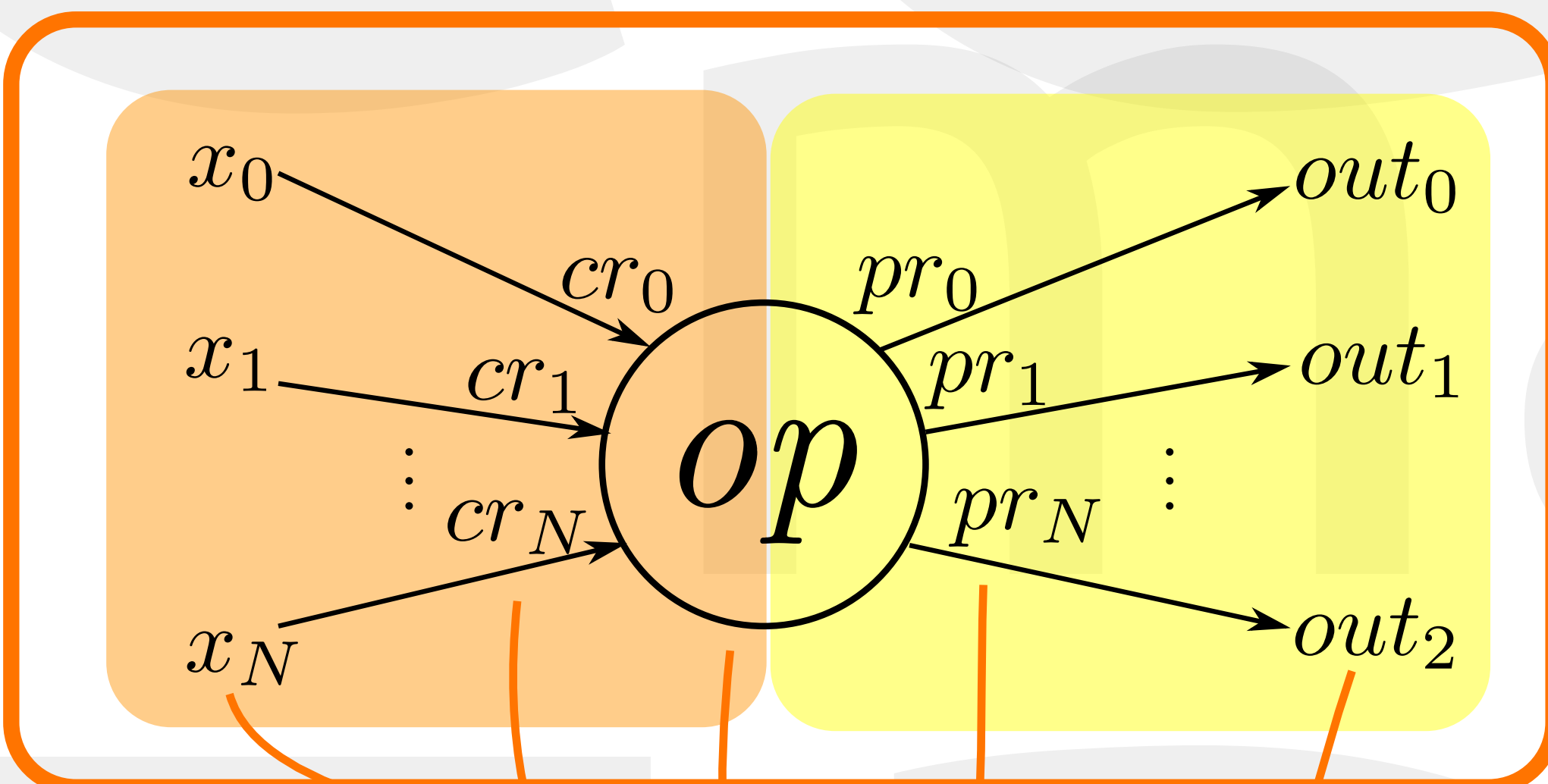
**map**

on a reconfigurable **multicore platform**



for that, we developed a **framework** to **describe streaming applications** on a **high** and **abstract level**

each **node** in the application is **characterised** in terms of its **input and output signals**, the **consumption** and **production rate**, and its **operation**



$k1 = node\ [1,1]\ Op_1\ [2,3,2]$   
 $k2 = node\ [4,6]\ Op_2\ [1]$   
 $k3 = node\ [1,1,4]\ Op_3\ [3,2]$   
 $k4 = node\ [1]\ Op_4\ [3,2]$   
 $k5 = node\ [2,2]\ Op_5\ [1,1]$

$data\ Expr = Op\ Opcode\ [Expr]$   
 $\quad | Input\ String$   
 $\quad | PR\ Int\ Expr$   
 $\quad | CR\ Int\ Expr$   
 $data\ Opcode = Op_1\ | Op_2\ | Op_3\ | Op_4\ | Op_5$

in the framework, **Haskell templates** are available to **describe each node** and the **data streams**

$input \triangleright node$

finally, the **streaming application** is **mapped** to the **target architecture**



the **complete streaming application** is then **defined** by its **dataflow structure**

$stream\ x\ y\ z = out$   
**where**  
 $k1' = [x, y] \triangleright k1$   
 $k2' = [k1'!!0, k3'!!0] \triangleright k2$   
 $k3' = [z, k5'!!0, k1'!!2] \triangleright k3$   
 $k4' = [k1'!!1, k3'!!1] \triangleright k4$   
 $k5' = [k2'!!0, k4'!!0] \triangleright k5$   
 $out = k5'$

