# G54SIM (Spring 2016)

## Lecture 04

## Simulation Methods: Discrete Event Simulation

Peer-Olaf Siebers

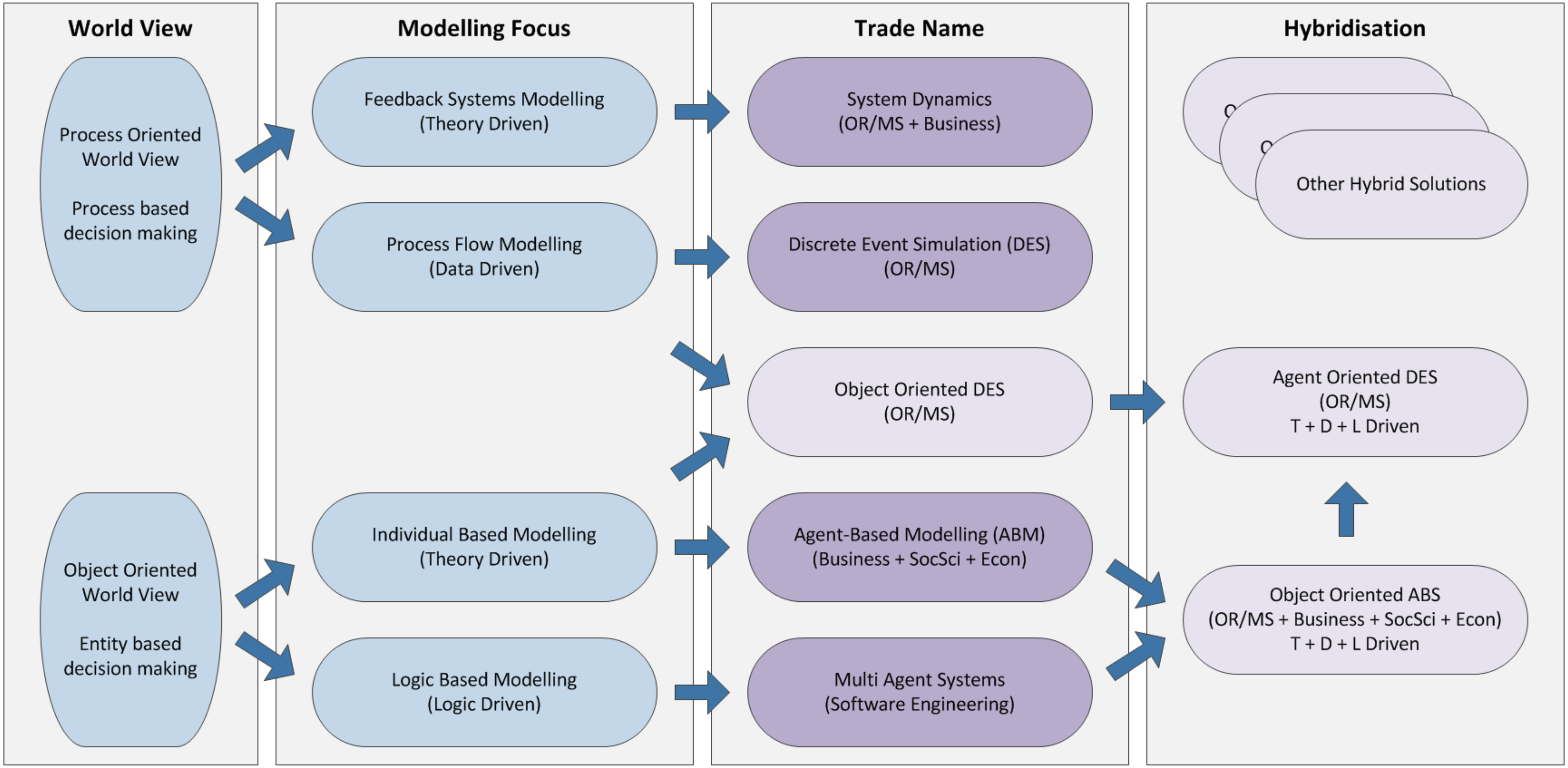
pos@cs.nott.ac.uk

The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

# Motivation

- Introduce Discrete Event Modelling (DEM)
- Introduce the Discrete Event Simulation (DES) execution cycle
- Show how DES can be applied in real world projects

# Simulation Modelling Framework




Theory Driven: Theories for model formulation; data for model validation
Data Driven: Data for model formulation (can be quantitative and qualitative); data for model validation
Logic Driven: Logic for model formulation; data for model validation

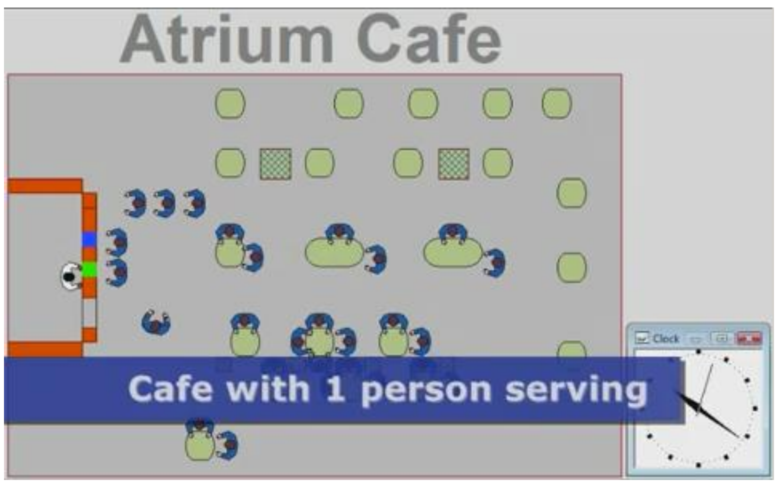
# Simulation Modelling Framework

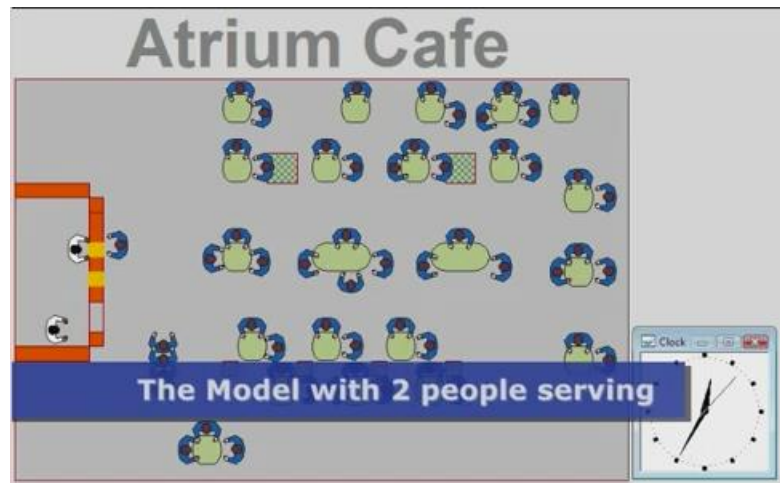
- Discrete Event Simulation
  - Study of queuing systems
  - Entities are routed through the system
  - Process: Organised in terms of queues and flows

- Object Oriented DES Modelling
  - Study of queuing systems
  - Entity templates defined as classes
  - Entities are passive object (they do not initiate any action)
  - Simple decisions are made within the entities
  - Process: Organised in terms of queues and flows

# Discrete Event Simulation Demo

- Coffee Shop Discrete Event Simulation Demo (Witness)



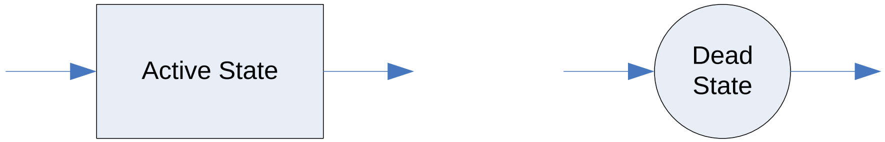



# Discrete Event Modelling

- Terminology:
  - Objects of the system
    - **Entities:** Individual system elements whose behaviour is explicitly tracked; organised in classes and sets; distinguishable by attributes
      - **Classes:** Permanent groups of identical or similar entities (e.g. bus passengers)
      - **Sets:** Temporary groups of identical or similar entities (e.g. passengers on a particular bus, passengers waiting in a queue)
      - **Attributes:** Items of information to distinguish between members of a class (e.g. index) or to control the behaviour of an entity (e.g. entity type)
    - **Resources:** Individual system elements but not modelled individually; treated as countable items (e.g. number of passengers waiting at a bus stop)

# Discrete Event Modelling

- Terminology (cont.)
  - Operations of entities
    - Over time entities co-operate and hence change state
      - **Event:** Instance of time in which a significant state change occurs
      - **Activity:** Operations which are initiated at an event, transforming the state of the entities
    - Entity states:
      - **Active state:** Involves the co-operation of different classes of entities; duration can be determined in advance, usually by taking a sample from an appropriate probability distribution if the simulation is stochastic
      - **Dead state:** No co-operation, entity waits for something to happen; duration cannot be determined in advance



# Discrete Event Modelling

- Hands-On Example:
  - The plot (Pidd, 1998)
    - A theatre booking clerk is employed to sell tickets and answer enquiries. Enquiries can come from someone at the box office or someone phoning the theatre.
    - The clerk is instructed to give priority to the personal customers. Customer and phone calls queue on a FIFO basis. Phone callers never hang up!
  - Classes:
    - Personal customers
    - Phone customers

