

# The software simulator of a parallel computing system with message passing

7<sup>th</sup> FRUCT seminar.

Eugene Gavrin

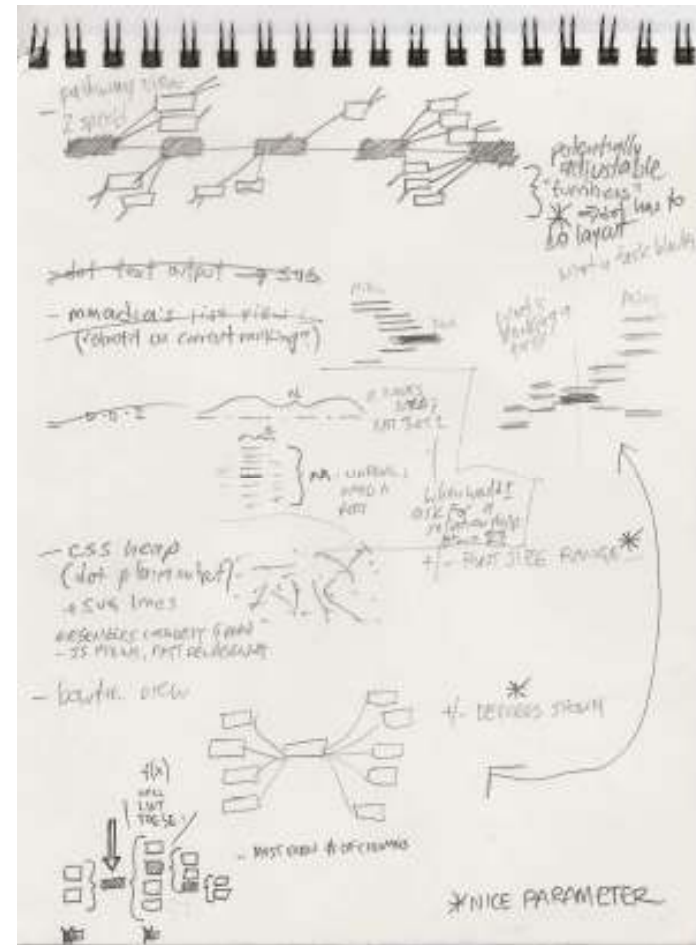
*eugene.gavrin@guap.ru*

# Introduction

- Information systems need to efficiently use massive distributed and parallel systems
- There is lack of tools aimed to development of parallel algorithms
- Requires ability to simulate and debug created algorithms

# Purposes

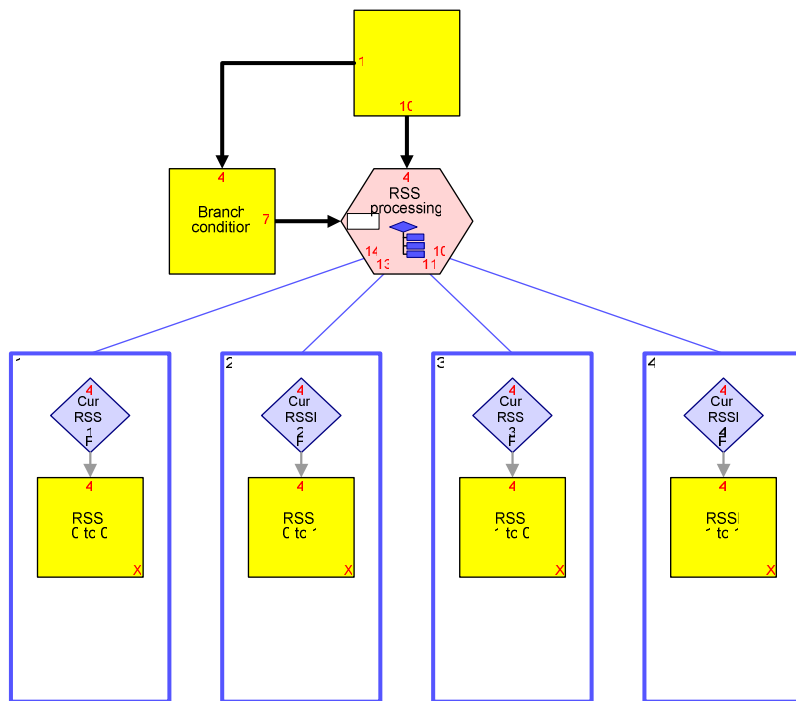
- Create tool aimed to simulate parallel algorithms
- Provide a mechanism to implement C/C++ code
- Allow execution of different computational tasks on independent processing elements
- Allow configuration of target platform
- Collect execution dump



# Existing alternatives

- Some number of hardware simulators like: NcSim, NoXim, NocSim and etc.
  - There are aimed to simulation of hardware.
  - Simulation of high-level algorithms becomes slower and more complicated.
- There are no software simulators.
  - + Hardware part is simulated only for adequate evaluation of algorithms

