Embedded Systems
INSTITUTE

Development of new warehousing systems and the use of modelling

Roelof Hamberg

Embedded Systems Institute

falcon project

Embedded Systems INSTITUTE

Period: 2006/10 – 2011/09

20 FTE/year, 9 PhDs, 2 PostDocs

VanDerLande INDUSTRIES

TU/e technische universiteit eindhoven

TU Delft Delft University of Technology

University of Twente The Netherlands

Embedded Systems INSTITUTE

DEMCON advanced mechatronics

Goal

Architectures and methods that combine system and component choices for optimal system functionality, i.e. performance & reliability

Issues

- Mapping system onto components
- Consistency in system aspect choices
- Model based engineering
- Increased automation by robotics solutions

architectures: processes, layout, dynamical behaviour, self-optimising control

performance models

relate analysis, design, and implementation

robotic hands & visual guidance

Outline

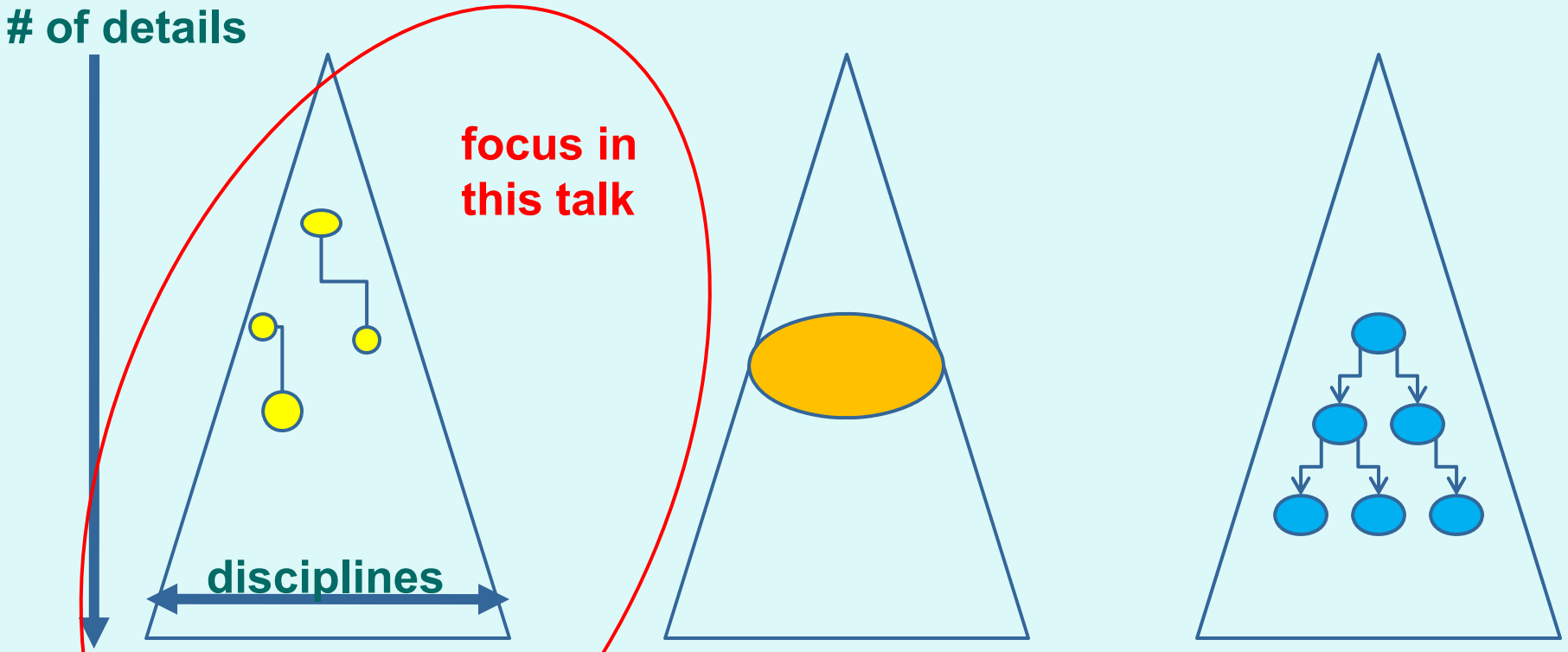
Embedded Systems INSTITUTE

- System design & modeling
  - What, why, cost/benefit, lifecycle
- 3 Cases
  - Simple models, fast results
- Concluding remarks