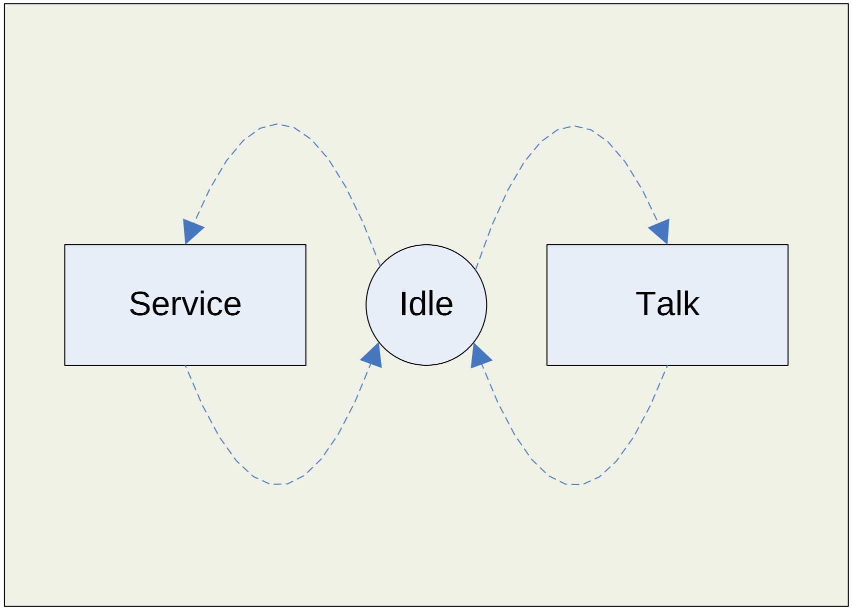
# Discrete Event Modelling

- Hands-On Example:
  - Sets:
    - Personal customers queuing
    - Phone customers queuing
  - Attributes:
    - Type of customer
  - Resources:
    - Booking clerk


The University of Nottingham
UNITED KINGDOM · CHINA · MALAYSIA

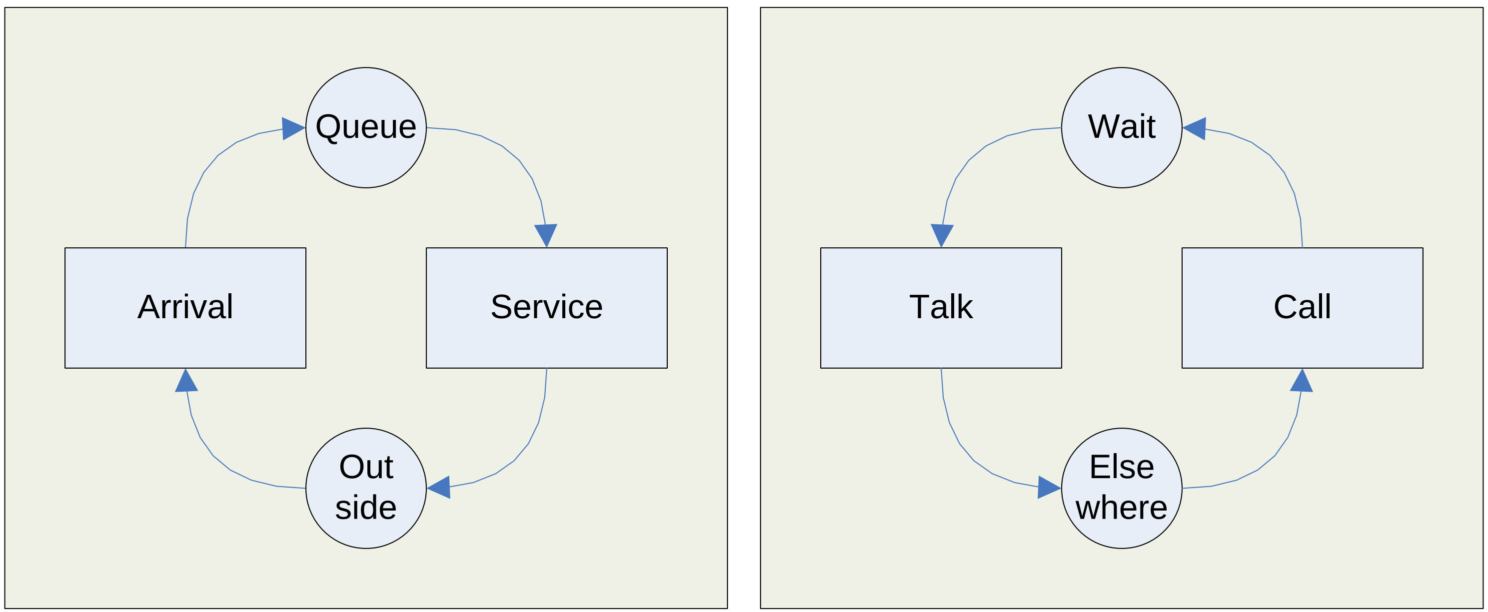
# Discrete Event Modelling

- Hands-On Example (cont.):
  - Activity Cycle Diagrams for booking clerk



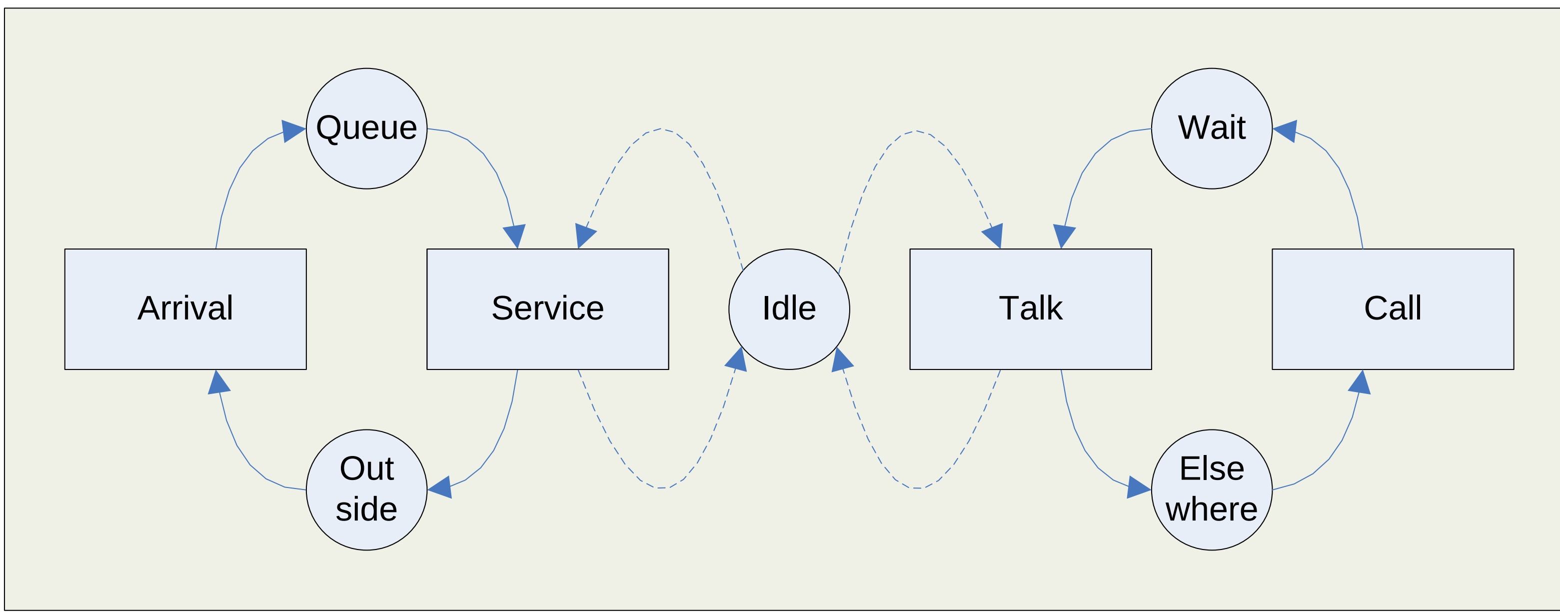
# Discrete Event Modelling

- Hands-On Example (cont.)
  - Activity Cycle Diagrams for personal enquirers and phone callers



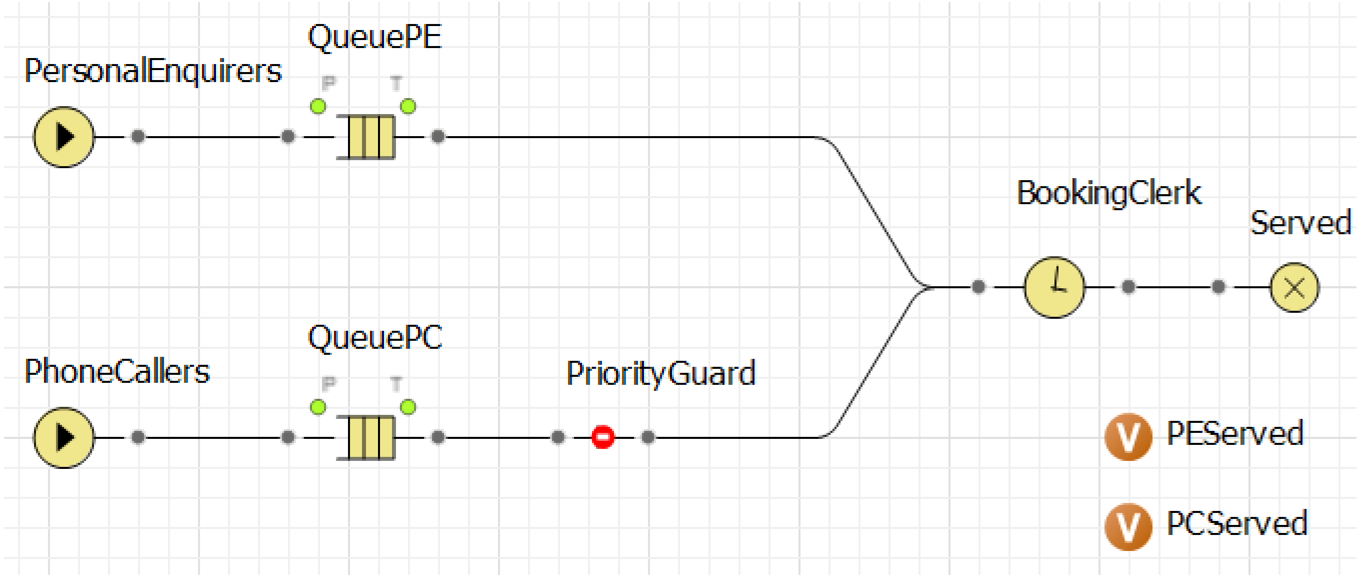
# Discrete Event Modelling

- Hands-On Example (cont.)
  - Activity Cycle Diagrams for ticket sales and enquiries



