# Construction Performance and Productivity Improvement

Changbum R. Ahn, PhD
Email : cbahn@snu.ac.kr

SEOUL NATIONAL UNIVERSITY

# Research Methodology (1)
# Introduction to Discrete-Event Simulation

# Acknowledgement

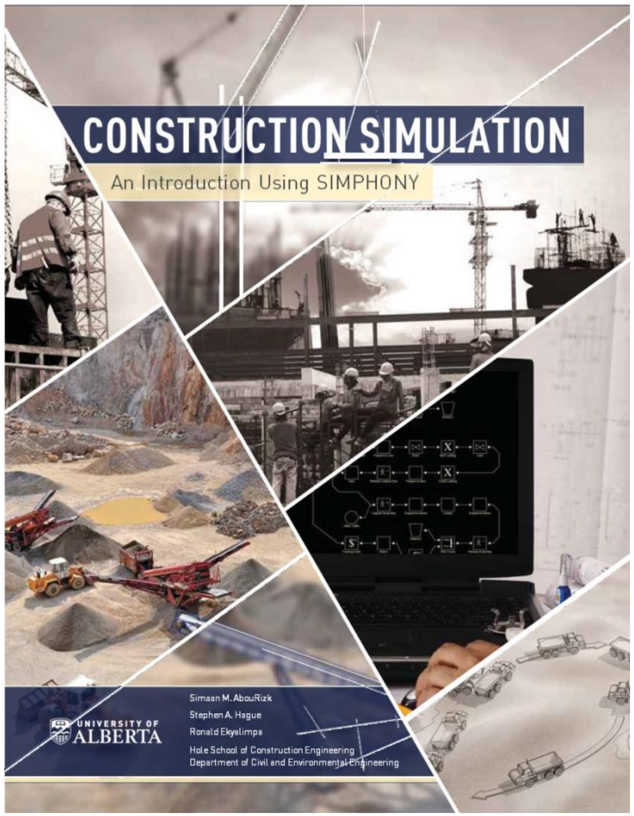
This material is reprinted and reused with permission from Dr. Simaan Abourizk at the University of Alberta. All the copyright of this material belongs to Dr. Simaan Abourizk

# Titans in CM research

- Dr. Robert Peurifoy: Founded the first undergraduate CM program at TAMU in the late 1940s.
- Dr. Daniel Halpin: Established CM as “researchable” area; Developed the first construction simulation model, CYCLONE.
- Dr. Julio Martinez: Developed STROBOSCOPE.
- Dr. Simaan Abourizk: Developed Simphony.

*Read more from “Fifty Years of Progress in Construction Engineering Research”, ASCE Journal of Construction Engineering and Management.*

# Class Materials




**Textbook:**

AbouRizk, S., Hague, S., and Ekyalimpa R. (2016) *An introduction to construction simulation using Simphony*. Edmonton, Alberta: University of Alberta.

**available on ETL*

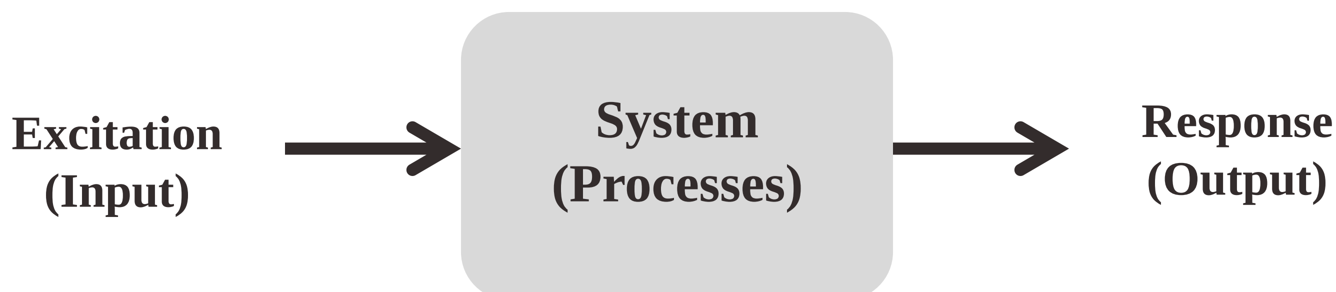
# Unit 1: Outline

- Real world systems, their abstraction, and their analysis
- Model types and simulation models
- Types of simulation and examples
- Simulation software/languages
- Areas of application of simulation

