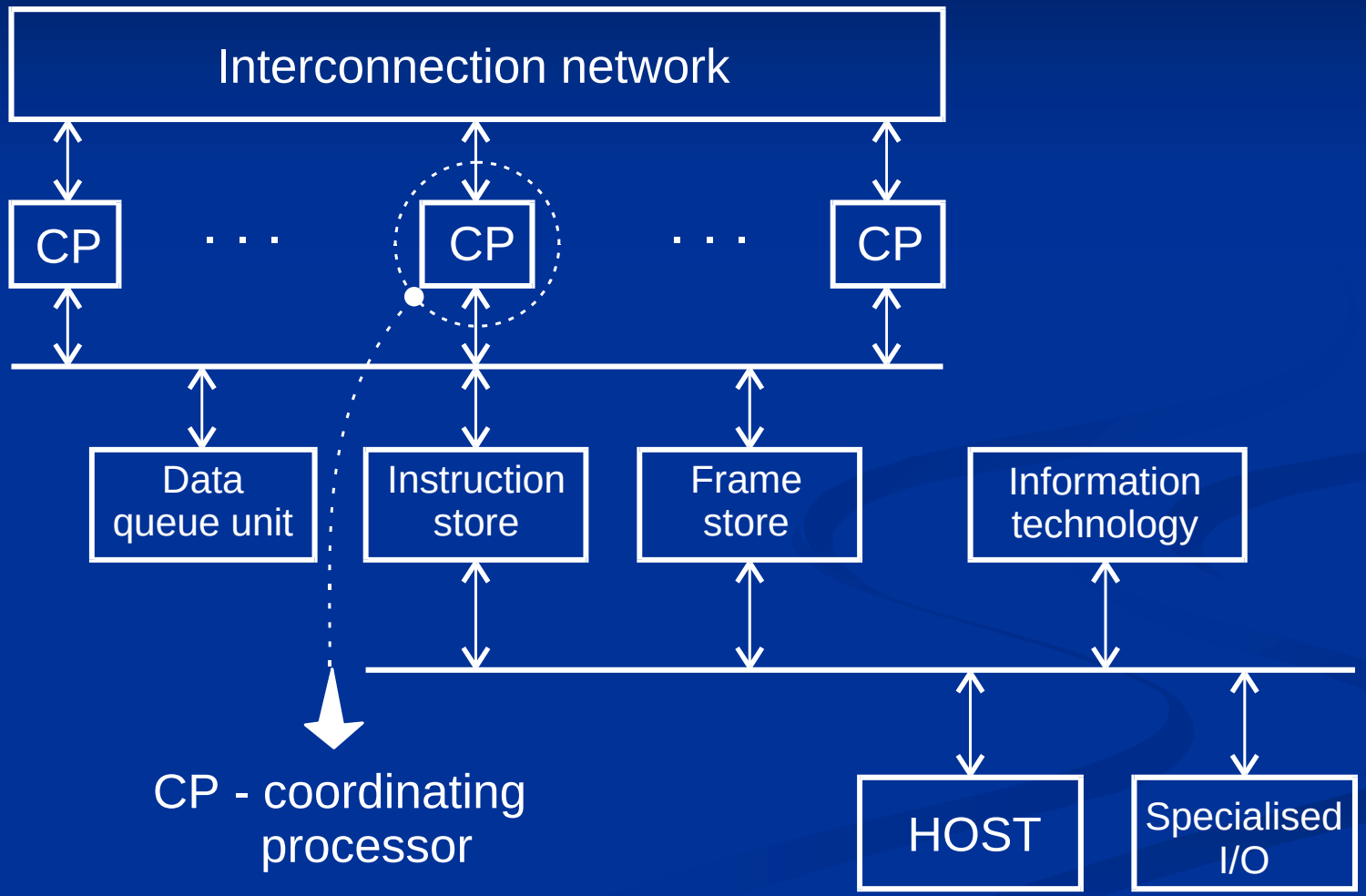
Simulation of Data Flow Architecture in Parallel environment

Marek STRAKA,
Branislav SOBOTA
Department of Computer and Informatics
Technical University of Košice

Introduction

- What is data flow architecture
- Coordinating processors
- Programming model of DF KPI architecture
- Simulation of DF KPI architecture
- Migration to parallel environment
- Using the GRID for simulation