Model-based design

Embedded Systems INSTITUTE

Three different ways of using models in system design



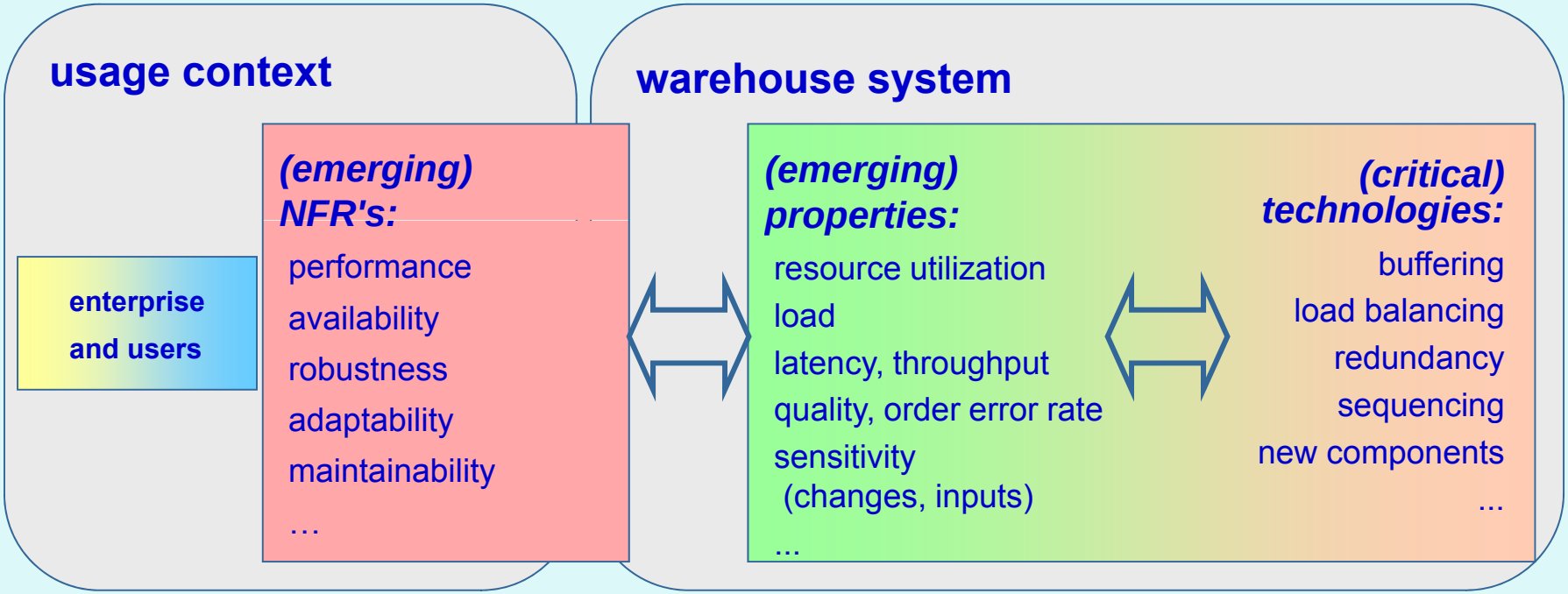
Supportive to critical decision making

Automatic design space exploration

Model-driven: generate design artifacts

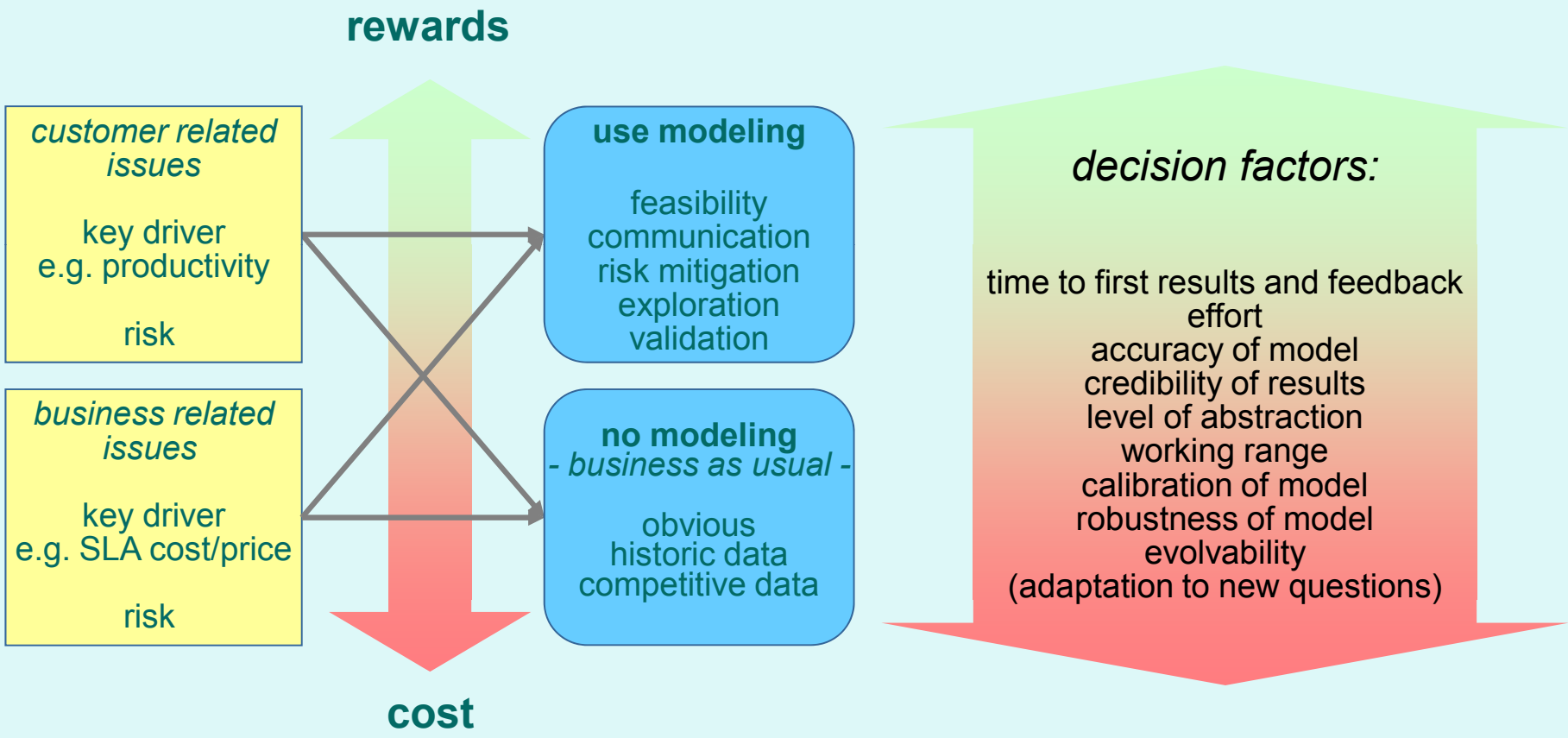
What to model?