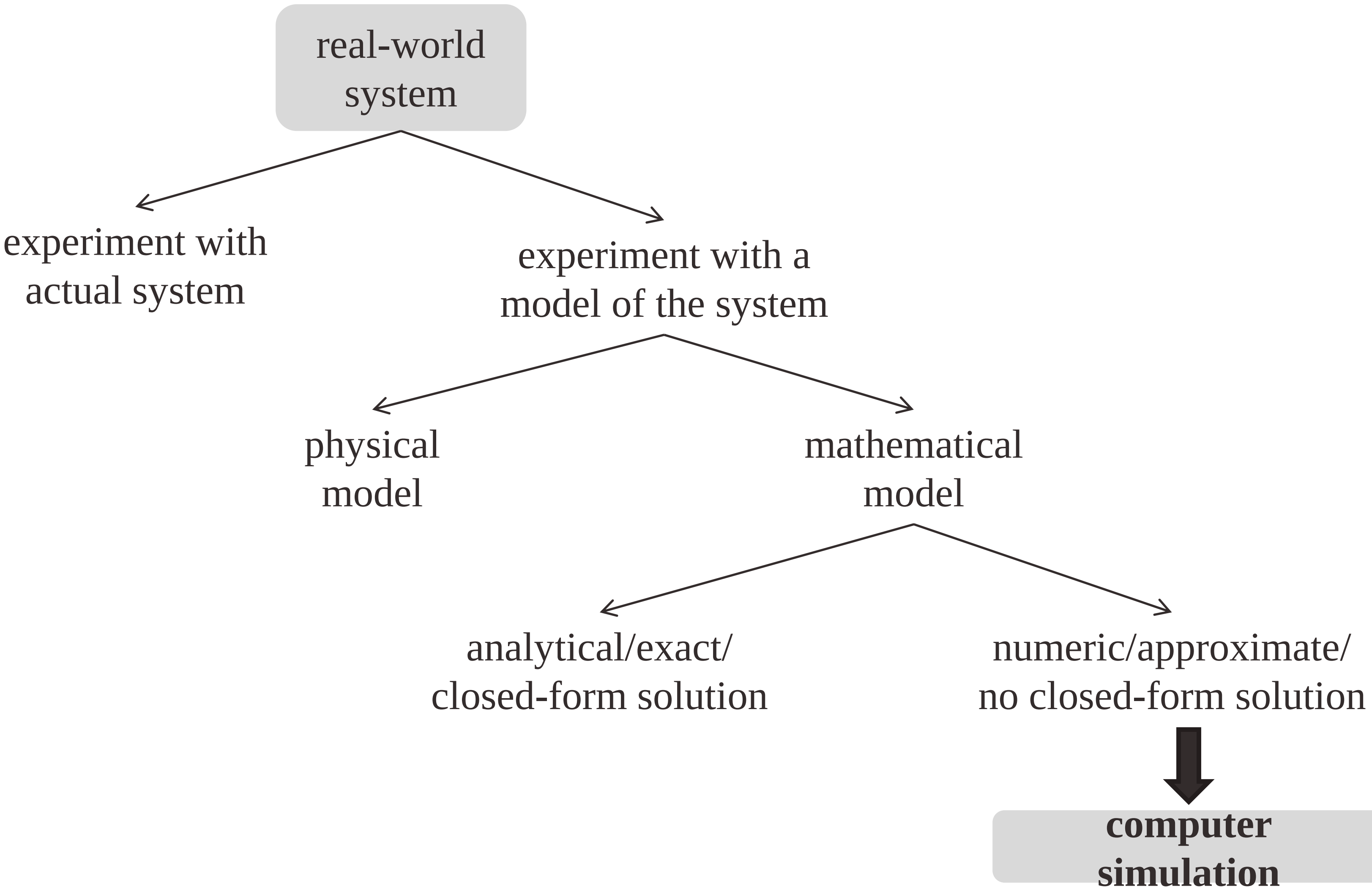
# Definitions of Systems

"A system is a portion of the universe that has been chosen for studying the changes that take place within it in response to varying conditions."

**Encyclopedia Britannica, 2014**



# Analyzing and Designing Systems



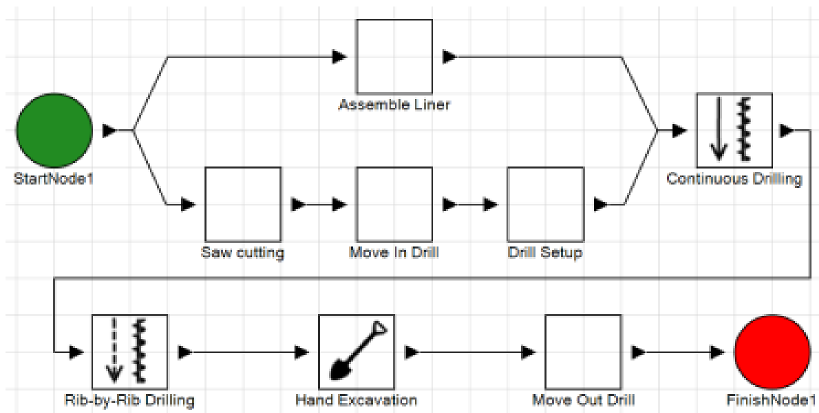
# Definition of a Model

*A useful definition of a model by Wikipedians, n.d.:*

**"A model is anything used in any way to represent/abstract anything else."**

**Examples of Models:**

- Physical Models
- Mathematical Models
  - Conceptual Model(s)
  - Simulation Model(s)




**A simulation model is:**

A composition of objects (often associated with graphical notations) that represent an abstraction of a system. The abstraction is generally in the form of concepts that describe the elements and behavior of a system that are relevant to the model, as determined by the modeller. This collection of objects is used to help us describe the system, study and understand it, and simulate its behavior.

# Matches game

- Setup: 6 players, 6 bowls, matches, 1 die
- Dump all matches in bowl #1
- Roll one die (starting with player #1) and pass that many matches from your bowl to the next person down the line
- Pass die to next player who rolls die and moves that number of matches from their bowl to next player, cannot pass more matches than what is in your bowl.
- Continue for each player, with last player handing die back to player #1.

- After 5 rounds, how many matches should the last player "produce"?

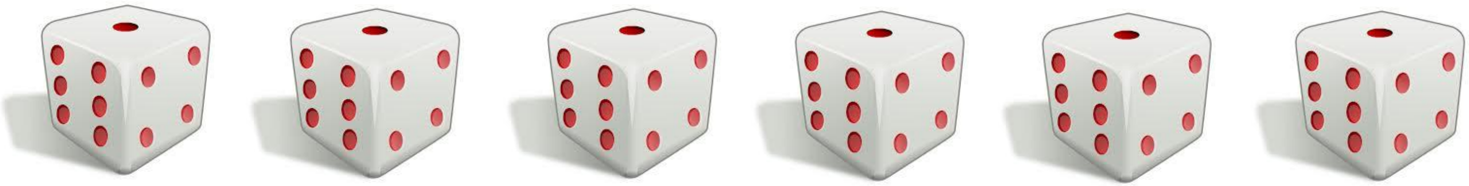
# Matches game: Expectation

| Average | 3.5 | 3.5 | 3.5 | 3.5 | 3.5 | 3.5 |
|---|---|---|---|---|---|---|

Average “produce” of each round 3.5

Final expected “produce” after 5 rounds $3.5 \times 5 = 17.5$

# Recap - Matches game



- Throughput of six dices system cannot reach to the average output of one dice. Why?

- What would be the solution to resolve this production loss?

