Conceptual Models of Production Processes

The Digital Production Designer: an environment to conceptually design production process model-value functionality

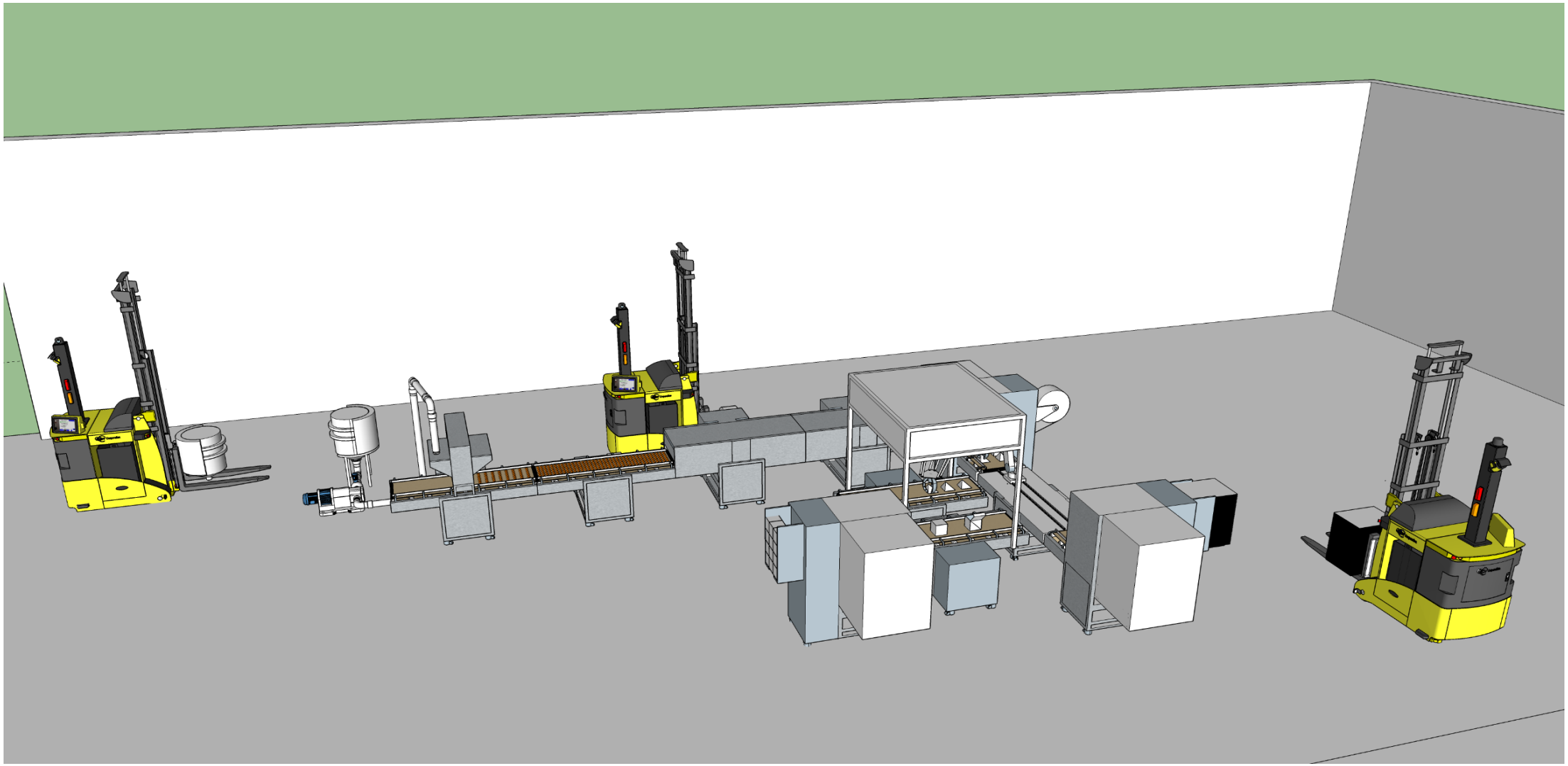
- Assoc.Prof Ion MIRONESCU
- ion.mironescu@ulbsibiu.ro
- Lucian Blaga University Sibiu

Flexible manufacturing systems

- ***Flexible manufacturing system (FMS)***
  - manufacturing systems able to reconfigure themselves
  - rapid reconfiguration
  - multiple types of products.
- A FMS consist of:
  - a set of workstations
    - Automatic execution of large sets of operations - *machine flexibility*
    - Cope with large-scale changes in **volume**, **capacity**, or **capability** demands.
  - a material handling system
    - flexible conveyors, automated guided vehicles (AGV) and loading-unloading robots - *routing flexibility*
    - changed to produce new product types, change the order of operations .
  - a complex command and control system - orchestrates the cooperation