Embedded Systems INSTITUTE



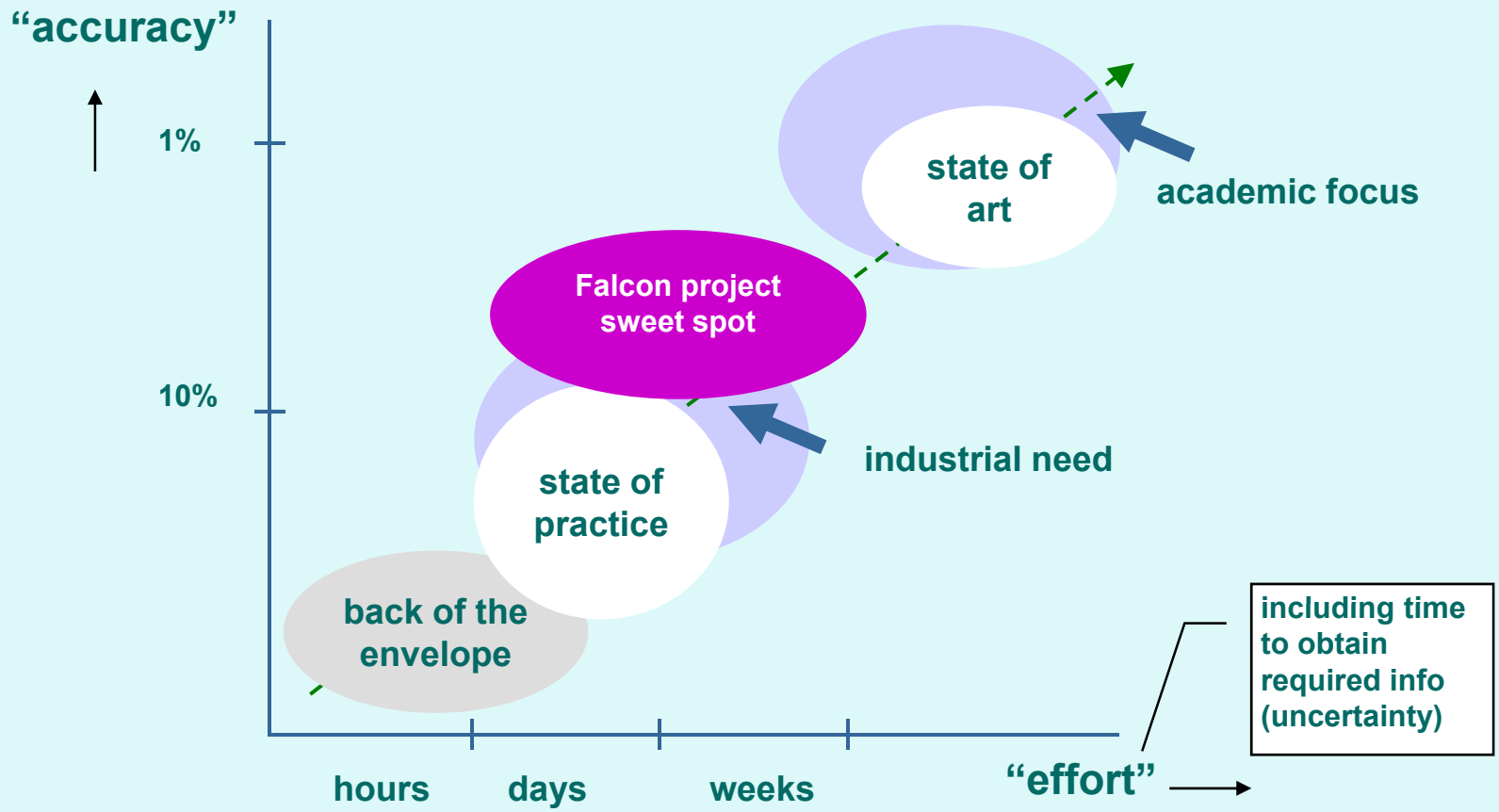
What & why to model?