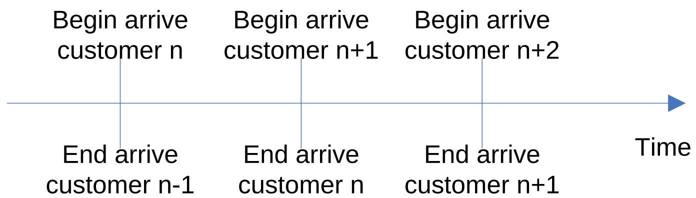
# Discrete Event Modelling

- Hands-On Example (cont.)
  - Process Flow Diagrams for ticket sales and enquiries

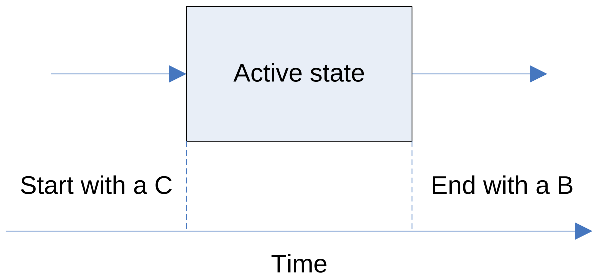


# Discrete Event Simulation (DES)

- Three-Phase Approach (first described by Tocker in 1963)
  - In this simulation approach events are classified into two types
    - **B (bound or booked) Events:** State changes that are scheduled to occur at a point in time. In general B events relate to arrivals or the completion of an activity.
    - **C (conditional) Events:** State changes that are dependent on the conditions in the model. In general C events relate to the start of some activity



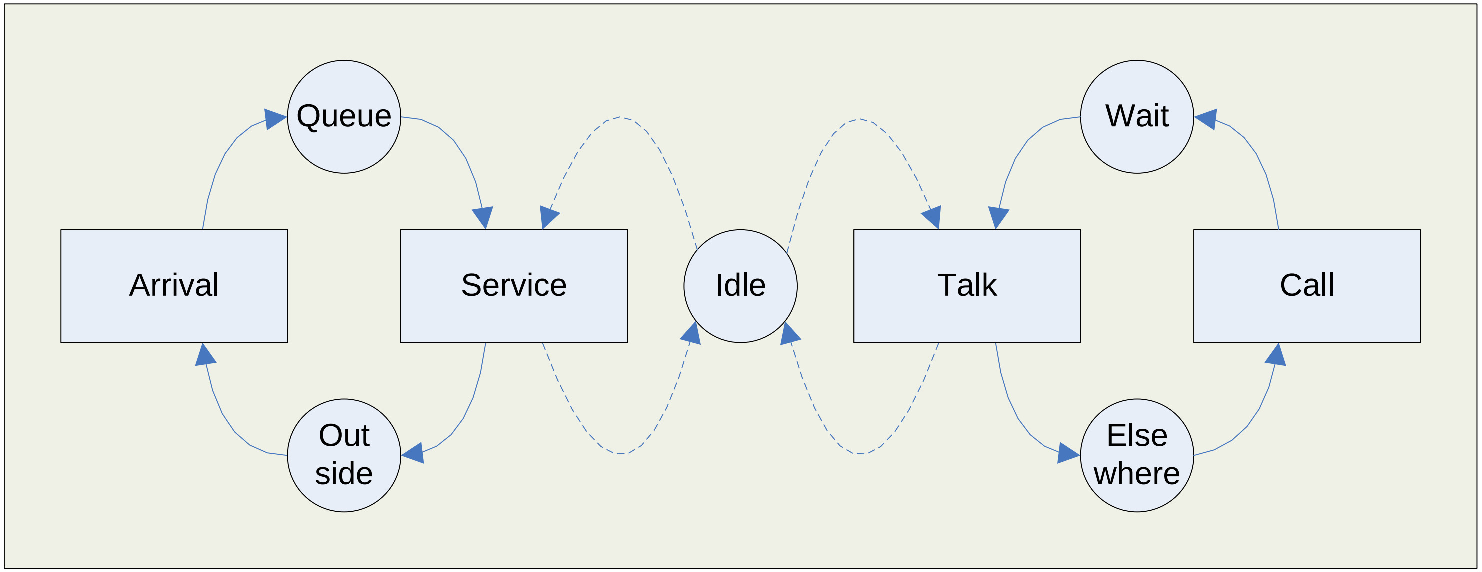

Arrival process




Activity

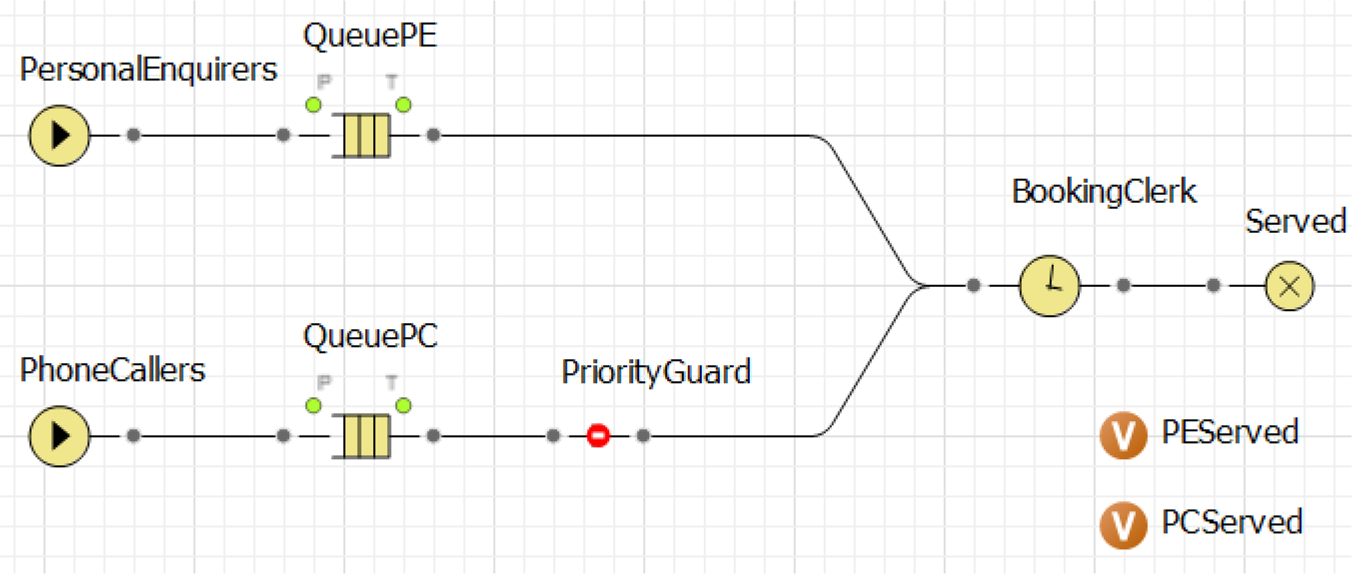
# Discrete Event Simulation

- Three-Phase Approach
  - Booking Clerk Bs and Cs from Activity Cycle Diagram
    - B1: Arrive ... B2: EndOfService ... B3: Call ... B4: EndOfTalk
    - C1: BeginService ... C2: BeginTalk

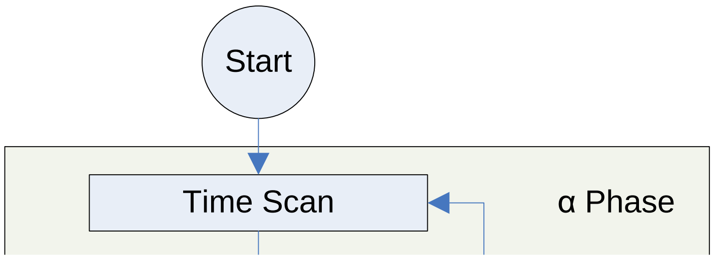


# Discrete Event Simulation

- Three-Phase Approach
  - Booking Clerk Bs and Cs from Process Flow Diagram
    - B1: ArrivePE ... B2: EndOfService ... B3: ArrivePC ... B4: EndOfTalk
    - C1: BeginService ... C2: BeginTalk



# Discrete Event Simulation




- **α:** Find out when the next event is due, move simulation clock to that time, put all entities due to engage in a B at that time into the *DueNow* list
- **β:** Execute activities of entities in the *DueNow* list
- **γ:** Executive must attempt each C in turn by checking if the condition in the test heads are satisfied