Data Flow Architecture



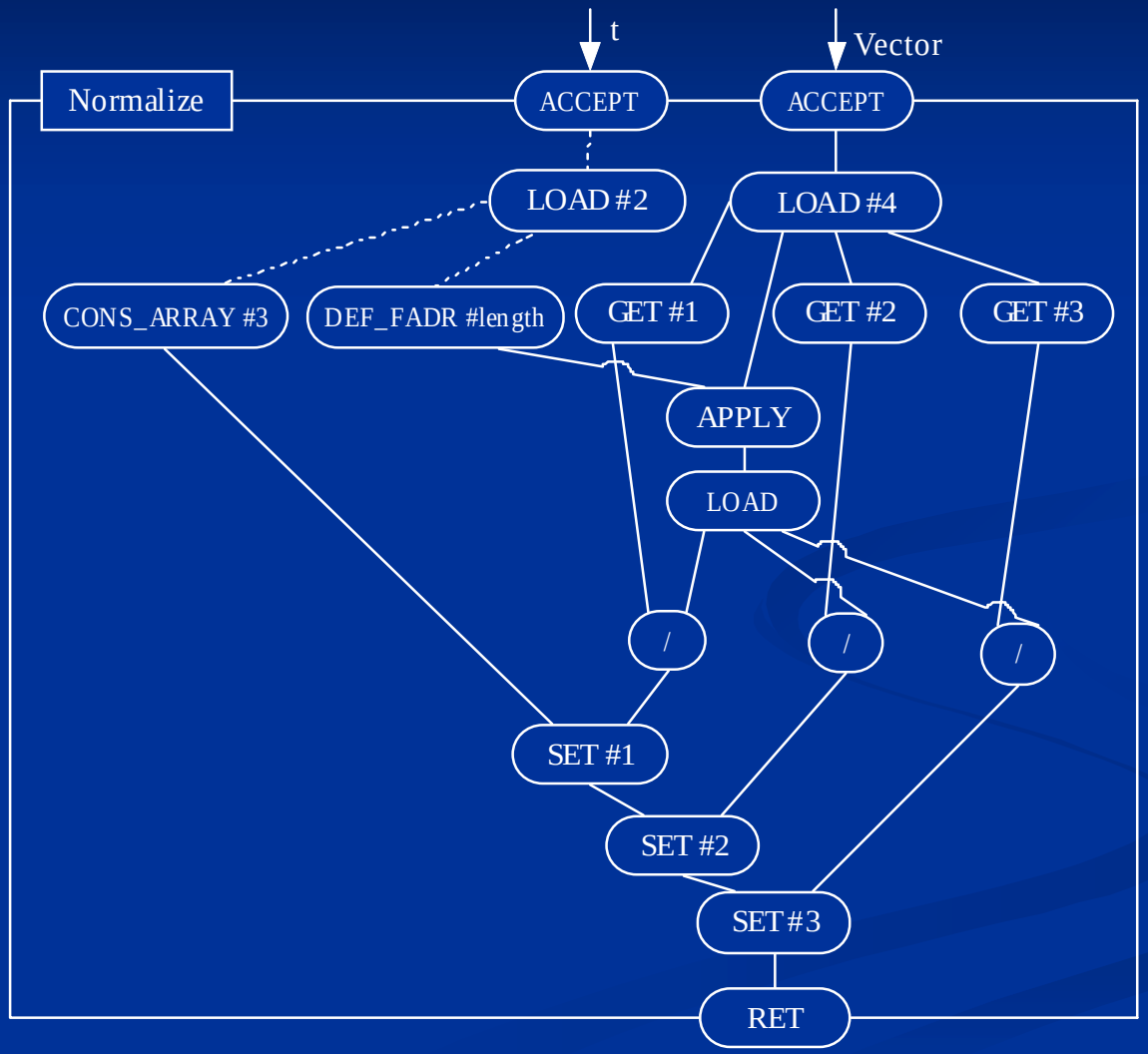
Program writing

- Description of problem
- Space partitioning problem (BSP Trees)
- Using functional languages
- Functional language HASKELL
- Building data flow graph DFG
- Simple transformation from Haskell to DFG