Example

DIGIFOF.
DIGITAL DESIGN SKILLS FOR FACTORIES OF THE FUTURE



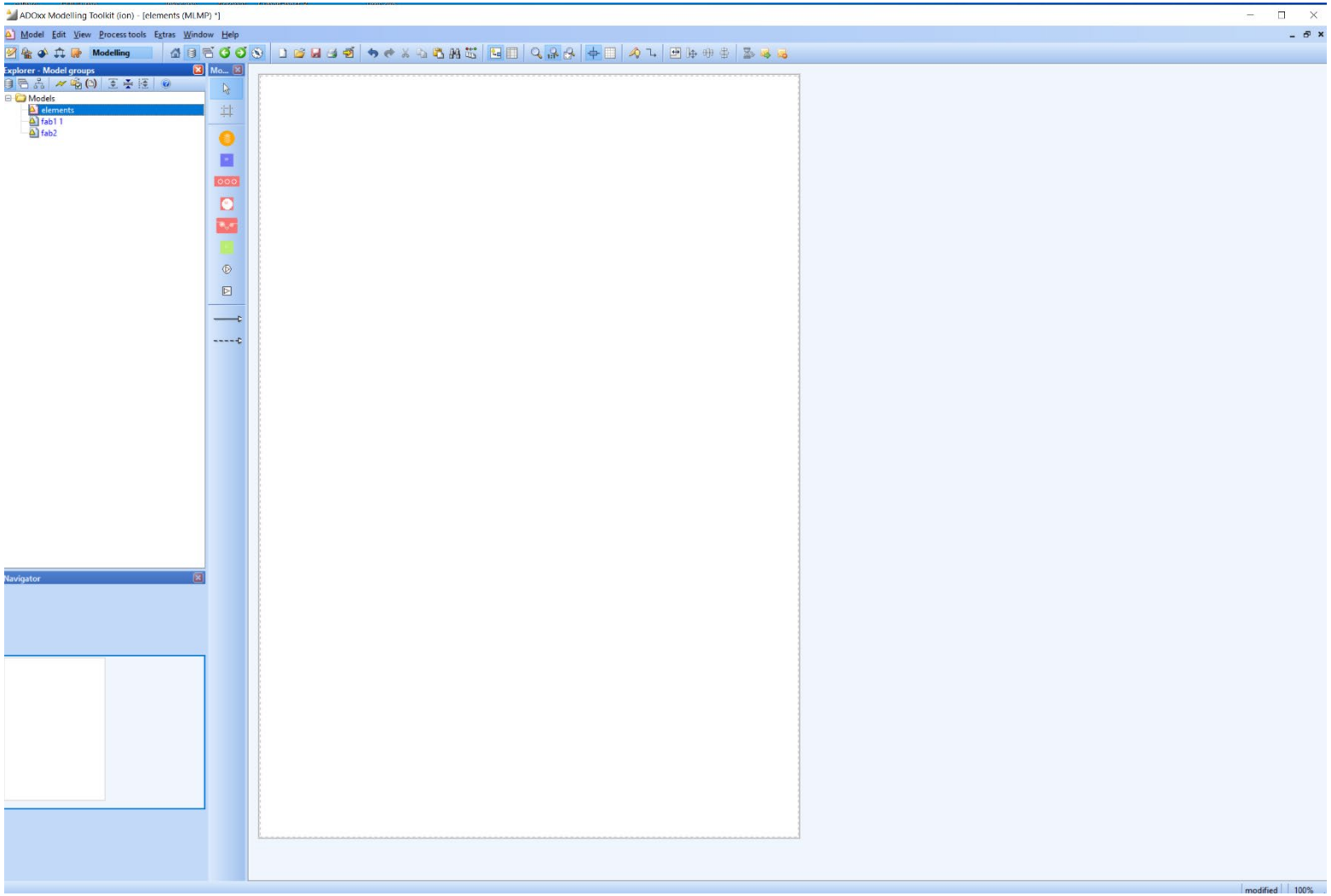
Targets

- Layout design
  - Model of the manufacturing line
    - Workstation with capabilities
    - Material handling system
  - Analysis of **volume**, **capacity** and **capability** of the represented system
- Operation design – Scheduling
- Automation – Control system design