# Discrete Event Simulation

- Required information about each entity
  - **Time cell:** Time when entity is next due to change state, if this is known; only meaningful if entity is committed to some B in the future
  - **Availability:** Boolean field showing whether the entity is committed to some B in the future; if **TRUE** entity is uncommitted and its time cell is meaningless; if **FALSE** time cell indicates when entity will next change state
  - **Next activity:** Only meaningful if the availability is FALSE and it indicates the B in which the entity is due to engage at the time shown by the time cell
- Reminder
  - B1: Arrive ... B2: EndOfService ... B3: Call ... B4: EndOfTalk;
  - C1: BeginService ... C2: BeginTalk

# Discrete Event Simulation

- Three-Phase Approach
  - **Initialisation:** Clerk is idle; first personal enquirer due to arrive at time 4 and first phone call due to arrive at time 6; time is 0; all queues are empty; no personal enquirers or phone calls have arrived
  - **Random number stream:** 4,6,5,5,3,3,6,4,8

**End of Init.:** Clock=0; Queue=0; Wait=0; PersIn=0; PhoneIn=0; DueNow=/

| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | 4 | FALSE | Personal Arrival |
| (2) Phone call arrival machine | 6 | FALSE | Phone Call |
| (3) Clerk | 0 | TRUE | |

# Discrete Event Simulation

- Three-Phase Approach
  - **First α-Phase:** Find out when the next event is due, move simulation clock to that time, put all entities due to engage in a B at that time into the *DueNow* list (at t=4 entity 1 is due to arrive)
  - **Random number stream:** 4,6,5,5,3,3,6,4,8

**End of α:** Clock=4; Queue=0; Wait=0; PersIn=0; PhoneIn=0; DueNow=1

| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | 4 | FALSE | Personal Arrival |
| (2) Phone call arrival machine | 6 | FALSE | Phone Call |
| (3) Clerk | 0 | TRUE | |

# Discrete Event Simulation

- Three-Phase Approach
  - **First β-Phase:** Execute activities of entities in the *DueNow* list; remember that the service does not start in the β-Phase (brings first persEnq into the system and schedules next persEnq (to arrive 5 min later); entity is put in queue and counter for persEnq is increased)
  - **Random number stream:** 4,6,5,5,3,3,6,4,8

**End of β:** Clock=4; Queue=1; Wait=0; PersIn=1; PhoneIn=0; DueNow=/

| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | 9 | FALSE | Personal Arrival |
| (2) Phone call arrival machine | 6 | FALSE | Phone Call |
| (3) Clerk | 0 | TRUE | |

# Discrete Event Simulation

- Three-Phase Approach
  - **First γ-Phase:** Executive must attempt each C in turn by checking if the condition in the test heads are satisfied (beginServ requires persEnq in queue and clerk to be idle; fulfilled; service takes 5 min)
  - **Random number stream:** 4,6,5,5,3,3,6,4,8

**End of γ:** Clock=4; Queue=0; Wait=0; PersIn=1; PhoneIn=0; DueNow=/

| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | 9 | FALSE | Personal Arrival |
| (2) Phone call arrival machine | 6 | FALSE | Phone Call |
| (3) Clerk | 9 | FALSE | EndService |

# Discrete Event Simulation

- Three-Phase Approach
  - **Second α-Phase:** Find out when the next event is due, move simulation clock to that time, put all entities due to engage in a B at that time into the *DueNow* list
  - **Random number stream:** 4,6,5,5,3,3,6,4,8

**End of α:** Clock=6; Queue=0; Wait=0; PersIn=1; PhoneIn=0; DueNow=2

| Entity | Time cell | Availability | Next Activity |
| --- | --- | --- | --- |
| (1) Personal enquirer arrival machine | 9 | FALSE | Personal Arrival |
| (2) Phone call arrival machine | 6 | FALSE | Phone Call |
| (3) Clerk | 9 | FALSE | EndService |

# Discrete Event Simulation

- Three-Phase Approach
  - **Second β-Phase:** Execute activities of entities in the *DueNow* list; remember that the service does not start in the β-Phase
  - **Random number stream:** 4,6,5,5,3,3,6,4,8

**End of β:** Clock=6; Queue=0; Wait=1; PersIn=1; PhoneIn=1; DueNow=/

| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | 9 | FALSE | Personal Arrival |
| (2) Phone call arrival machine | 9 | FALSE | Phone Call |
| (3) Clerk | 9 | FALSE | EndService |

# Discrete Event Simulation

- Three-Phase Approach
  - **Second γ-Phase:** Executive must attempt each C in turn by checking if the condition in the test heads are satisfied
  - **Random number stream:** 4,6,5,5,3,3,6,4,8

**End of γ:** Clock=6; Queue=0; Wait=1; PersIn=1; PhoneIn=1; DueNow=/

| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | 9 | FALSE | Personal Arrival |
| (2) Phone call arrival machine | 9 | FALSE | Phone Call |
| (3) Clerk | 9 | FALSE | EndService |

# Discrete Event Simulation

- Three-Phase Approach
  - Third Round?

# Discrete Event Simulation

- Three-Phase Approach

**End of α3:** Clock=9; Queue= ; Wait= ; PersIn= ; PhoneIn= ; DueNow=1,2,3

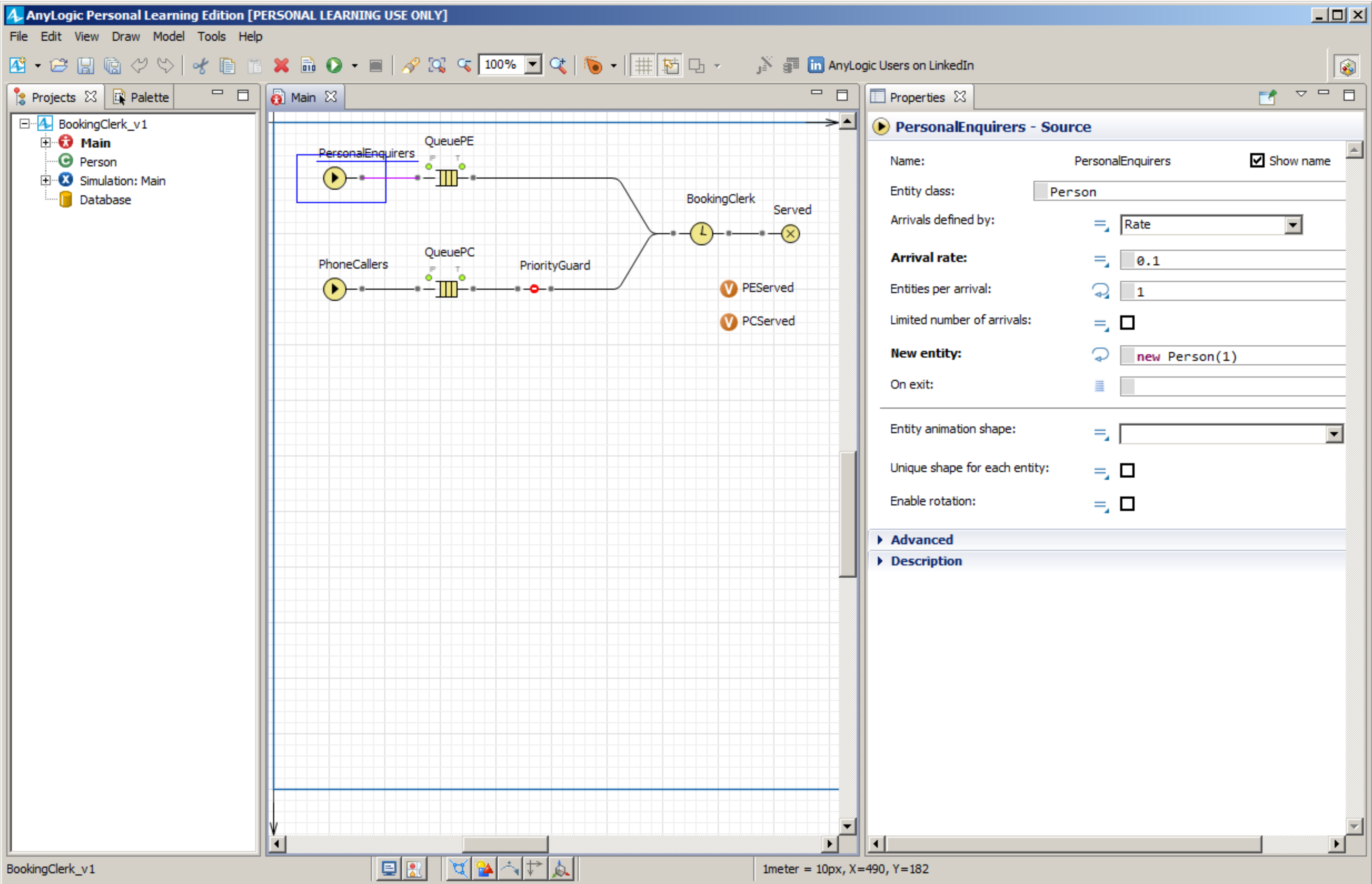
| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | | | Personal Arrival |
| (2) Phone call arrival machine | | | Phone Call |
| (3) Clerk | | | EndService |

**End of β3:** Clock= ; Queue=1; Wait=2; PersIn=2; PhoneIn=2; DueNow=/

| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | 12 | | Personal Arrival |
| (2) Phone call arrival machine | 15 | | Phone Call |
| (3) Clerk | 9 | TRUE | - |