# Attributes of a Simulation Model(s)/Problem(s)

**Real-world system/operation/process**

**Abstraction (Model):**
Logic
Time variation

**Abstraction (Model):**
Logic
Randomness

**Abstraction (Model):**
Logic
Time variation
Randomness

# Types of Simulation

## ❶ Deterministic vs. Stochastic Simulation Models:

### Probability/Randomness Dimension

- **Stochastic Simulations:**
  *Exhibit randomness.*
  - Use probability/statistical distributions
  - Use probabilistic branching concepts
  - Example is Monte Carlo simulation models

# Types of Simulation

## 2 Static vs. Dynamic Simulation Models

### Time Dimension

- **Dynamic Simulations:**
  *State variables change as time evolves.*
  - Discrete Event Simulation (DES)
  - Continuous Simulation (CS)

- **Static Simulations:**
  *No time aspects.*
  - Some Monte Carlo simulation models (e.g. direct cost estimation, risk assessment models, etc.)

# Types of Simulation

## 3 Distributed vs. Monolithic Simulation

## 4 Simulation Modelling Paradigms

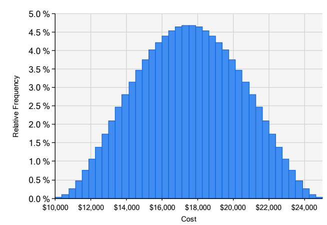
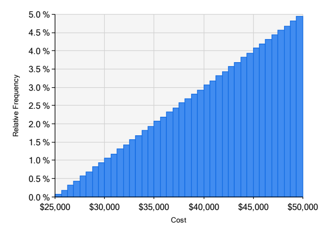
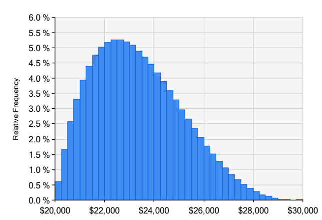
*Can apply to any of the above.*

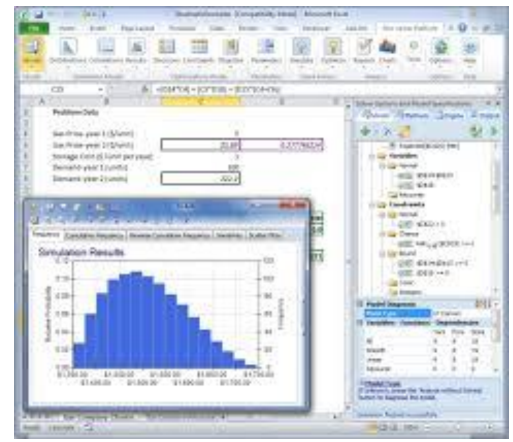
- Process interaction modelling (DES)
- Agent-Based Modelling
- System Dynamics

# EXAMPLES OF THE DIFFERENT SIMULATION MODEL TYPES

## Monte Carlo-Based Simulation Models

*Static and Stochastic*

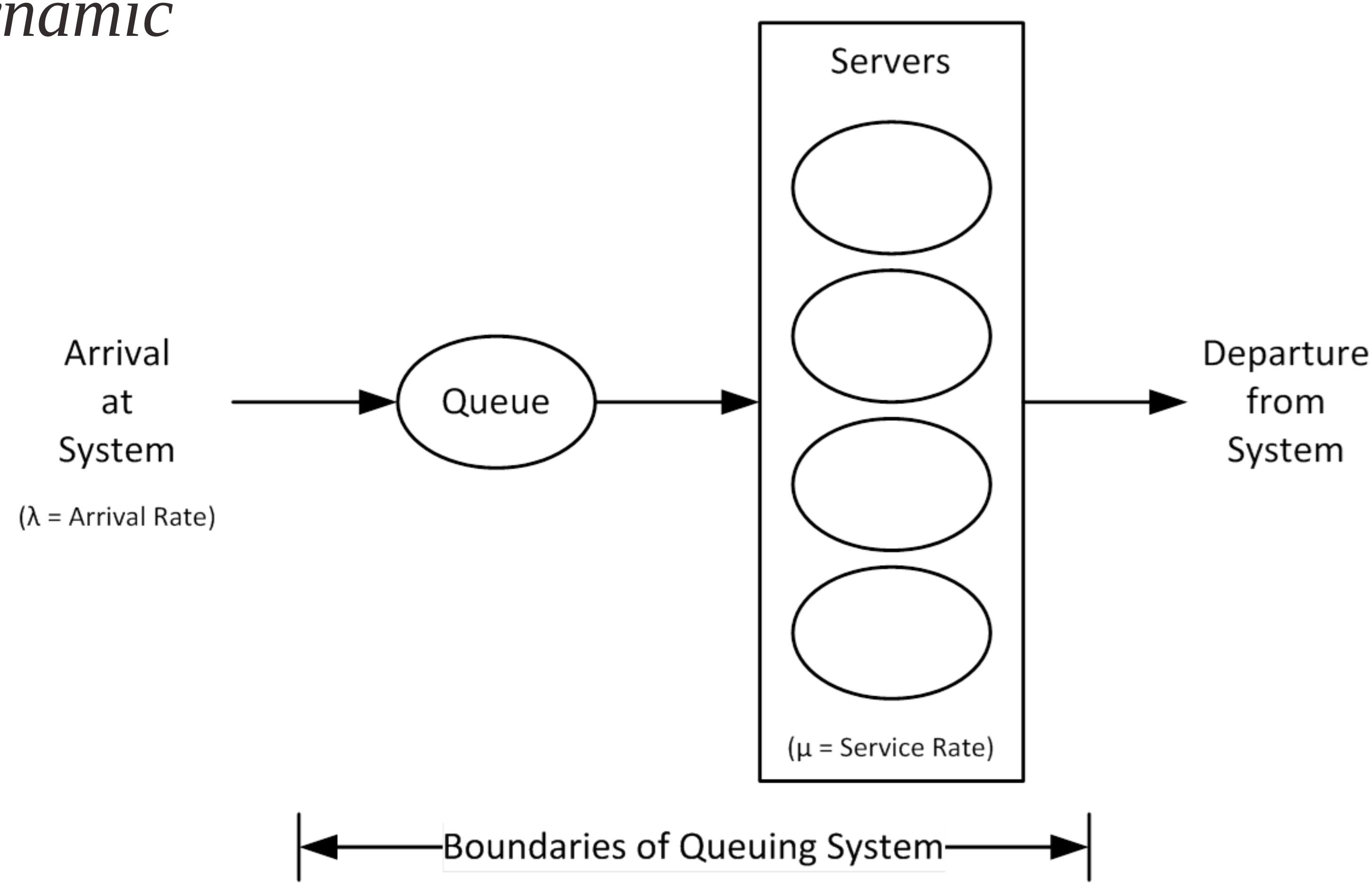
| Work Package | Minimum | Maximum | Impact Distribution |
|---|---|---|---|
| Work Package A | $10,000 | $25,000 |  |
| Work Package B | $25,000 | $50,000 |  |
| Work Package A | $20,000 | $30,000 |  |