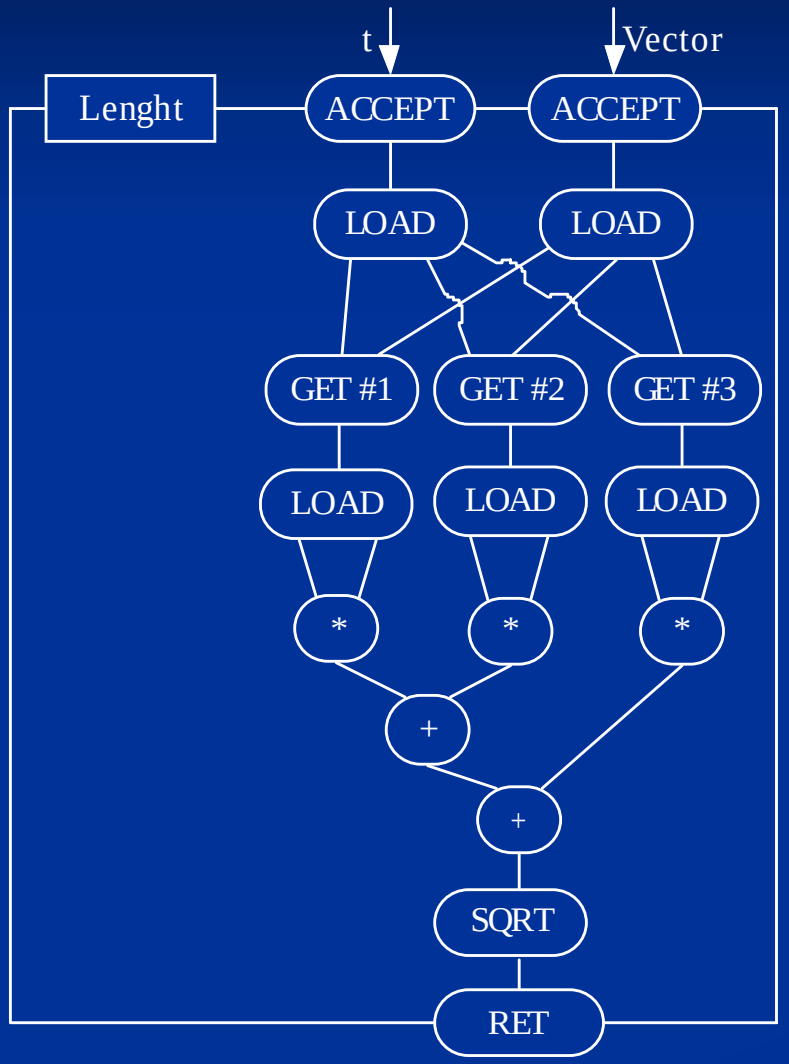
Example of Haskell code

```
normalize :: Vector -> Vector
normalize v =  ( Vector x y z)
    where
        dist = lenght v
        x =  vector_X v / dist
        y =  vector_Y v / dist
        z =  vector_Z v / dist

lenght :: Vector -> Double
lenght v = sqrt(vector_X v * vector_X v +
                vector_Y v * vector_Y v +
                vector_Z v * vector_Z v)
```

DFG of function Normalize



DFG of function Lenght



Example of Haskell source code:

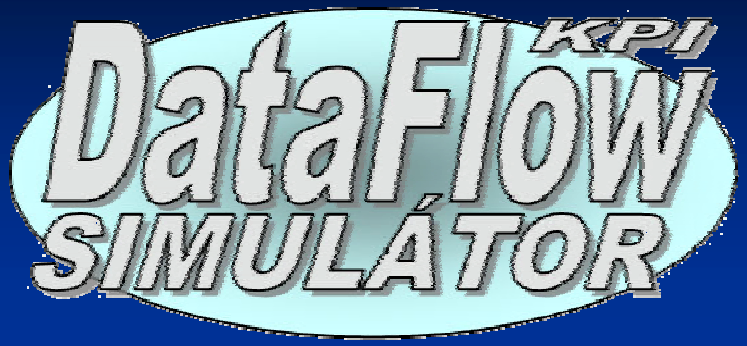
```
lenght :: Vector -> Double

lenght v = sqrt(  vector_X v * vector_X v +
                  vector_Y v * vector_Y v +
                  vector_Z v * vector_Z v )
```

Simulation tasks

- Emulation of instruction set
- 32 instructions (operators)
- 5 stages in coordinating processor
- Memory system – DQU, FS, IS
- Matching vector – MV
- Structures (array, list, tree)

Migration to parallel environment

- The precise simulator it is possible to obtain by using:
  - Grid solution
  - Cluster solution
  - Tightly coupled multiprocessor computer
- Coordinating processors = one node
- Communication traffic between coordinating processors



Questions ?

Marek.Straka@tuke.sk
Branislav.Sobota@tuke.sk