Embedded Systems INSTITUTE



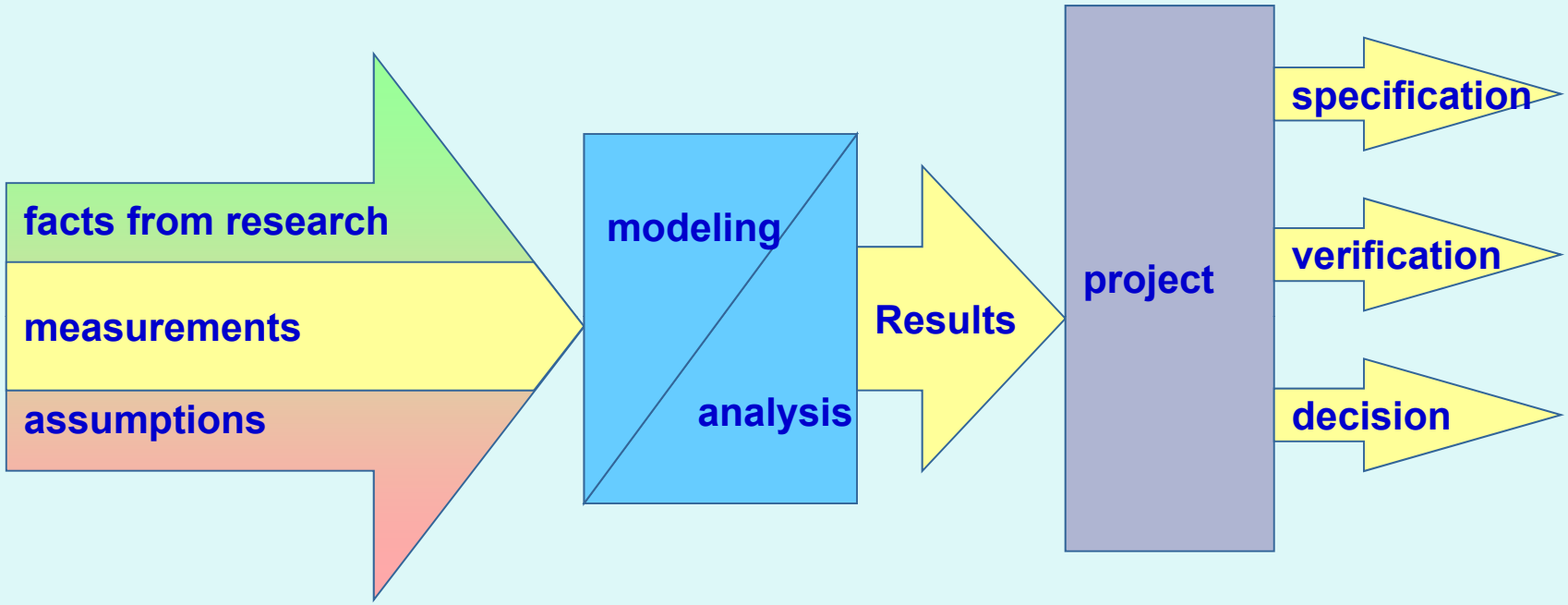
One aspect of model cost/benefit

Embedded Systems INSTITUTE



Purpose of Modeling

Embedded Systems INSTITUTE



uncertainties
unknowns
errors

→

accuracy
working range
credibility

→

risk
customer satisfaction
time, cost, effort
profit margin

Model versus Reality

Embedded Systems INSTITUTE

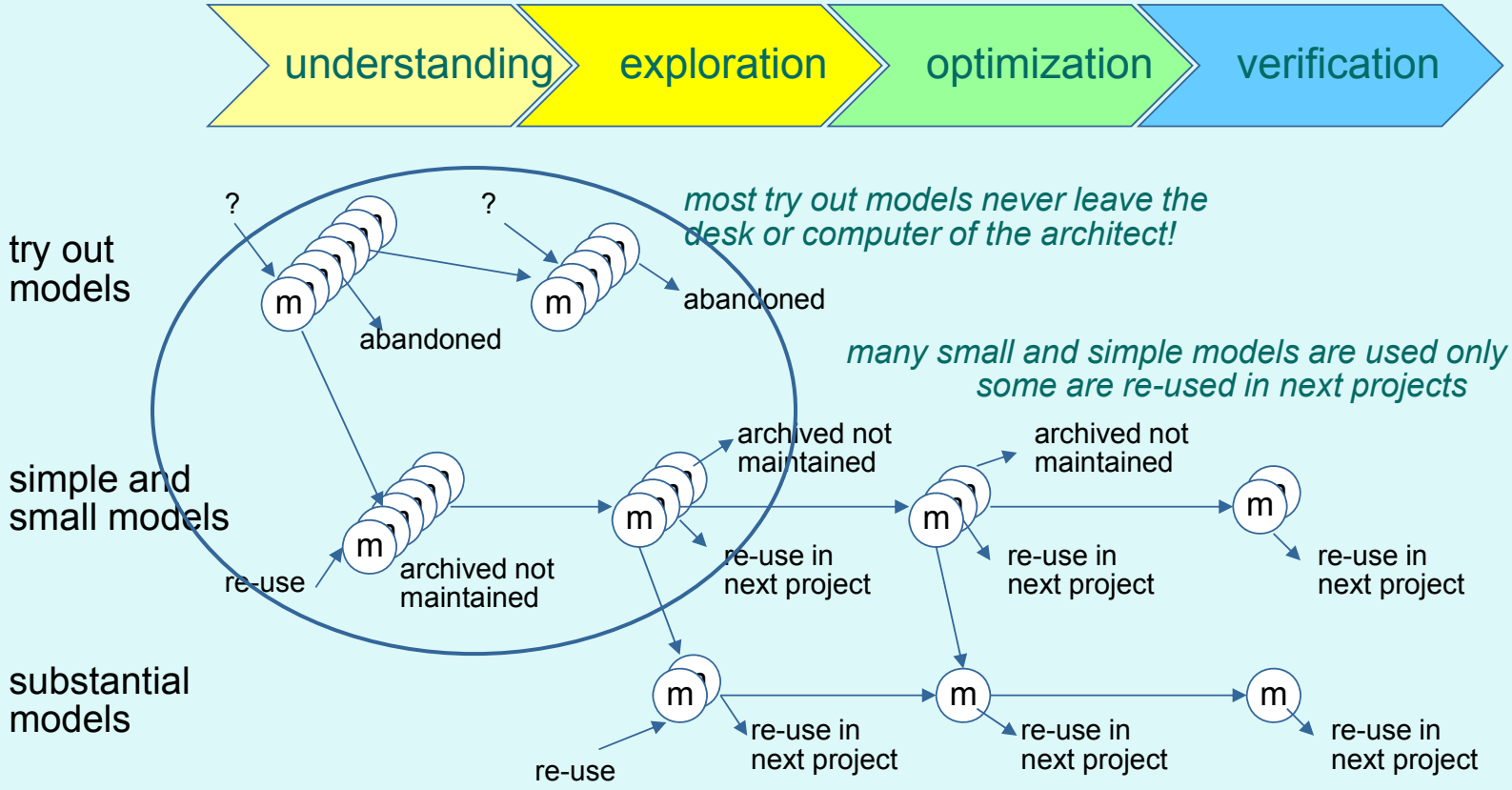
Reality ≠ Model

"all models are wrong, some are useful"

George Box, statistician

Lifecycle of Models

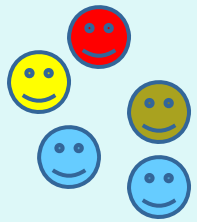
Embedded Systems INSTITUTE



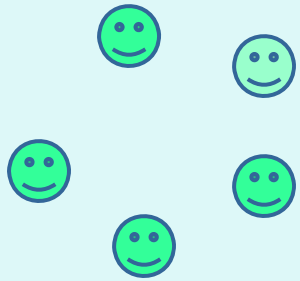
Cases for modeling in Vanderlande

Embedded Systems INSTITUTE

- **Support in a new product development team**
  - New variant of an automated retrieval system …
  - Early phase: understanding & exploration
  - Rapid changes of focus: balancing act
  - *Cases 1 & 2*



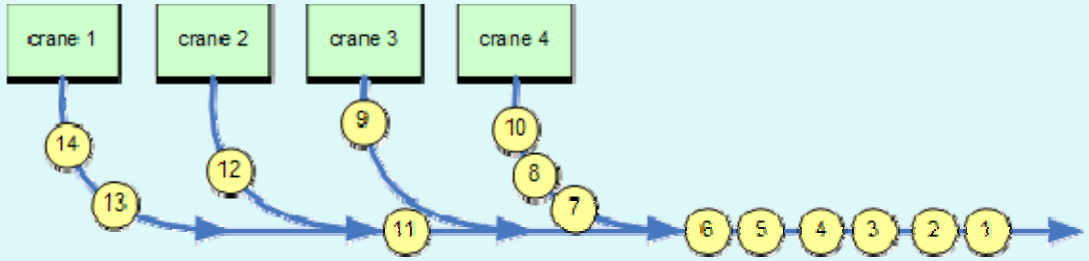
- **Support for sales engineering**
  - Middle phase: exploration & optimization
  - *Case 3*
  - *All modeling activities critically depend on set of people & their backgrounds*



Case 1: in-sequence item retrieval
context

Embedded Systems INSTITUTE

What is the effect on system performance?



Reality	Model
crane performance • dependent on route • dependent on items	**crane performance** • uniform probability distribution around average
items • have identities • have to be handled in sequence • number per job varies	**only consider sets of items** • "black box" sets • # of items sampled (from customer distribution) • "in sequence" is hidden in service
transport • causes different time-delays for different connections	**no transport model** • only interested in throughput
layout • widely different variants	**layout** • variants result in *n→1* mergers