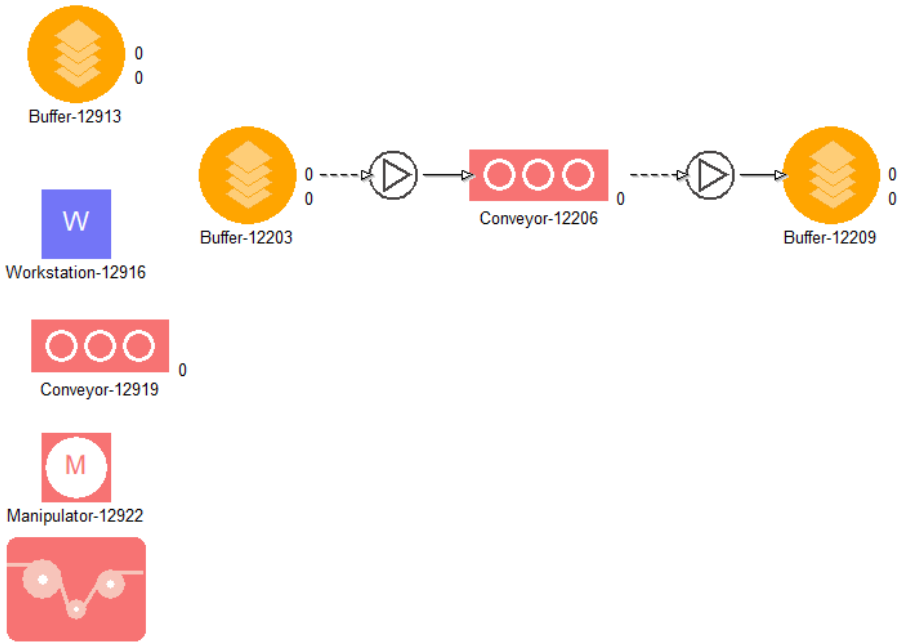
Layout Design Goals

- Connections & placement
- Assessment of general capabilities
- Design constraints
- Basic control loops