# EXAMPLES OF THE DIFFERENT SIMULATION MODEL TYPES

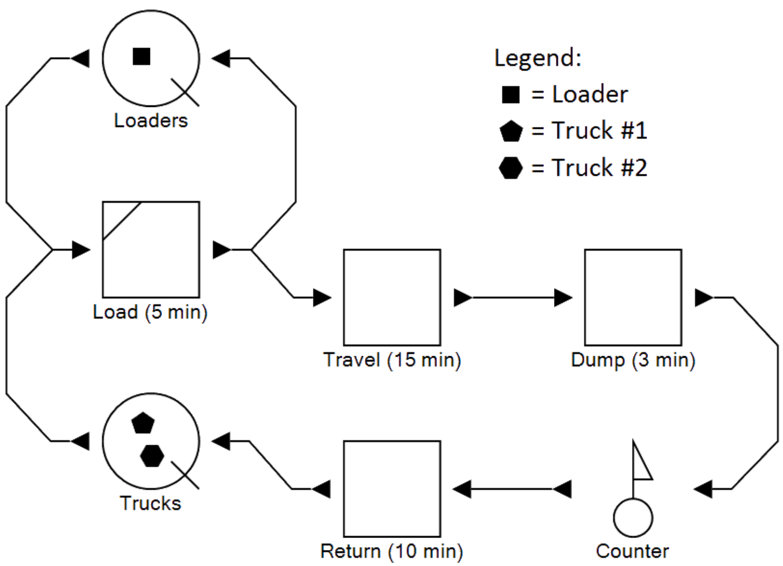
## Queuing System

*Dynamic*



## EXAMPLES OF THE DIFFERENT SIMULATION MODEL TYPES

# Dynamic, Deterministic DES Simulation Models

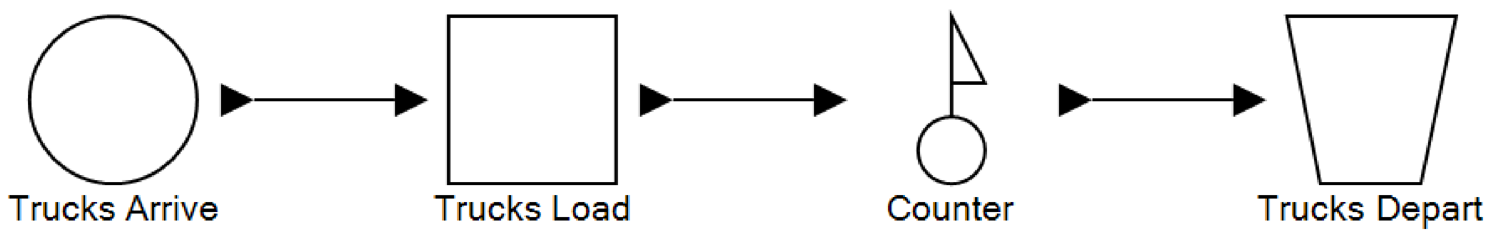



| Activity | Completion Time (min.) | Production Rate |
|---|---|---|
| Load (Truck #1) | 5 | |
| Load (Truck #2) | 10 | |
| Travel (Truck #1) | 20 | |
| Dump (Truck #1) | 23 | 1/23 = 0.0435 |
| Travel (Truck #2) | 25 | |
| Dump (Truck #2) | 28 | 2/28 = 0.0870 |
| Return (Truck #1) | 33 | |
| Load (Truck #1) | 38 | |
| Return (Truck #2) | 43 | |
| Load (Truck #2) | 53 | |
| Travel (Truck #1) | 53 | |
| Dump (Truck #1) | 56 | 3/56 = 0.1304 |
| Travel (Truck #2) | 58 | |
| Dump (Truck #2) | 61 | 4/61 = 0.1739 |
| Return (Truck #1) | 66 | |
| Return (Truck #2) | 71 | |