**End of γ3:** Clock= ; Queue=0; Wait=2; PersIn=2; PhoneIn=2; DueNow=/

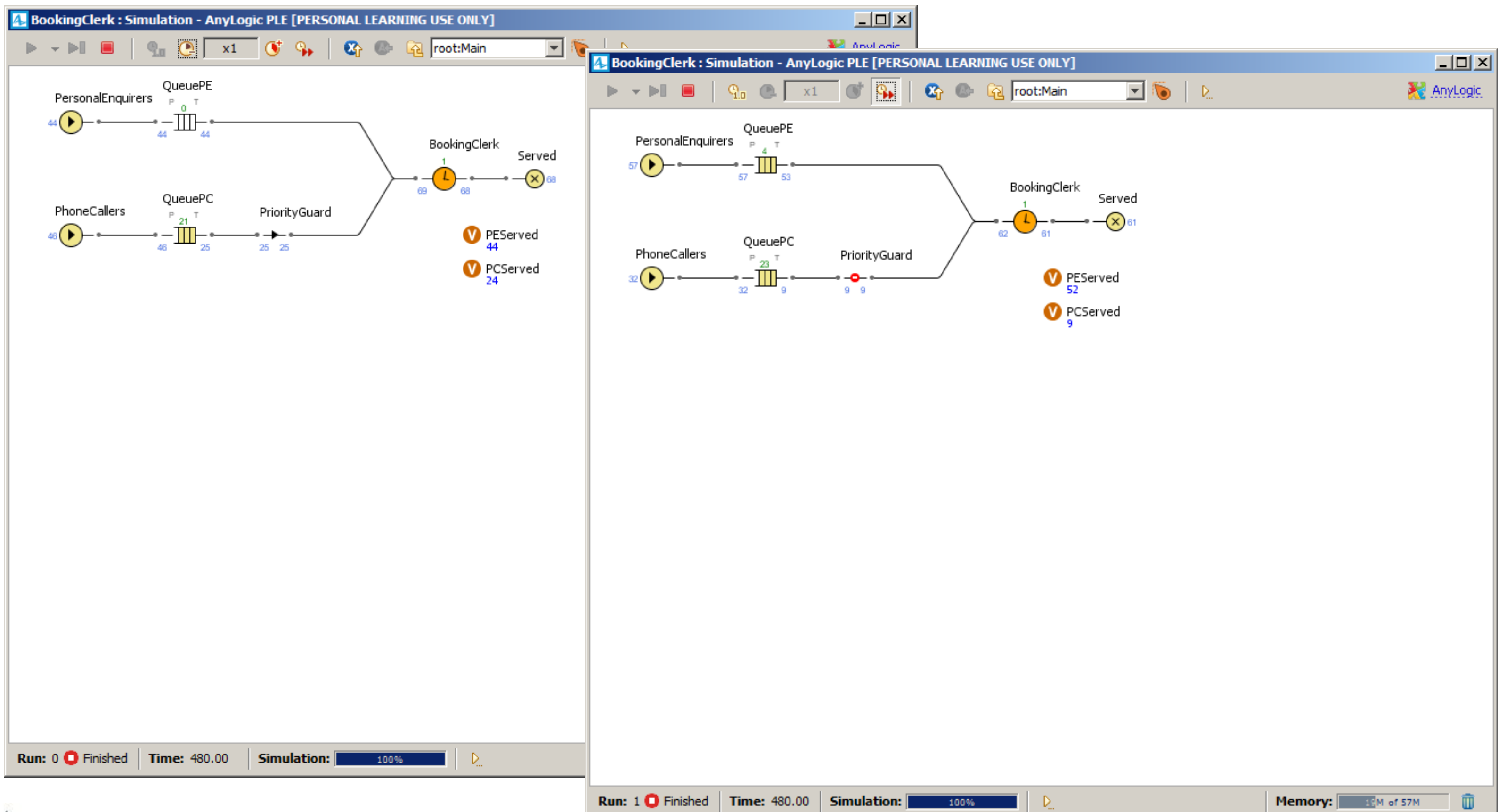
| Entity | Time cell | Availability | Next Activity |
|---|---|---|---|
| (1) Personal enquirer arrival machine | | | Personal Arrival |
| (2) Phone call arrival machine | | | Phone Call |
| (3) Clerk | 13 | FALSE | EndService |



AnyLogic Personal Learning Edition [PERSONAL LEARNING USE ONLY]
File Edit View Draw Model Tools Help
AnyLogic Users on LinkedIn
Projects
Palette
BookingClerk_v1
Main
Person
Simulation: Main
Database
Main
PersonalEnquirers
QueuePE
PhoneCallers
QueuePC
PriorityGuard
BookingClerk
Served
PEServed
PCServed
Properties
PersonalEnquirers - Source
Name: PersonalEnquirers
Show name
Entity class: Person
Arrivals defined by: Rate
Arrival rate: 0.1
Entities per arrival: 1
Limited number of arrivals:
New entity: new Person(1)
On exit:
Entity animation shape:
Unique shape for each entity:
Enable rotation:
Advanced
Description
BookingClerk_v1
1meter = 10px, X=490, Y=182

# Discrete Event Simulation

- Simulation Output (one booking clerk)



# Output Analysis

- What can statistics do?
  - Statistical estimation
    - Estimate population mean from sample mean
  - Statistical hypothesis testing
    - Decide whether observed difference is likely to be caused by chance
  - Statistical modelling
    - Test how well experimental data fits a model

# Output Analysis

- Single Scenario Analysis:
  - Calculate point estimate
    - Mean
  - Calculate interval estimator
    - Confidence interval
      - ... concluded with x% confidence that the true mean lies between ...

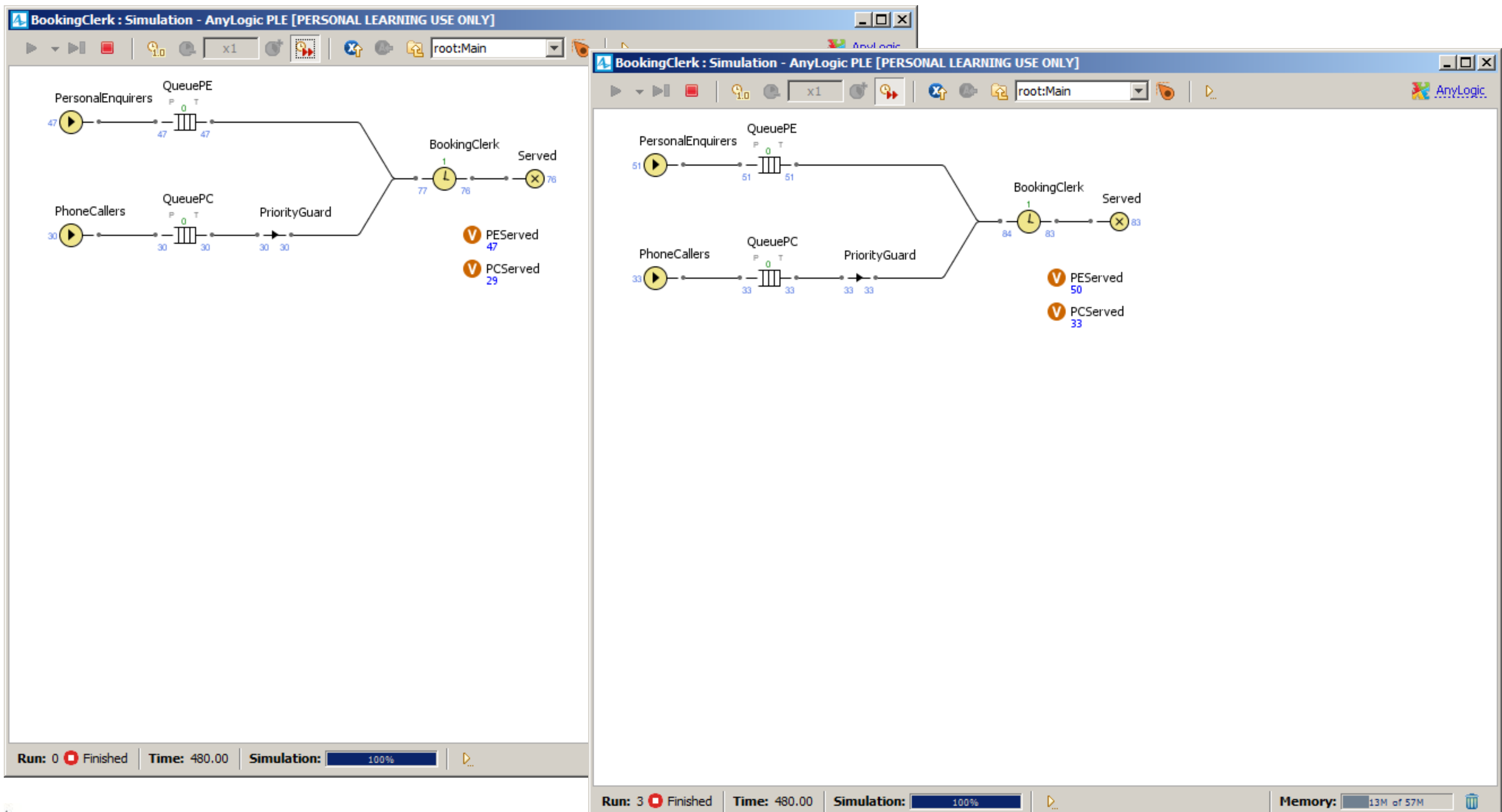
| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | 67 | 55 | 59 | 70 | |
| 2 | 77 | 54 | 67 | 65 | |
| 3 | 72 | 56 | 80 | 64 | |
| 4 | 60 | 75 | 73 | 65 | |
| 5 | 72 | 72 | 65 | 74 | |
| 6 | | | | | |
| 7 | = AVERAGE(A1:D5) | | | 67.10 | |
| 8 | = STDEV(A1:D5) | | | 7.50 | |
| 9 | = CONFIDENCE(0.05,D8,20) | | | 3.29 | |
| 10 | | | | | |
| 11 | Sample Mean | | | | 67.10 |
| 12 | Standard Deviation | | | | 7.50 |
| 13 | 95% Confidence Interval | | | 63.81 | 70.39 |