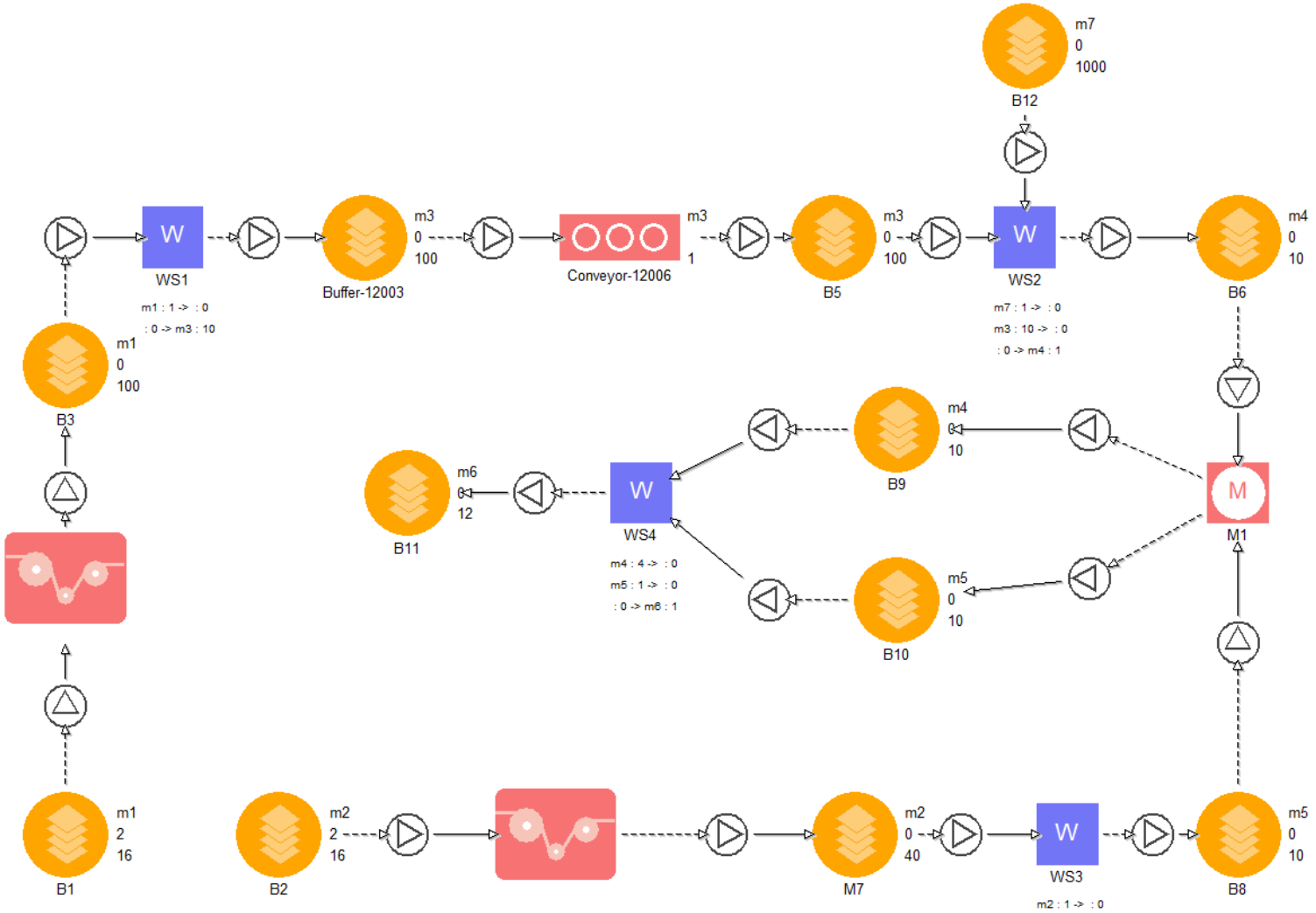
Digital Design Planner



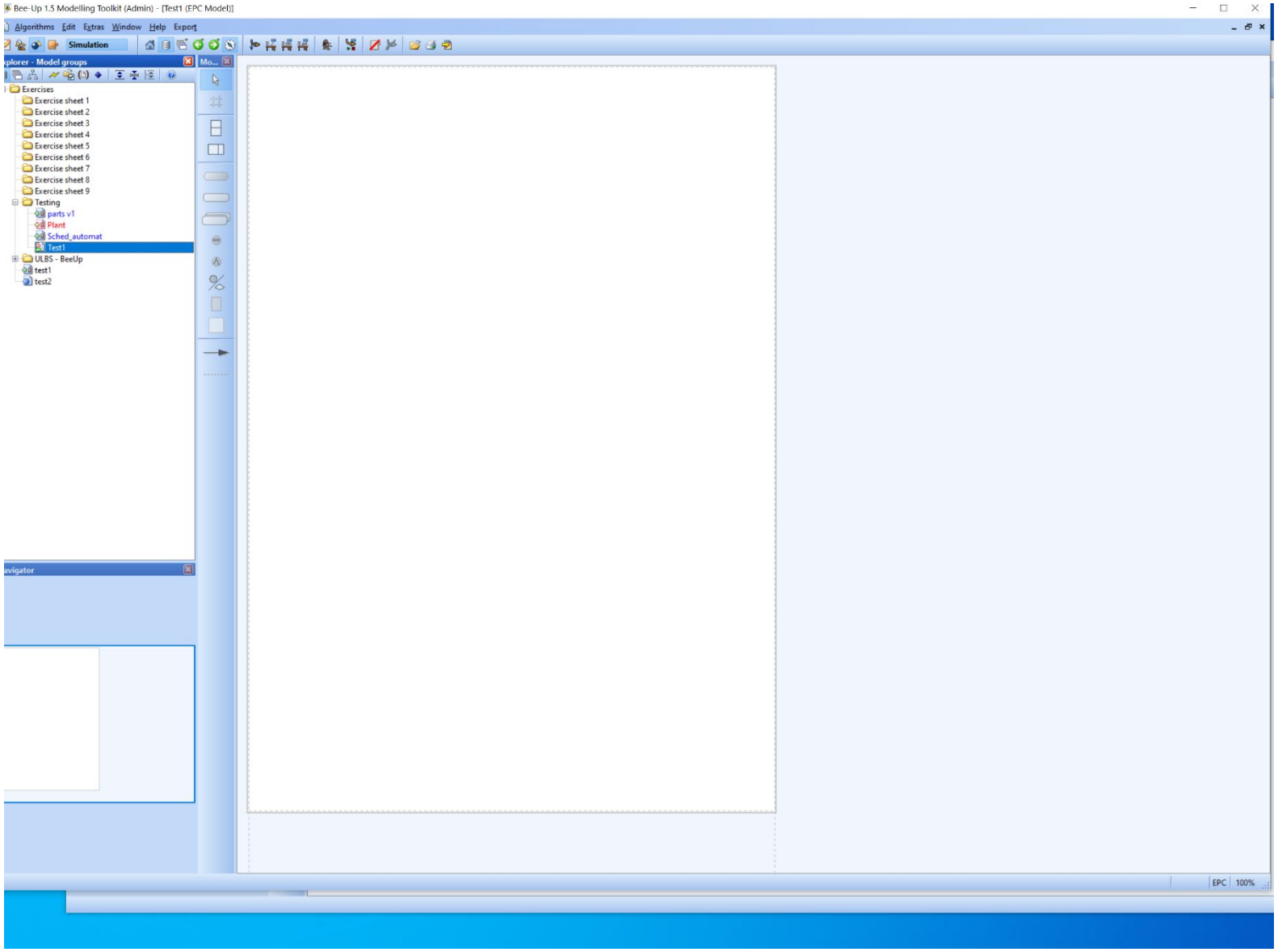
Modelling Language for Manufacturing Processes (MLMP)