# EXAMPLES OF THE DIFFERENT SIMULATION MODEL TYPES

## Truck-Shovel DES Simphony Model

*Dynamic, Deterministic*

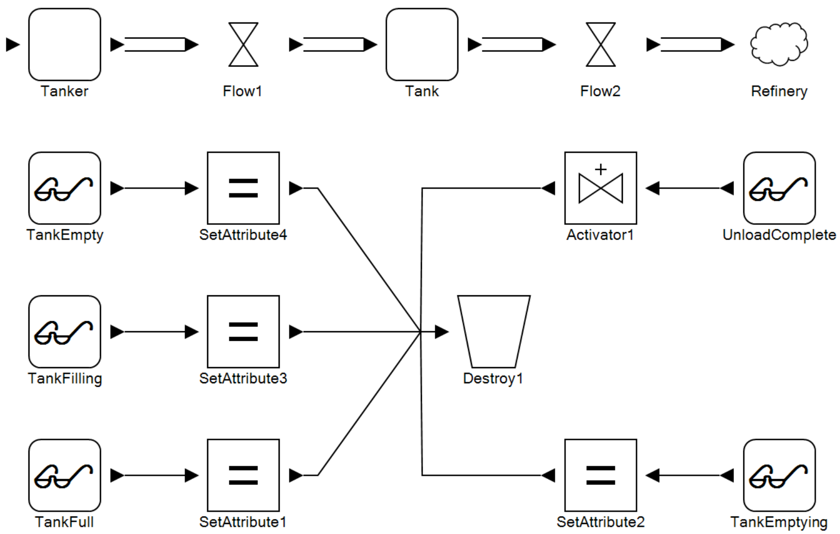


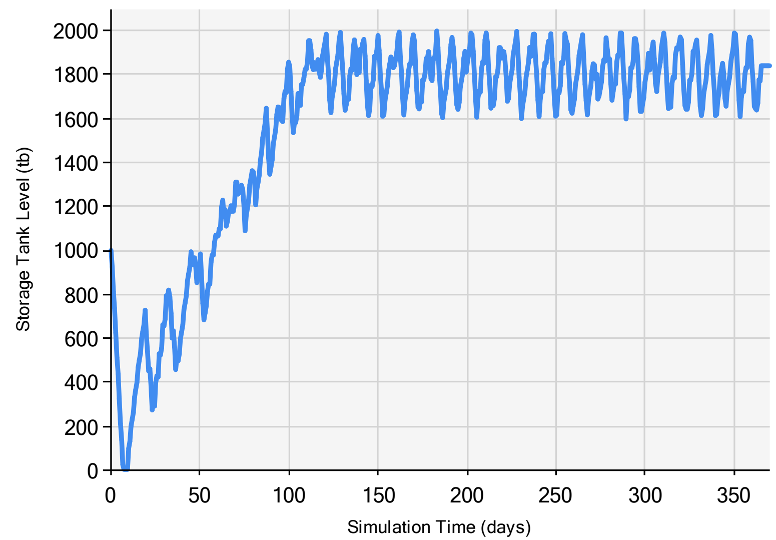

- The arrival rate: $\lambda = A / T = 18$ trucks / 60 min = 0.30 trucks/min;
- The service rate: $\mu = C / T = 15$ trucks / 60 min = 0.25 trucks/min;
- The server utilization: $U = B / T = 60$ min / 60 min = 100%; and
- The average service time per truck: $S = B / C = 60$ min / 15 trucks = 4 min/truck.

# EXAMPLES OF THE DIFFERENT SIMULATION MODEL TYPES

## Continuous Change Models

*Dynamic, Stochastic*



# Simulation Software

**Simulation software developed in academic institutions:**

- Micro CYCLONE (Halpin, 1973)
- STROBOSCOPE (Martinez, 1996)
- Simphony (AbouRizk and Hajjar, 1998)
- ABC (Approximate Bayesian Computation) (Beaumont, Zhang, & Balding, 2002)
- SLAM (Initially developed at Purdue University and then became a fairly well-known commercial system) (Pritsker, O'reilly, & Laval, 1997)
- Repast Simphony

**Commercial simulation software:**

- ARENA (Rockwell Automation, 2000)
- AnyLogic (The AnyLogic Company, 2000)
- SIMSCRIPT (CACI Advanced Simulation Lab, 2014)
- SIMIO
- Vensim

# Key Features in Simulation Software

- An interface (modelling surface, properties, traces)
- A simulation engine
- Modelling constructs
  - Flow tokens (entities)
  - Templates or libraries
  - Other libraries (e.g. math)

# Key Features in Simulation Software

A template is defined as a collection of abstract modelling elements.

**Types of templates:**

- General Purpose Template (GPT):
  - A collection of high-level elements that may not resemble the system in the real world.
- Special Purpose Template (SPT):
  - A collection of modelling elements that are designed to have a behavior that is customized to a specific process. These elements usually have icons that resemble the real world system they represent.

# Applications of Simulation in Construction

## During Design:

- Risk analysis,
- Value analysis,
- Constructability reviews (scenario-based planning),
- Construction plan development,
- Budget development, and
- Estimating.

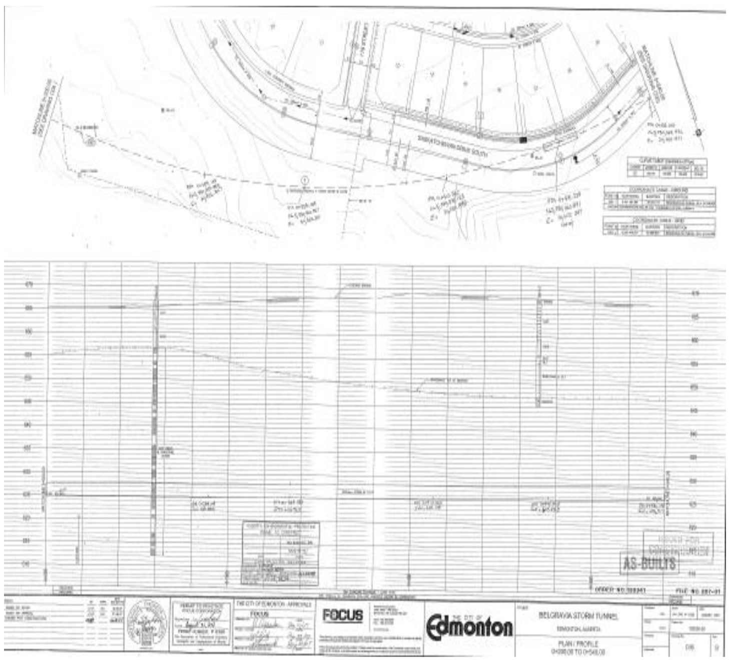
## During/Post Construction:

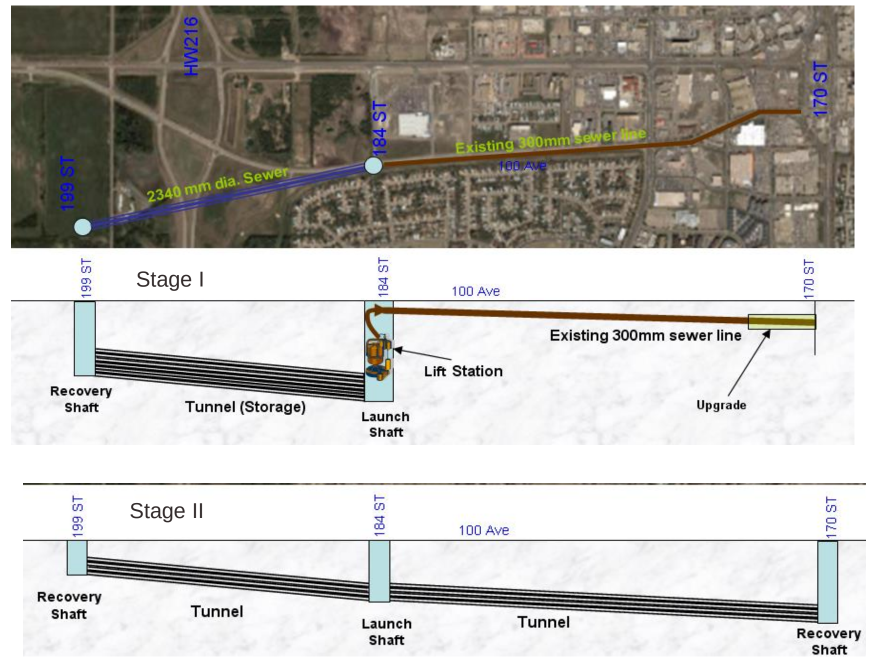
- Planning and control,
- Continuous improvement, and
- Claims and dispute resolution.

## SIMULATION APPLICATION IN CONSTRUCTION

# Tunneling

Sample project sanitary servicing for a new development (City of Edmonton)



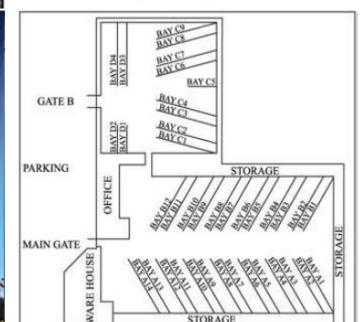



## SIMULATION APPLICATION IN CONSTRUCTION

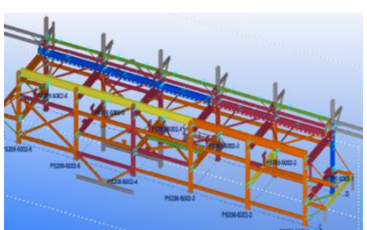
# Fabrication Facilities

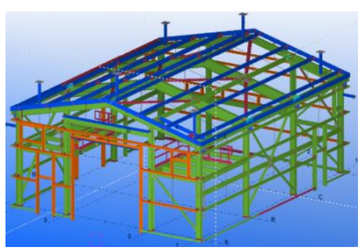
Pipe Spool or Structural Steel Fabrication Shop

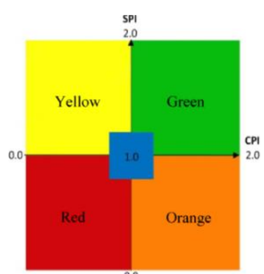
Module Yard




Construction Site (Erection)