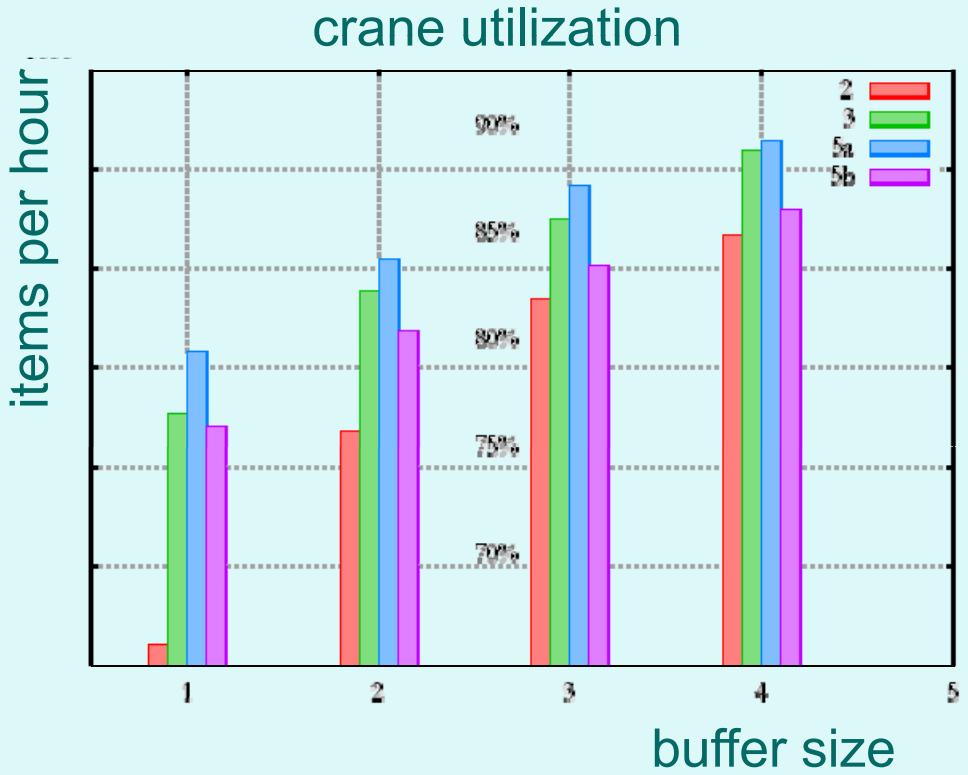
Summary of results

Embedded Systems INSTITUTE

Result

- Model of different variants in 1 day
- First analyses of variants after 2 days

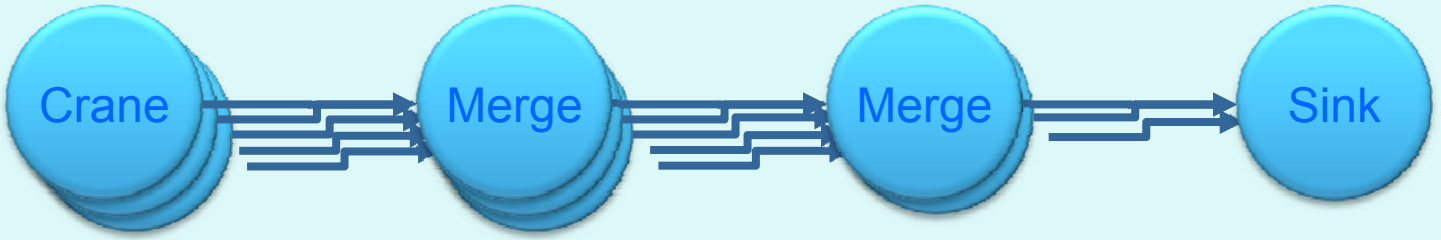
- Observations:
  - discussion about abstraction in buffer-size parameter
  - calibration of model was done w.r.t. other models
  - “we don’t want to know how you did it”



Case 1: in-sequence item retrieval

Model: simulated queuing network

Embedded Systems INSTITUTE




```
while there_is_work do
  p := new(set) setsize(size sample);
  delay EPT * p getsize();
  o!p
od.
```

```
par
  while true do
    [ buffer not full ] i?p;
    buffer put(p)
  od
and
  …
and
  while true do
    [ no buffer empty ] q = ∑ buffers get();
    delay EPT * q getsize();
    o!q
  od
rap.
```

```
while true do
  i?p;
  fileout logto(currentTime,p)
od.
```

Concluding remarks

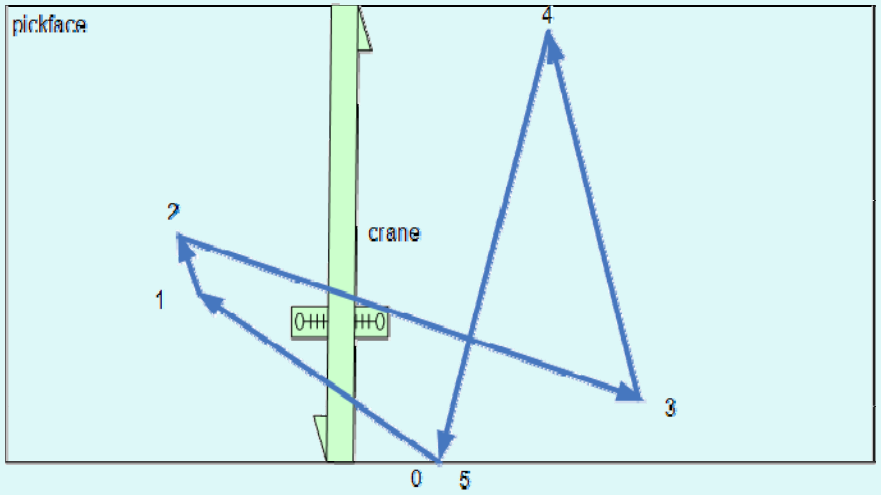
Embedded Systems INSTITUTE

- **Simple simulation (just beyond the static calculus)**
  - Accuracy: mainly relative between different variants
  - Effort: very limited
- **Sufficient to give direction to further development**
  - Answer the relevant design question at that moment
  - Focus on that question only
- **Hurdles for transfer**
  - Specific model abstractions
  - Prerequisite knowledge of tools

Case 2: crane performance
context

Embedded Systems INSTITUTE

What is the effect of picking sequence on crane performance?