# Output Analysis

- Comparing two scenarios:
  - Paired-t test (Paired-t confidence interval)
- Comparing many scenarios:
  - Paired-t test and Bonferroni inequality
- For more details:
  - Robinson (2004)

# Discrete Event Simulation

- Simulation Output (two booking clerks)



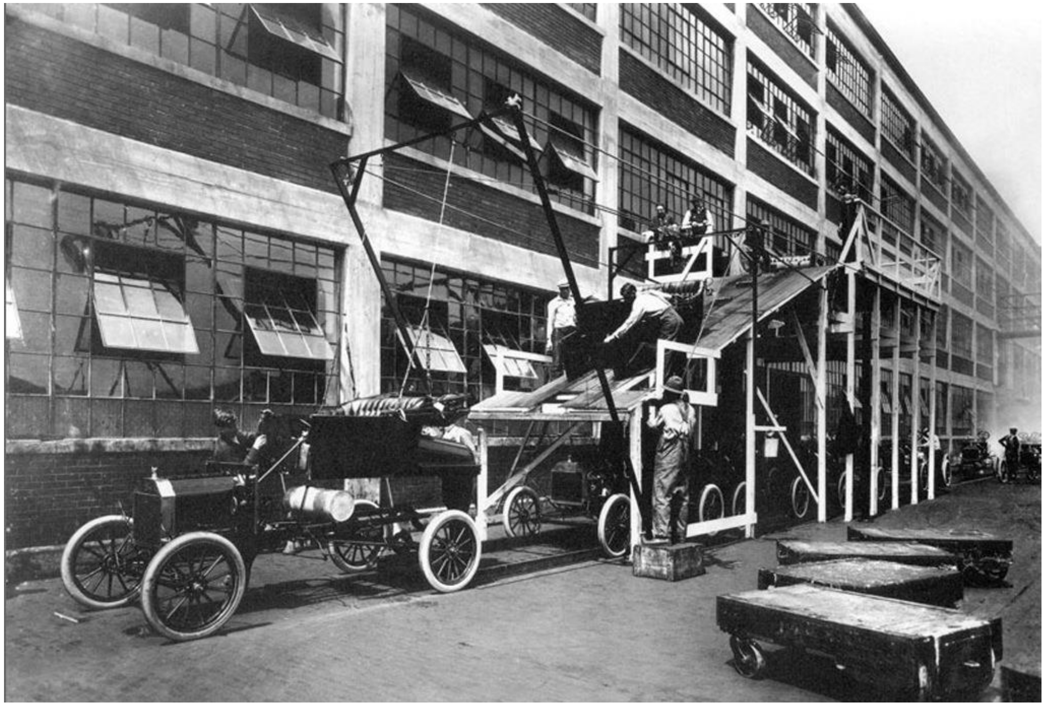
# Case Study (my PhD)

The University of Nottingham

UNITED KINGDOM · CHINA · MALAYSIA

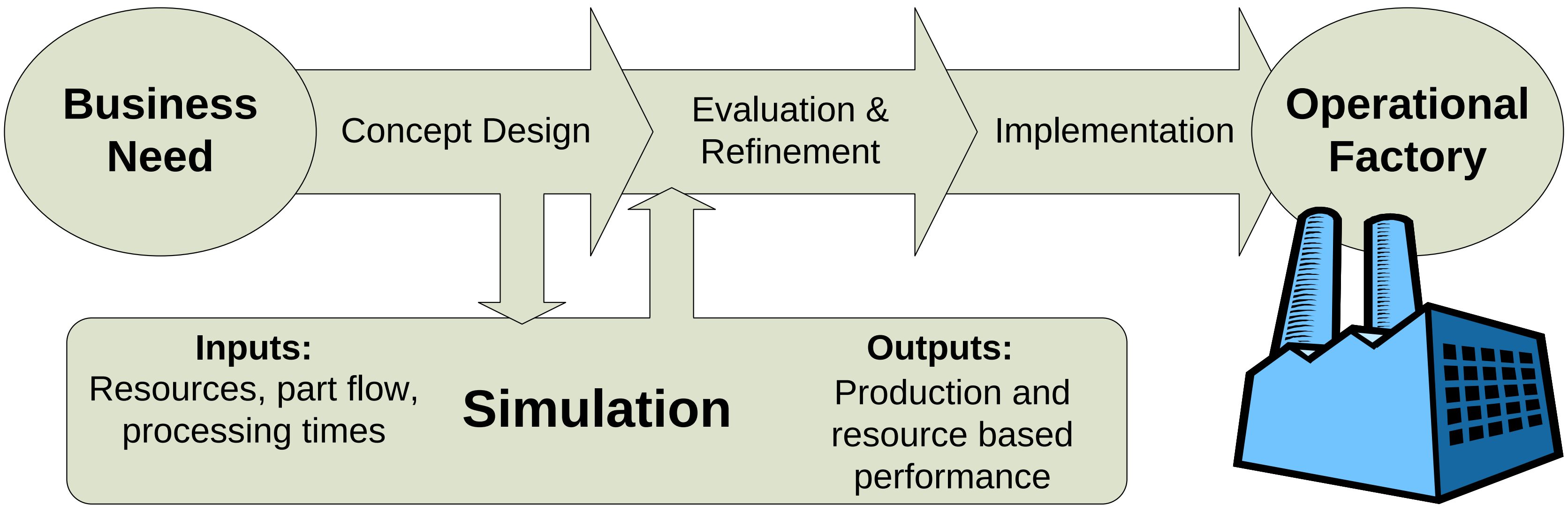
# Case Study

- The Impact of Human Performance Variation on the Accuracy of Manufacturing Systems Simulation Models (Siebers 2004)



# Case Study

- Manufacturing System Design Process



# Case Study


- Statement:
  - Discrete Event Simulation (DES) is now a standard tool used for the design of manufacturing systems within the automotive industry

- Common Observations:
  - A gap exists between the performance prediction of a system model and the performance of the real system
  - Magnitude of the gap is bigger when simulating non existing systems
  - Magnitude of the gap is bigger when simulating manual lines
  - A standard way of taking workers into account is to model them as deterministic resources

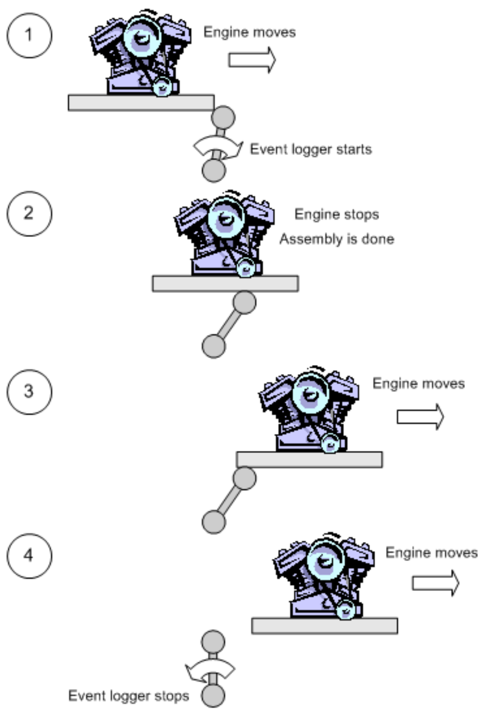
# Case Study

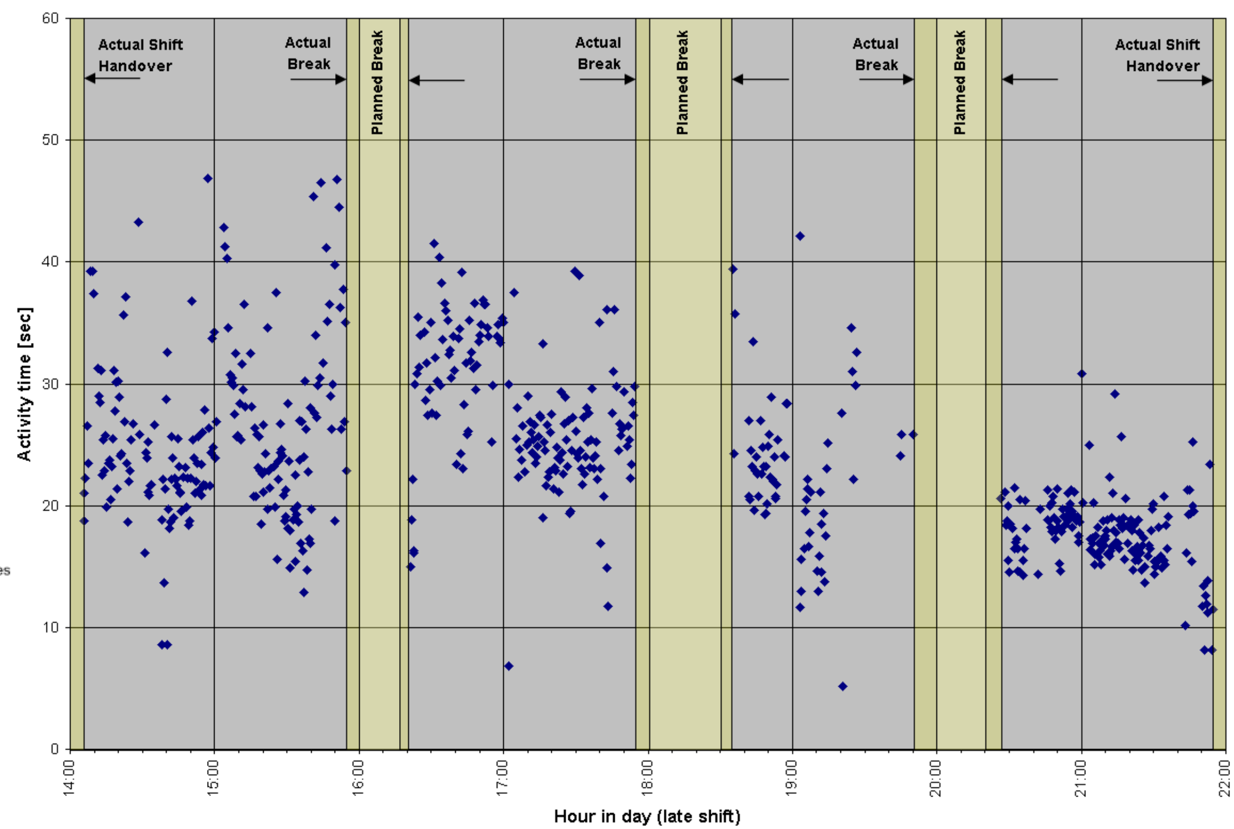
- Research Aim:
  - To demonstrate the importance of incorporating Human Performance Variation (HPV) models into manufacturing system simulation models