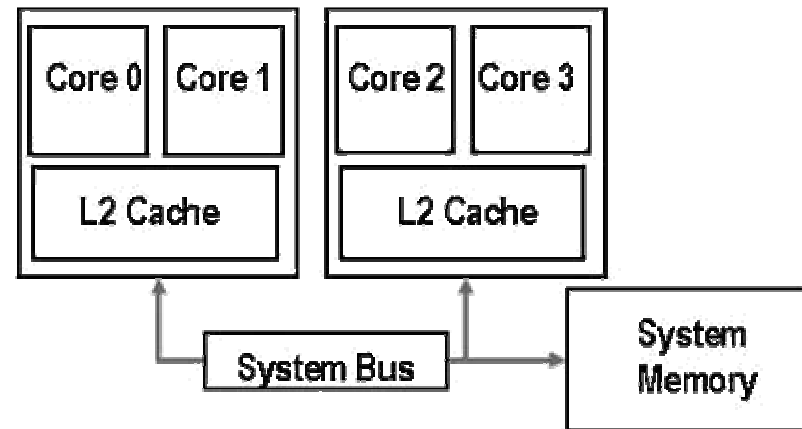
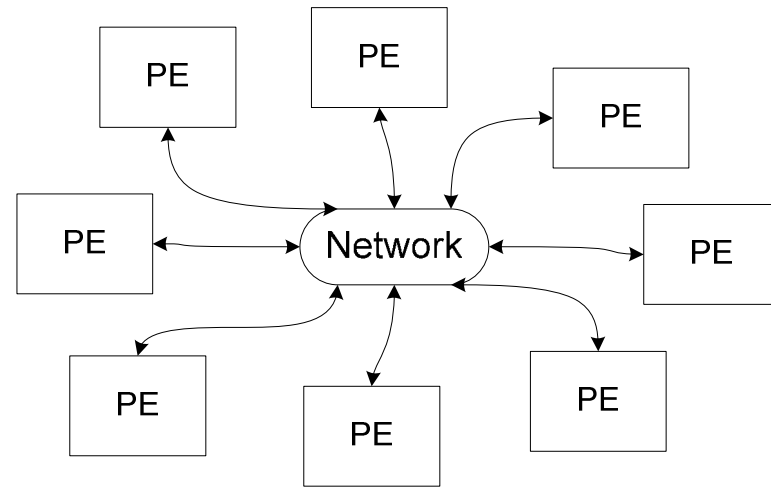
# Program



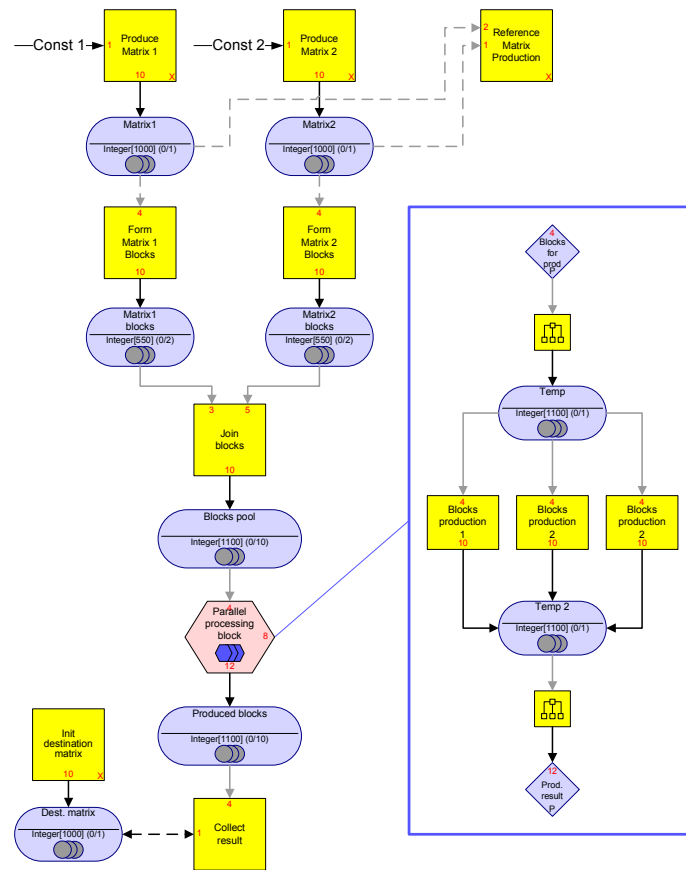
- Directed graph represents parallel program
- Nodes are active program elements
- Links are data dependencies

# Platform

- Set of processor elements connected to each other
- Can be configured in special file
  - Network bandwidth
  - Count of PE's
  - Type of PE's