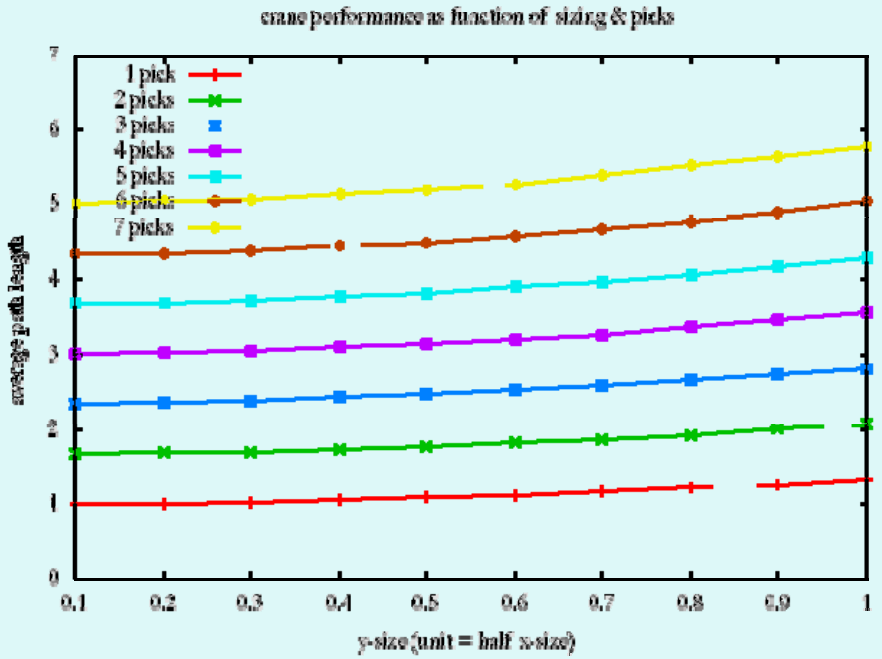
Reality	Model
crane performance, dependent on • route (how many items) • docking & undocking times • acceleration, velocity • loading & unloading times **layout** • height and width • discrete locations	**crane performance** • random route, path length calculation • first: constant velocity second: velocity & acceleration **layout** • height relative to width • continuous locations

Summary of results

Embedded Systems INSTITUTE

- Results:
  - First model after 2 hours
  - Second model after 1 day

- Observations:
  - Penalty on path length for low number of picks is very limited
  - Calibrated with Excel sheet with defined path calculation



Case 2: crane performance

Model: script involving rand()

Embedded Systems INSTITUTE

- Model in gawk
  (but anything will do)

```
BEGIN {
  ymax = 1
  x0 = 0
  y0 = 0
  steps = 6
  samples = 10000
}

function cycle(st) {
  x1,y1 = x0,y0
  t = 0
  for ( i=0 ; i<st ; i++ ) {
    if ( i<st-1 ) x2,y2 = 2*rand() – 1, ymax*rand()
      else x2,y2 = x0, y0
    t += max(abs(y2-y1),abs(x2-x1))
    x1,y1 = x2,y2
  }
  return t
}

END {
  srand()
  T = 0
  for ( n=0 ; n<samples ; n++ ) T += cycle(steps)
  printf("%f ",T/samples)
}
```

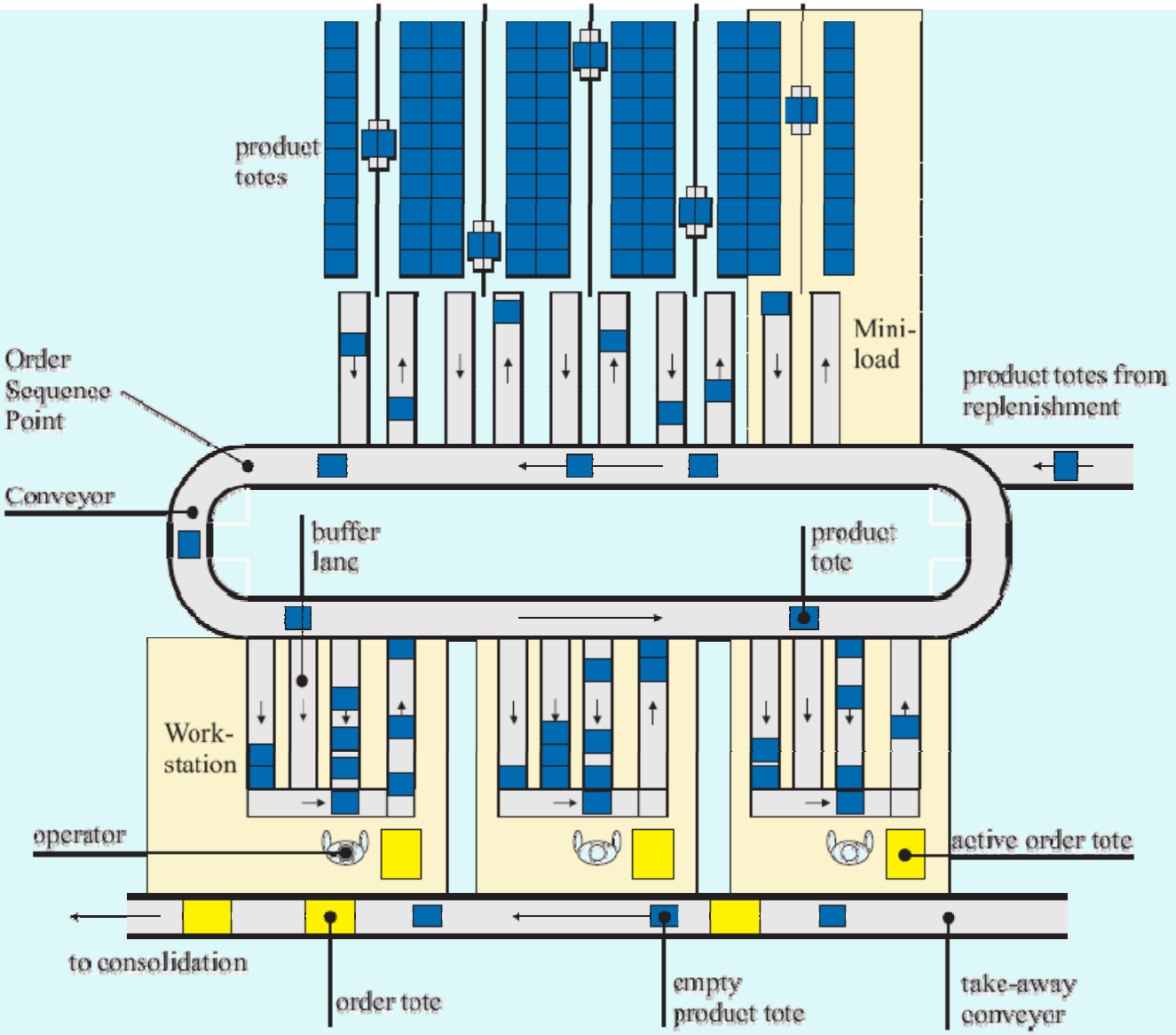
Concluding remarks

Embedded Systems INSTITUTE

- **Simple simulation (just beyond the static calculus)**
  - Accuracy: good?
  - Effort: very limited
- **Sufficient to address uncertainty**
  - Answer the relevant design question at that moment
  - Only answer that question
- **Hurdles for transfer**
  - Too much focus on reusability
  - Build it in Excel…

Case 3: compact pick system sizing
sketch of the system

Embedded Systems INSTITUTE

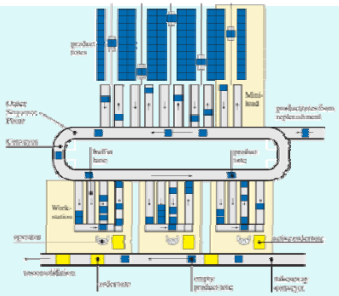


Case 3: compact pick system sizing

Context

Embedded Systems INSTITUTE

What is the effect of configuration on system performance, buffer utilization, and component utilization?