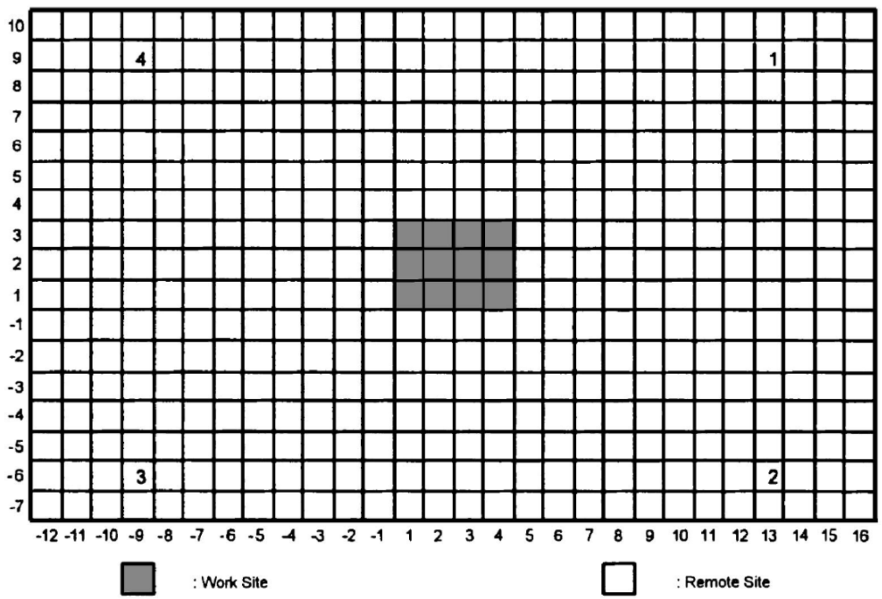




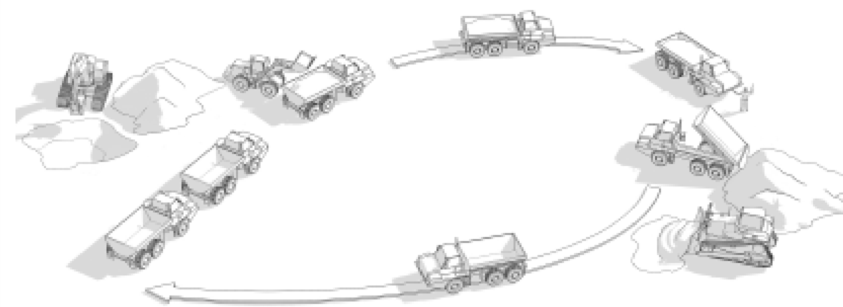

## SIMULATION APPLICATION IN CONSTRUCTION

# Earthmoving

Optimizing Mass Excavation Operations




Micro-Level Analysis of Earthmoving Operations

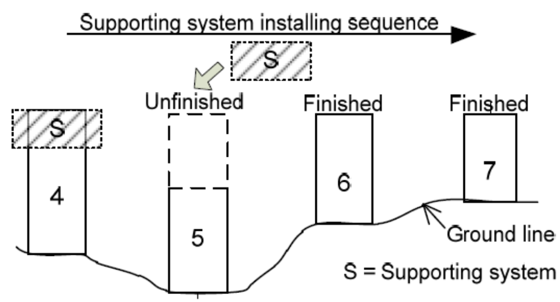


## SIMULATION APPLICATION IN CONSTRUCTION

# Bridge Construction

### Construction Method – Incremental Launching

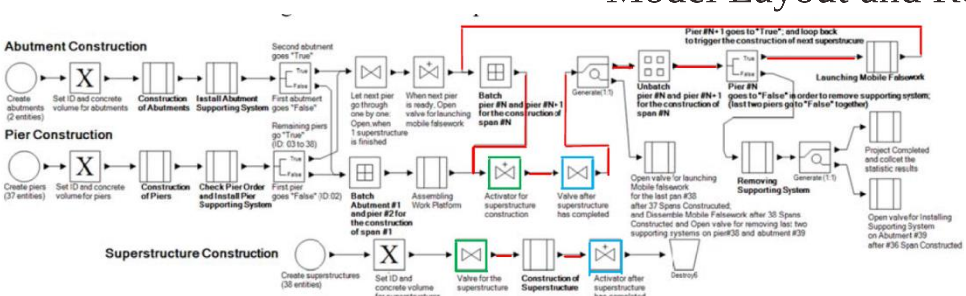
### Site Condition Constrains




### Scenarios to Investigate – Construction Sequence

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Alternative 1 | → | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Alternative 2 | → | 2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 2 | ← |
| Alternative 3 | | | | | | | | | | | | | | | | | | 1 | ← | → | 2 | | | | | | | | | | | | | | | | | |

### Model Layout and Results




| Resource | Alternative 1 | Alternative 2 | Alternative 3 |
|---|---|---|---|
| Formwork for Superstructure | 0.2057 | 0.2675 | 0.2684 |
| Formwork for Pier Head | 0.084 | 0.1056 | 0.102 |
| Formwork for Shaft | 0.1125 | 0.1388 | 0.1327 |
| Formwork for Foundation | 0.0776 | 0.0958 | 0.0918 |
| Concrete Pump | 0.173 | 0.1981 | 0.1977 |
| Concrete Worker | 0.1389 | 0.1686 | 0.1624 |
| Excavator | 0.0904 | 0.113 | 0.1081 |
| Reinforcement Worker | **0.5816** | **0.6706** | **0.6994** |
| Pre-stress Crew | 0.1413 | 0.1725 | 0.1705 |
| Launching Crew | 0.2417 | 0.2985 | 0.2873 |
| Working Platform | **0.9111** | **0.7386** | **0.6464** |
| Supporting System (Segment 1) | **0.9246** | **0.8585** | 0.7983 |
| Supporting System (Segment 2) | - | 0.6636 | **0.8596** |

# Benefits of Using Simulation

**Information-based decision support:**

- Bottleneck identification
- Optimal solutions obtained
- Performance measure generation (e.g. production rates, cycle times, wait times)
- Scenario comparison

**Standard creation using results:**

- Creation of standards
- Recommendation of best practices

# Conclusions

- Gain an appreciation of how to identify problems that are suitable for solving using simulation based approaches.
- Insights into different types of models and modelling approaches: Affects the way you run models and analyze results (e.g. deterministic vs. stochastic models).
- Appreciation of other general simulation concepts.

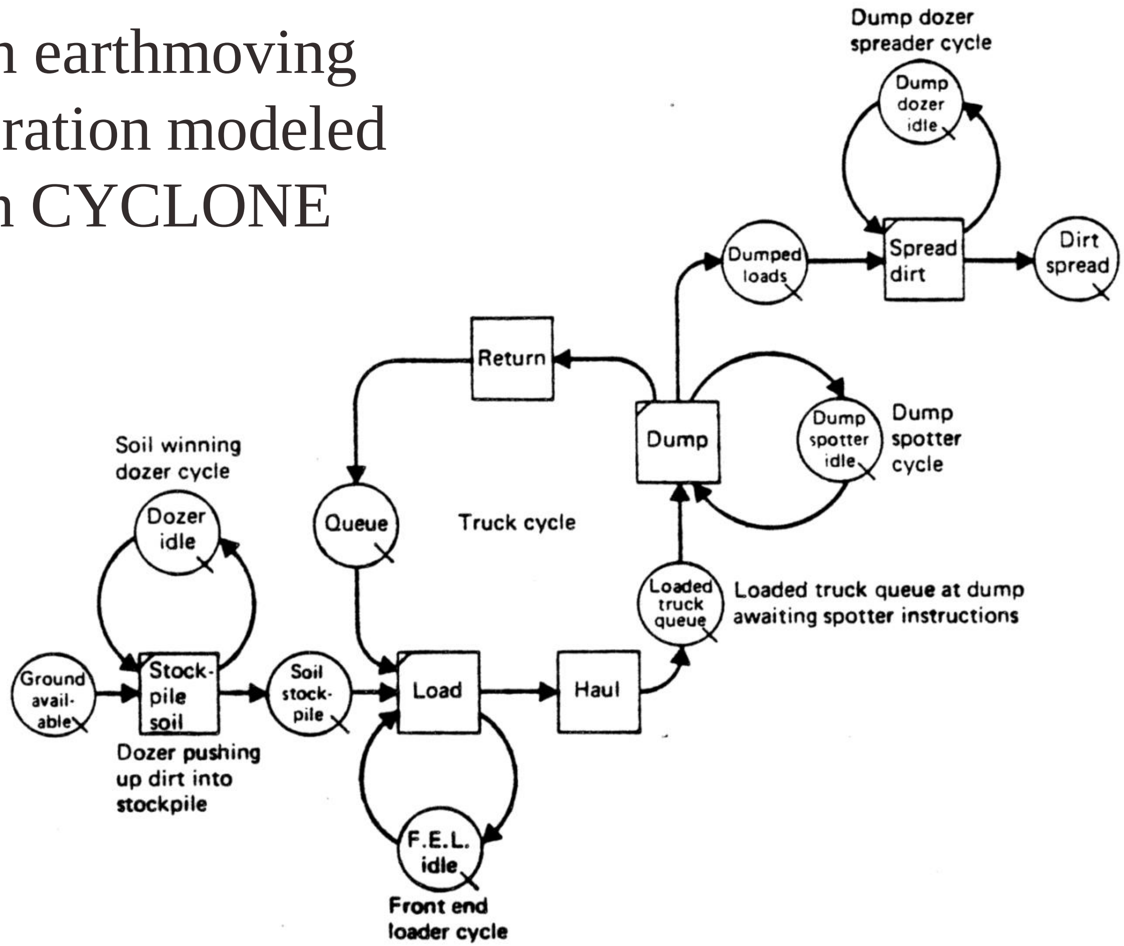
# Research Methodology (1)
# Introduction to Cyclone