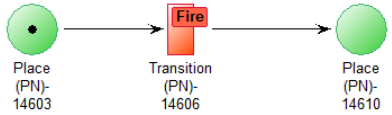
Layout Design

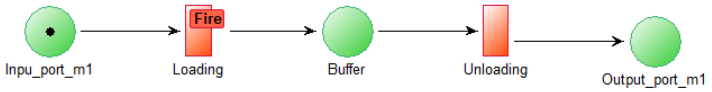


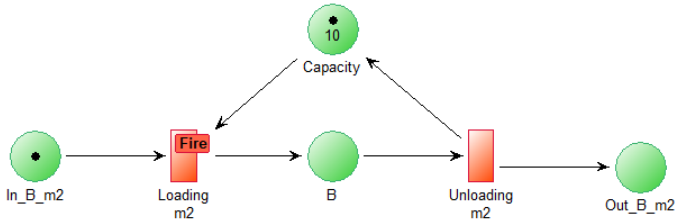
BeeUp /Petri net

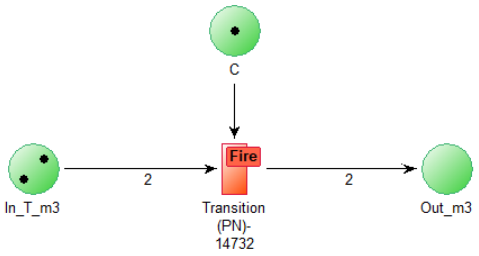


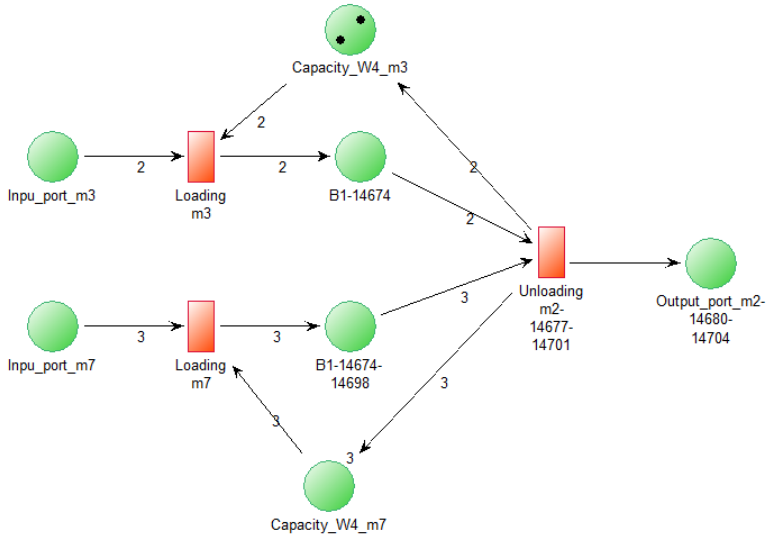
Petri net (BeeUp)