Reality	Model
ASRS • number, batching, performance statistics • dynamic stock **items** • number per job varies, distribution varies **Transport** • different time-delays for different connections • buffer function **workstation** • number, buffer spaces, performance statistics • # active orders, # open orders • type of buffer **control** • priority rules (job age, sequence, pipeline) • smart order planning	**ASRS** • performance with prob. distrib., batching • no stock allocation **items** • # per job sampled from customer distribution **transport** • Δt ~ connection index • parameterized buffer function **workstation** • performance with distribution, buffer size • # active orders, # open orders • FIFO versus random access **control** • priority rules as in real control • virtually no order planning

Approach

Embedded Systems INSTITUTE

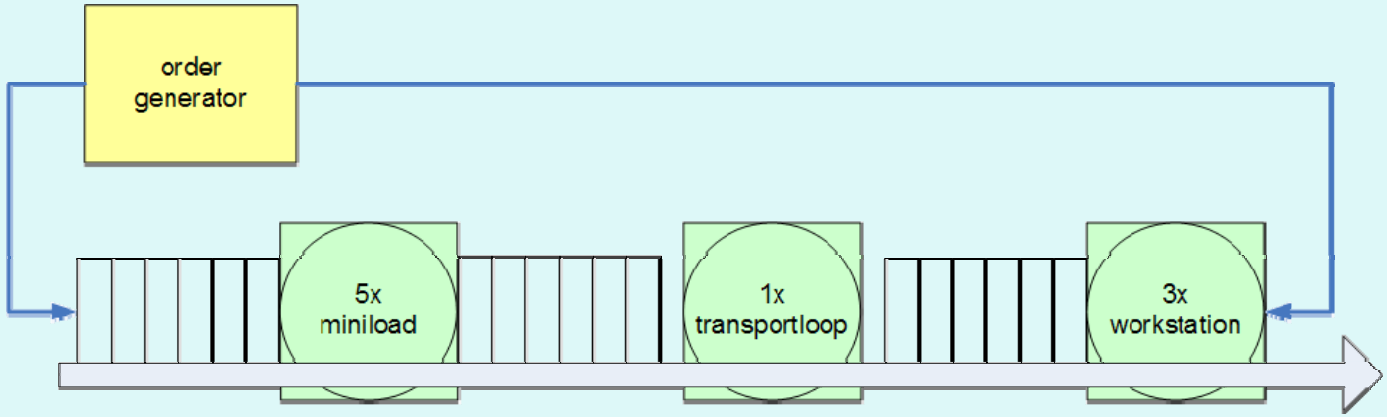
- **Target group**
  - Sales engineers (quick answers to their questions)
  - System engineers (tool support to provide answers)
- **“Simulation-in-a-box” approach**
  - standard components
  - external input parameters
  - sampled order size distribution
  - self-monitored statistics (run-till-statistics-are-met)
  - retrospect visualization

Model Structure

Embedded Systems INSTITUTE

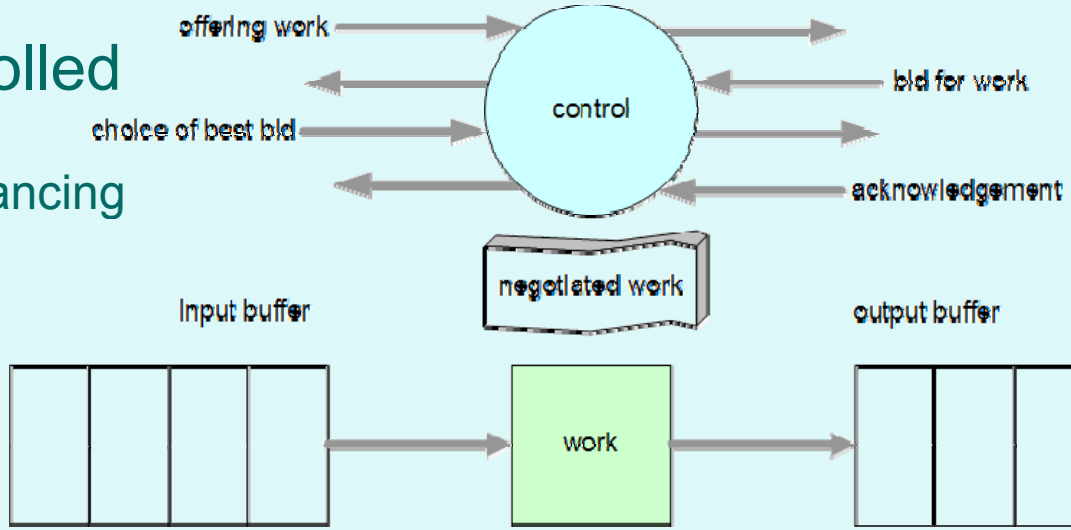
- Basic queuing network

  orders are stock as well as work to be done



- explicitly controlled

  negotiation in order to achieve load balancing



Model input parameters

Embedded Systems INSTITUTE

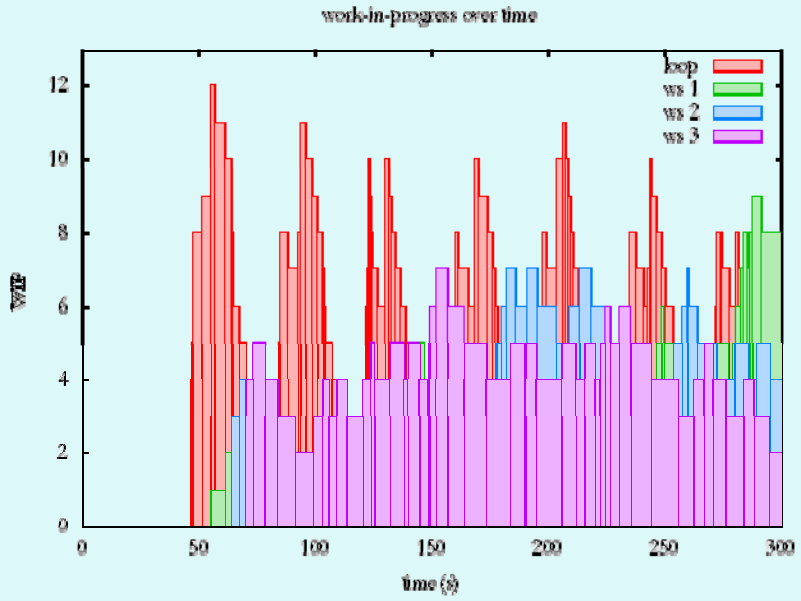
For each component: (they all are identical)