# Computational tasks

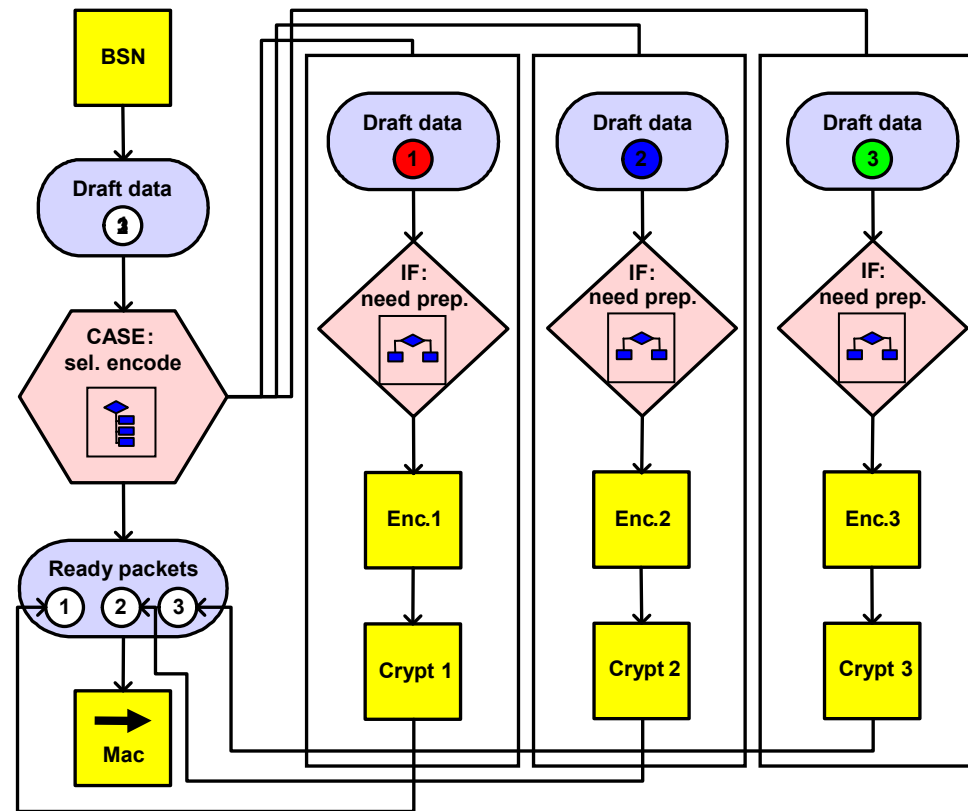


- Terminals
  - + Data transformation code execution
- Shared Data Objects
  - + Shared data in distributed systems
- Dynamic control
  - + Conditional unrolling
  - + Parallel iterative cycles
  - + Parallel conditional cycles

# Implemented dynamics

Supports:

- Dynamic operators generation and removal
- Dynamic branch unrolling



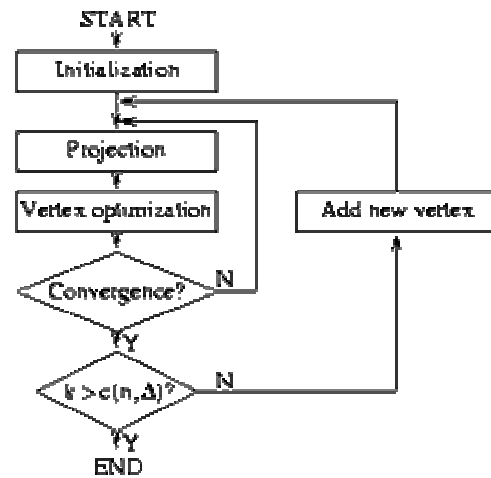


# Possible usage

- Create parallel algorithms
- Implement real functionality using existing C/C++ code
- Debug real algorithms on configurable platform
- Collect and analyze different dumps as a result of simulation

# How it works

Simulation



Comm. algorithm

xml

```
cmd D:\_work\Доклад - Симмулятор\sim.exe
[478]; proc = 50022; time = 4306500; finished
[478]; proc = 50002; time = 4308000; started
[478]; proc = 73098; time = 4309500; finished
[478]; proc = 50002; time = 4311000; finished
[479]; proc = 50022; time = 4312500; started
[479]; proc = 73098; time = 4314000; started
[479]; proc = 50022; time = 4315500; finished
[479]; proc = 50002; time = 4317000; started
[479]; proc = 73098; time = 4318500; finished
[479]; proc = 50002; time = 4320000; finished
[480]; proc = 50022; time = 4321500; started
[480]; proc = 73098; time = 4323000; started
[480]; proc = 50022; time = 4324500; finished
[480]; proc = 50002; time = 4326000; started
[480]; proc = 73098; time = 4327500; finished
[480]; proc = 50002; time = 4329000; finished
[481]; proc = 50022; time = 4330500; started
[481]; proc = 73098; time = 4332000; started
[481]; proc = 50022; time = 4333500; finished
[481]; proc = 50002; time = 4335000; started
[481]; proc = 73098; time = 4336500; finished
[481]; proc = 50002; time = 4338000; finished
[482]; proc = 50022; time = 4339500; started
[482]; proc = 73098; time = 4341000; started
[482]; proc = 50022; time = 4342500; finished
```



Output

- \*.vcd schemes
- Debug info
- Process execution info
- PE workload



Algorithm correction



# Summary

## Hardware part:

- PE types
- Task queue
- Data buffers

## Software part:

- Dynamic operators
- Shared data
- Code implementation
- Statistics on results of work

Thank you