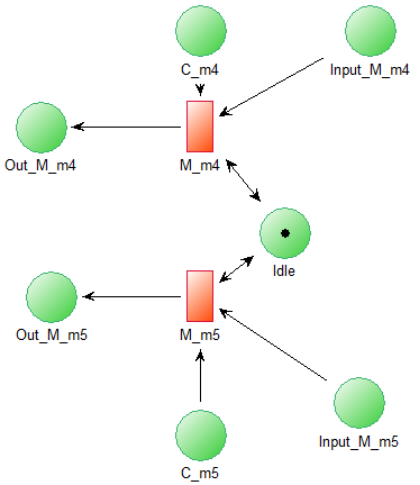




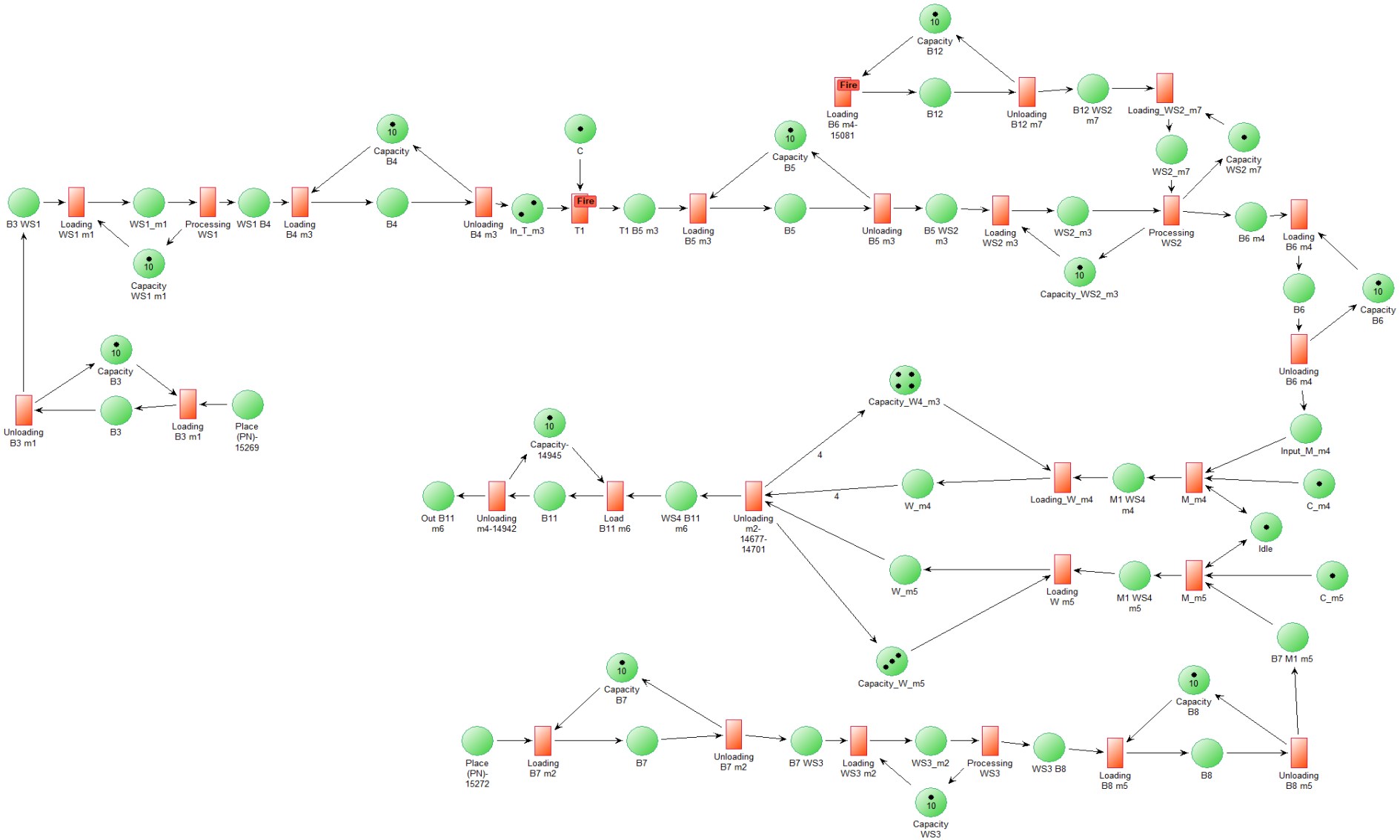




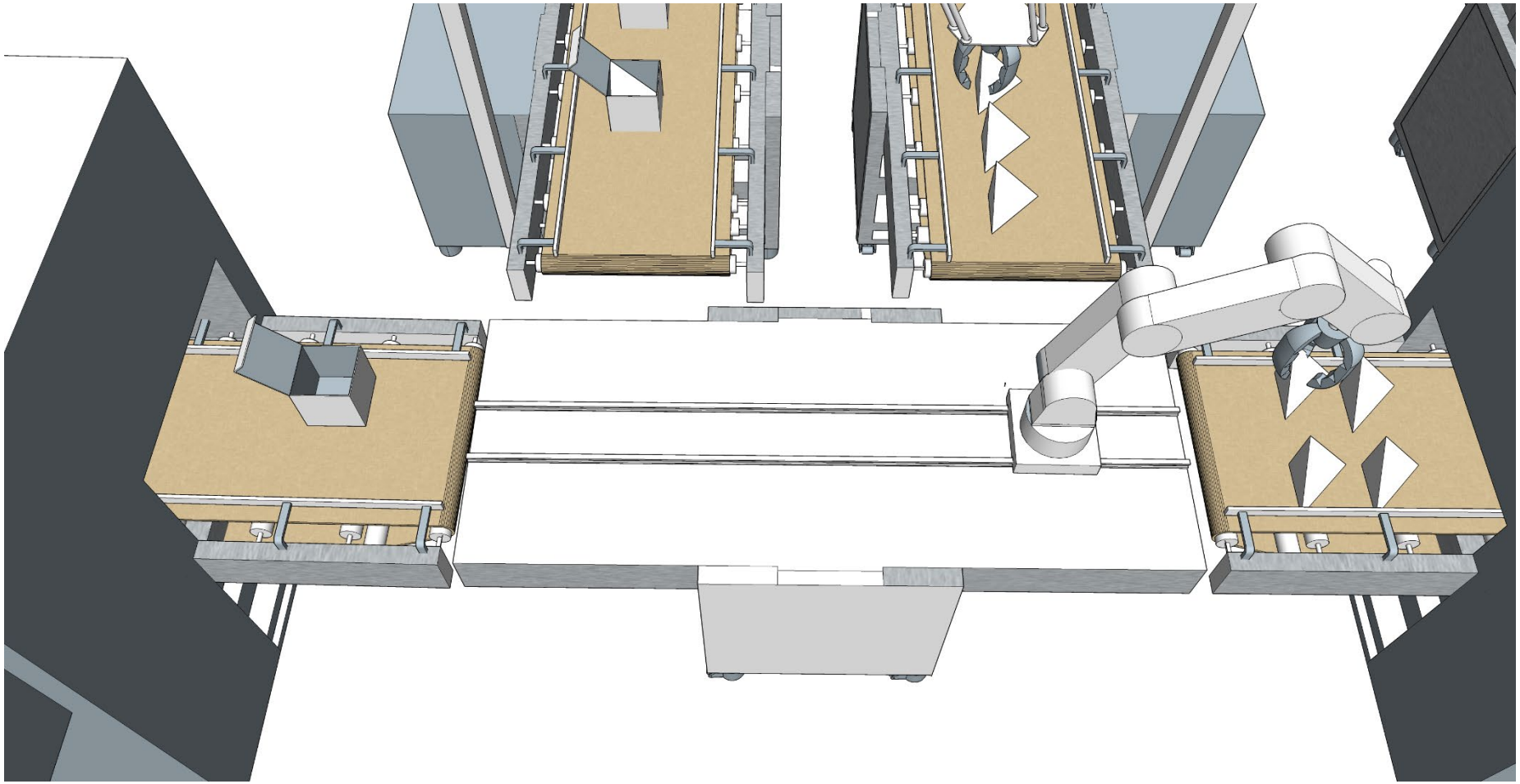




Petri net


B3 WS1
Loading WS1 m1
WS1_m1
Processing WS1
WS1 B4
Loading B4 m3
B4
Capacity B4
Unloading B4 m3
In_T_m3
C
Fire
T1
T1 B5 m3
Loading B5 m3
B5
Capacity B5
Unloading B5 m3
B5 WS2 m3
Loading WS2 m3
WS2_m3
Capacity_WS2_m3
Processing WS2
WS2_m7
Capacity WS2 m7
Loading_WS2_m7
B12 WS2 m7
Unloading B12 m7
B12
Capacity B12
Loading B6 m4-15081
B6 m4
Loading B6 m4
B6
Capacity B6
Unloading B6 m4
Unloading B1 m1
B3
Capacity B3
Loading B3 m1
Place (PN)-15269
Capacity_W4_m3
Capacity-14945
Out B11 m6
Unloading m4-14942
B11
Load B11 m6
WS4 B11 m6
Unloading m2-14677-14701
W_m4
Loading_W_m4
M1 WS4 m4
M_m4
Input_M_m4
C_m4
Idle
W_m5
Loading W m5
M1 WS4 m5
M_m5
C_m5
B7 M1 m5
Capacity_W_m5
Capacity B7
Place (PN)-15272
Loading B7 m2
B7
Unloading B7 m2
B7 WS3
Loading WS3 m2
WS3_m2
Capacity WS3
Processing WS3
WS3 B8
Loading B8 m5
B8
Capacity B8
Unloading B8 m5

Scheduling



Scheduling principles

- Scheduling
  - Time planning the tasks of multiple jobs and mapping them on the available resources
- Types of scheduling
  - Forward scheduling – start with the time when resources become available to determine the due time.
  - Backward scheduling - planning the tasks from the due time or required-by time to determine the start time On- line
  - Off-line
  - Single stage
  - Multistage
- The benefits of production scheduling include:
  - Process change-over reduction
  - Inventory reduction, leveling
  - Reduced scheduling effort
  - Increased production efficiency
  - Work load leveling
  - Accurate due date
  - Real time information

Scheduling

- Optimization problem represented by a triple $\alpha \mid \beta \mid \gamma$
  - α The design of the resources
  - β The running properties and constraints
  - γ The target function to be minimized
- α - only 1 value of the following:
  - *1* There is only one machine
  - P_m There are *m* identical machines that run in parallel
  - F_m Flow shop with *m* machines, every job must be carried out on every machine in the same predetermined succession
  - J_m Job shop with *m* machines Each job must also be executed on every machine, but the succession is predetermined, but not necesary the same for all jobs
  - O_m Open shop with *m* machines. very order must be processed on every machine, but in arbitrary order
- β - Can be empty or to have multiple values. Some examples:
  - *p* job have same processing tine on al machines
  - *d* job have same due dates
  - *pmnt* preemption job can be interrupted
- γ
  - absence of an objective value is denoted by a single dash. This means that the problem consists simply in producing a feasible scheduling, satisfying all given constraints
  - maximisation or minimisation of a combination (weighted sum) of completion time C_j, flow time F_j, Lateness L_j Throughput U_j, Tardiness T_j, Earlisness E_j