- component multiplicities
- input & output buffer sizes
- averages and spreads of triangularly distributed process times
- batch sizes
- type of input buffer (FIFO or SEQ)
- # open orders, # active orders
- process time dependency on connection
- availabilities & repair times
- priority rule (piece of code, not parameterized for now)

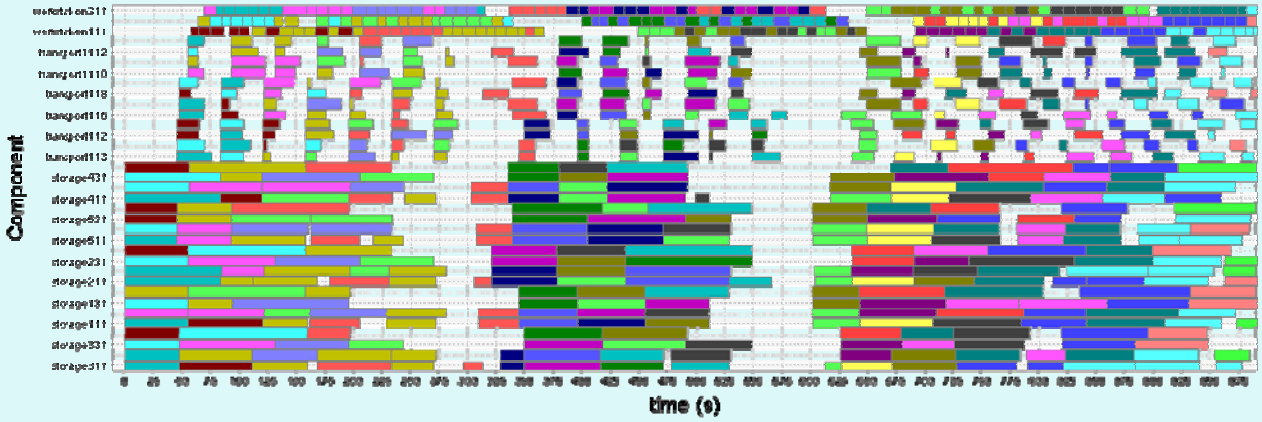
Example graphs

Embedded Systems INSTITUTE

- Detailed log-files to study and validate system behavior over time



- Gantt charts representing tasks carried out by different system components



Results & concluding remarks

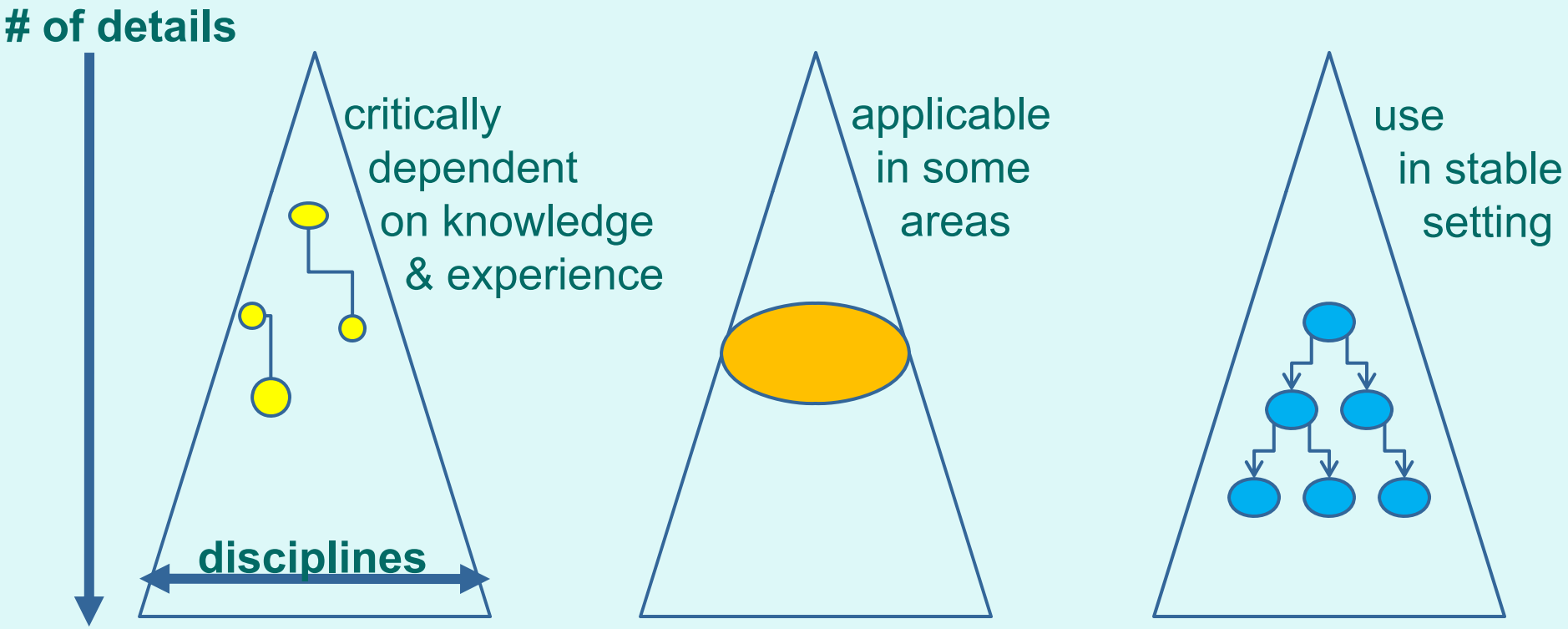
Embedded Systems INSTITUTE

- **Approximately 3 weeks of work**
  - Validation has to be completed
  - Easy-to-use front-end has to be connected
- **For sales engineers**
  - Averages, maxima of throughput, work-in-progress, flow-time
  - For every component separately
  - For nearly all occurring system variations
- **For system engineers**
  - Detailed analyses
- **Hurdles for transfer**
  - Prerequisite knowledge of tools

Concluding remarks

Embedded Systems INSTITUTE

Three different ways of using models in system design



Supportive to critical decision making

Automatic design space exploration

Model-driven: generate design artifacts