- Research Method:
  - Examine of the level of randomness inherent in HPV for different tasks
  - Design of representative HPV models
  - Sensitivity analysis to identify the impact that HPV has on the accuracy of manufacturing systems DES models
  - Literature review for more advanced methods of representing the human element within simulation models

# Case Study

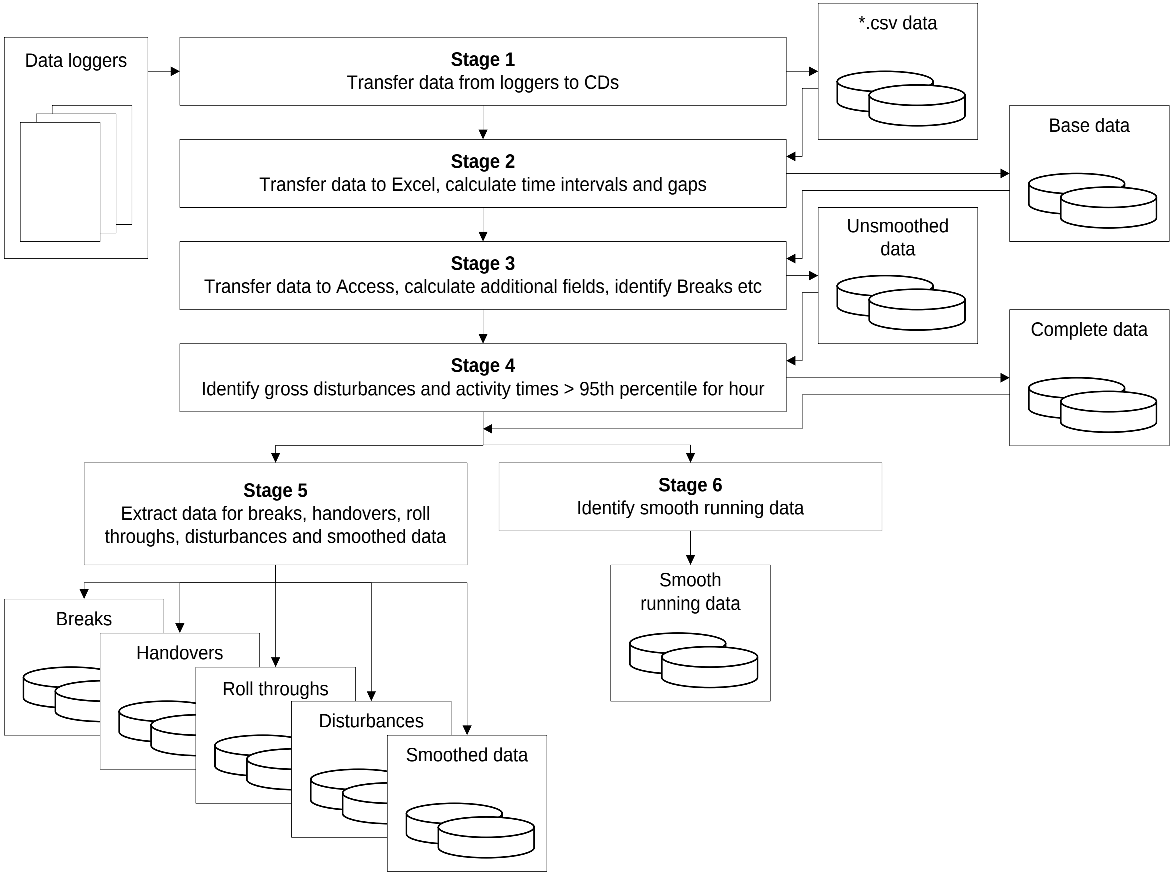
- Step 1: Examining the level of randomness