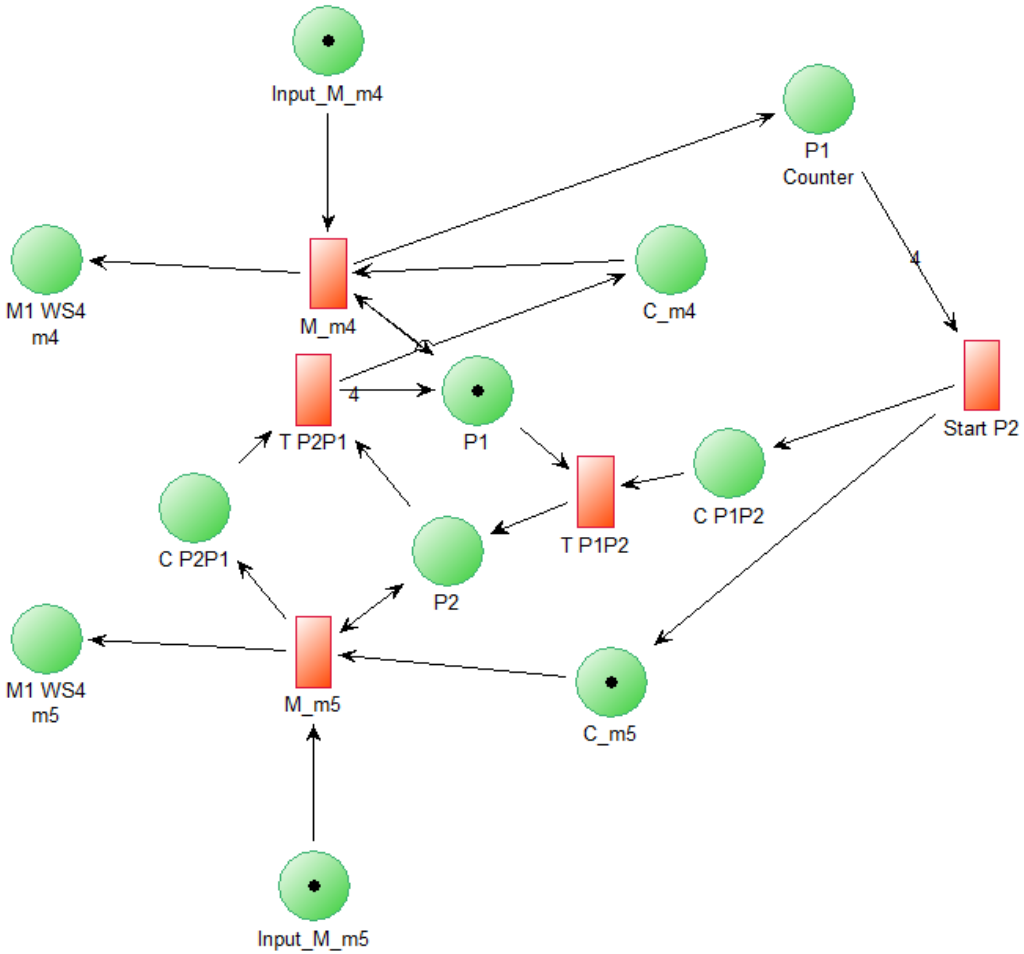
Scheduling

- is an NP problem for $m>2$
- Intensive researched domain
- need intensive mathematical support for solving

But

- modeling can qualitatively help understand and classify the scheduling problem ($\alpha \mid \beta \mid \gamma$)
- simulation can qualitatively and quantitatively validates the results and algorithms

Automation


Input_M_m4
P1
Counter
M1 WS4
m4
M_m4
C_m4
4
1
4
T P2P1
P1
Start P2
C P2P1
T P1P2
C P1P2
P2
M1 WS4
m5
M_m5
C_m5
Input_M_m5

Automation with Petri nets

- Supervisory control - the process of limiting the action of a plant (described as an DES) to a set of safe, allowable or desirable behaviors
- Standard procedure of control design
  - theorem of controller synthesis described in [Moody98]
  - based on incidence matrix and constraint formulation
  - Transition - observable and controllable or not
  - Constraints
    - Logical - transition fires if
    - Non-timed constraints - limit the tokens number in a place
    - Timed constraints – transition fires after *x* seconds
- Deadlock control
  - Deadlock removal by design – deadlock free nets – siphon based methods
  - Deadlock avoidance trough

Summary

Methods and tools for

- Layout design – MSML/Digital Design Planer Petri net/ BeeUp
- Operation design – Scheduling - MSML/Digital Design Planer and ?
- Automation – Control system design - Petri net/ BeeUp