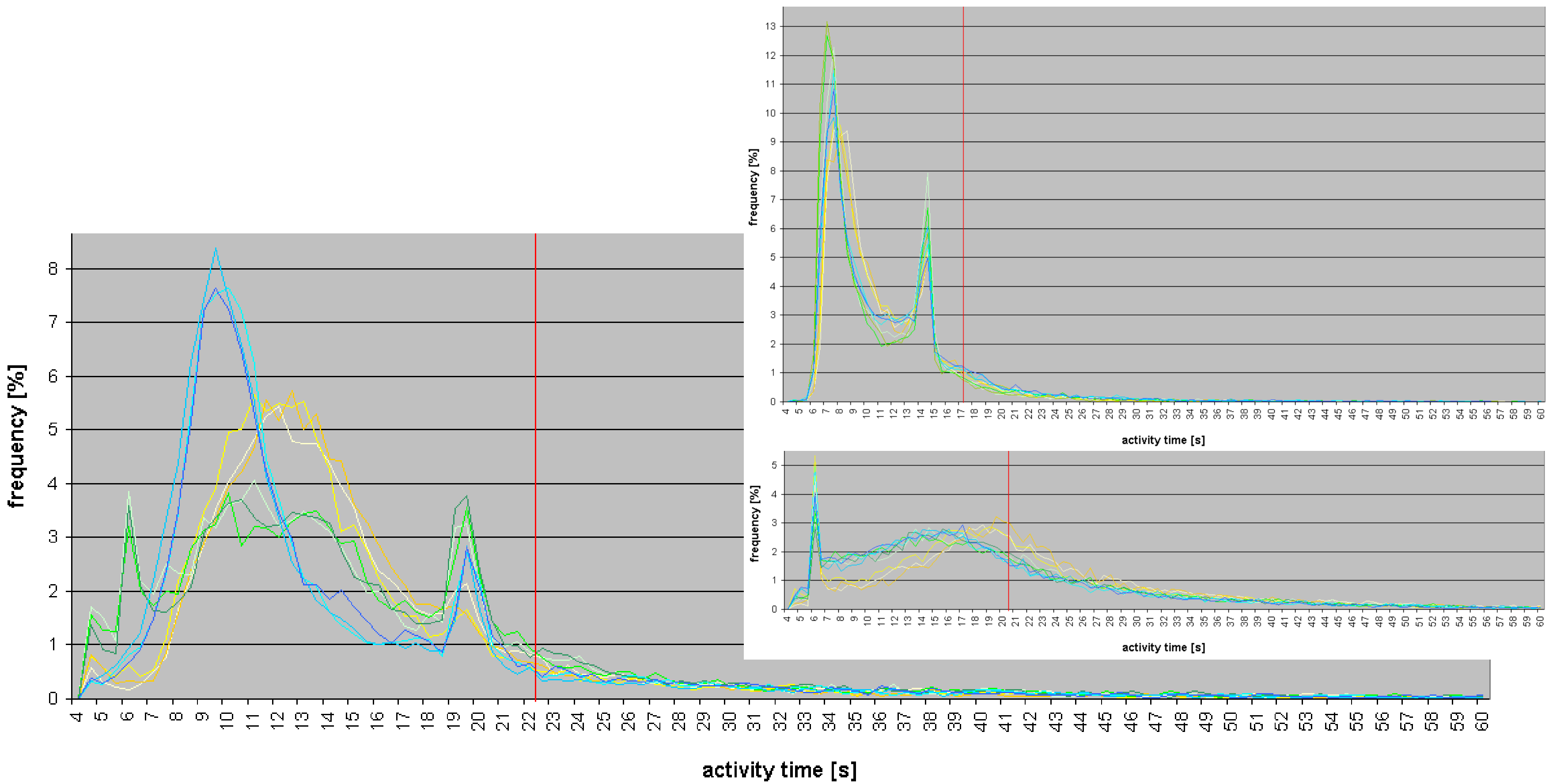
# Case Study

- Step 2: Data analysis



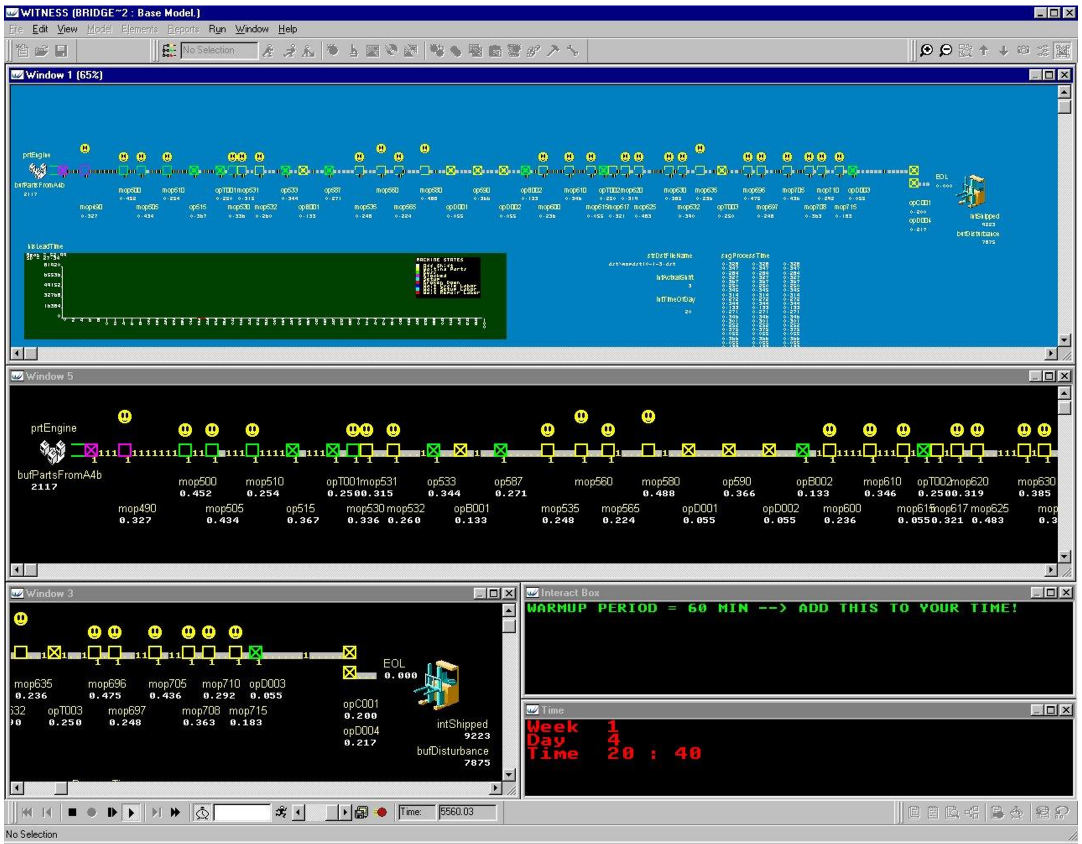
# Case Study

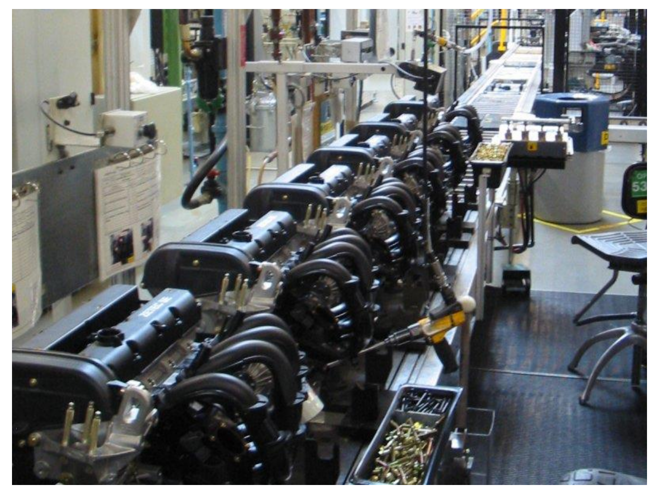
- Step 3: Designing HPV models



# Case Study

- Step 4: Conducting sensitivity analysis



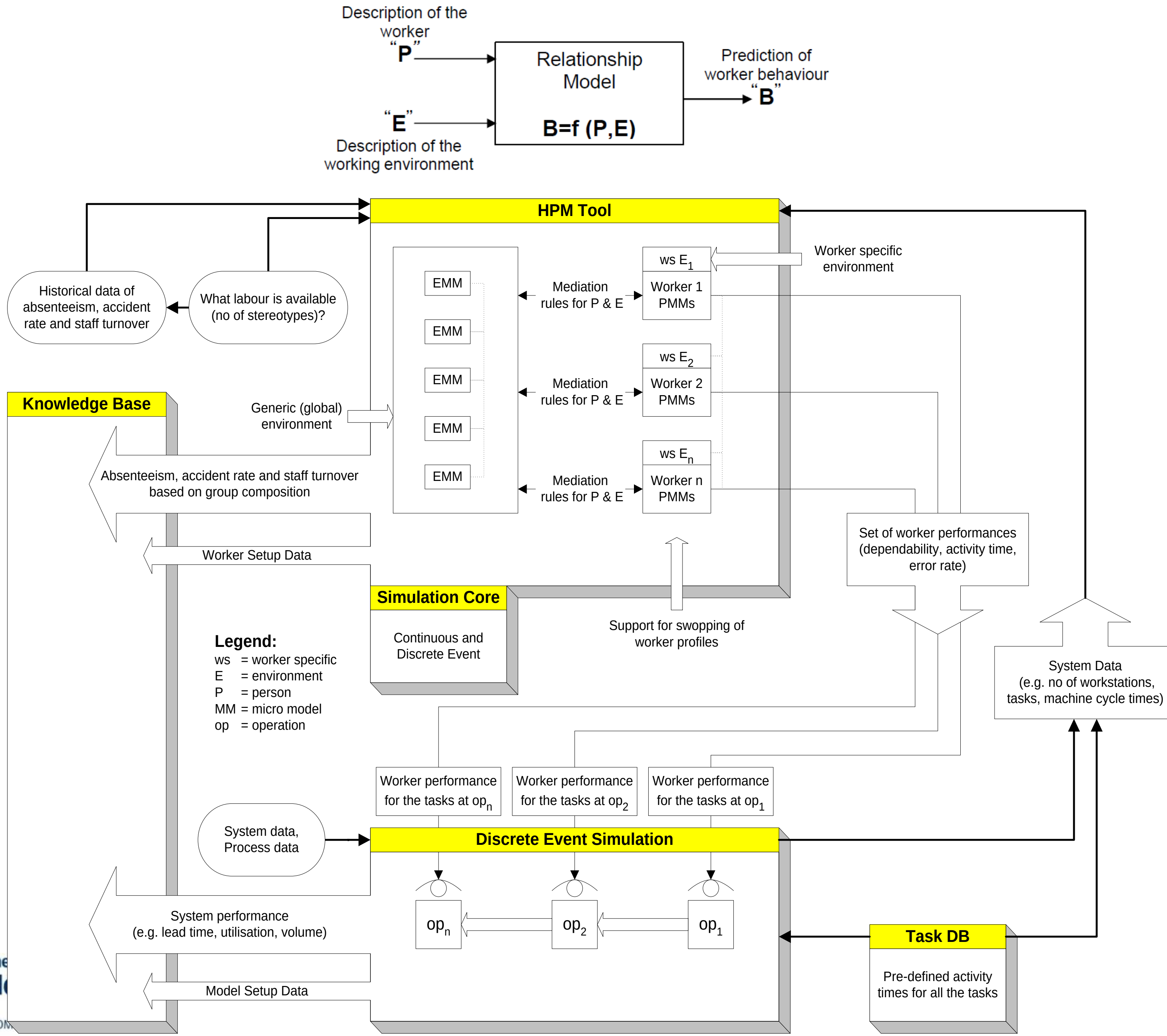
# Case Study

- Key Findings about HPV:
  - Differences in activity times when workers repeat a task, between different workers, and between different work crews
  - Form of activity time distributions depends on the nature of the task
  - Variation of break start and duration does not depend on the length

- Key Findings from Sensitivity Analysis:
  - Representation of HPV can have a significant effect on the behaviour of manufacturing system simulation models
  - The magnitude of impact depends on the type of variation to be represented as well as on the system to be modelled

# Case Study

- Main limitation of current HPV modelling approach:
  - Independent representation of sources of randomness
- Possible solution:
  - Using Computational Organisation Theory as a methodological approach and multi-agent based simulation as a technique
- Issues:
  - Complexity of the task
  - Concept of pro-activeness



Description of the worker
"P"
"E"
Description of the working environment
Relationship Model
B=f (P,E)
Prediction of worker behaviour
"B"
HPM Tool
Historical data of absenteeism, accident rate and staff turnover
What labour is available (no of stereotypes)?
EMM
EMM
EMM
EMM
EMM
Mediation rules for P & E
Mediation rules for P & E
Mediation rules for P & E
ws $E_1$
Worker 1 PMMs
ws $E_2$
Worker 2 PMMs
ws $E_n$
Worker n PMMs
Worker specific environment
Knowledge Base
Generic (global) environment
Absenteeism, accident rate and staff turnover based on group composition
Worker Setup Data
Set of worker performances (dependability, activity time, error rate)
Simulation Core
Continuous and Discrete Event
Support for swopping of worker profiles
System Data (e.g. no of workstations, tasks, machine cycle times)
Legend:
ws = worker specific
E = environment
P = person
MM = micro model
op = operation
Worker performance for the tasks at $op_n$
Worker performance for the tasks at $op_2$
Worker performance for the tasks at $op_1$
System data, Process data
Discrete Event Simulation
$op_n$
$op_2$
$op_1$
System performance (e.g. lead time, utilisation, volume)
Task DB
Pre-defined activity times for all the tasks
Model Setup Data

# Questions and Comments

# References

## References


- Pidd M (1998) Computer Simulation in Management Science {Book}
- Robinson S (2004) Simulation: The Practice of Model Development and Use {Book}
- Siebers PO (2004) The Impact of Human Performance Variation on the Accuracy of Manufacturing System Simulation Models {PhD Thesis}


## Bibliography


- Schriber et al (2012) How Discrete-Event Simulation Software Works and Why it Matters