# What is CYCLONE?

- CYCLONE is a graphical tool for simulation of discrete systems (i.e. construction operations).
- CYCLONE is an acronym that stands for CYCLic Operations NEtwork.
- CYCLONE was developed in the 1970s and 1980s by Dr. Dan Halpin at Purdue University.
- These slides are based on the MicroCYCLONE System Manual, Halpin (1990).

# What Does a Model Look Like?

An earthmoving operation modeled in CYCLONE



# A CYCLONE Model

## Models in CYCLONE consist of:

- Modelling elements that define the behaviour of the model.
- Entities (flow units) that flow through and between the modelling elements.
- Arrows that direct the flow of entities.

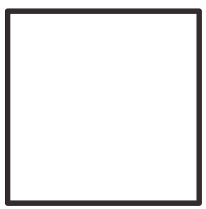
# Entities

Entities are the things that flow through a process; examples might be:

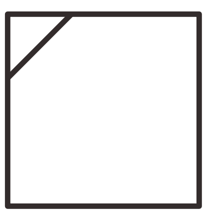
- The widgets being manufactured in a manufacturing process.
- The trucks traveling in an earthmoving operation.

When creating a simulation model, you need to decide what the (abstract) entities represent.

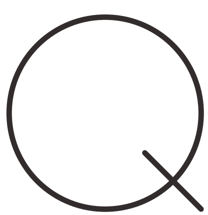
# Primary Modelling Elements



**NORMAL:** An unconstrained work task. Entities are delayed for a specified amount of time.

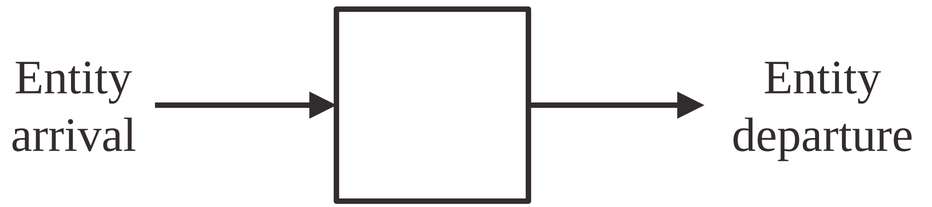


**COMBI:** A constrained (in terms of its starting logic) work task; otherwise, identical to a NORMAL.



**QUEUE:** The idle state of an entity. Represents a queue in which entities are waiting for use.

# The NORMAL Element

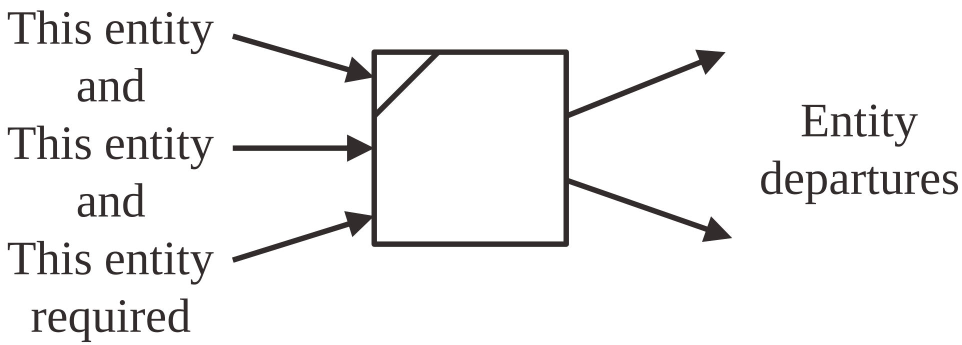



For NORMAL work tasks, the arrival of an entity at the input side is sufficient to allow the task to begin.

It is permissible for several entities to traverse a NORMAL simultaneously.

Its output point cannot be connected to the Combi.

# The COMBI Element

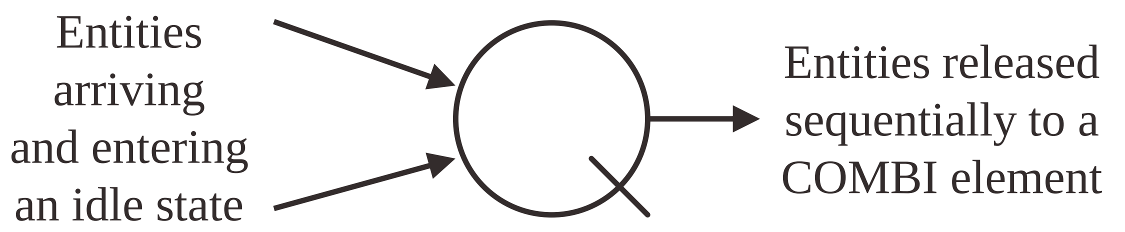



Similar to the NORMAL, but with the added requirement that a prescribed set of input requirements must be met before the task can begin.

Its input point should be connected to Queue.

Its output point can be connected to any element except Combi.

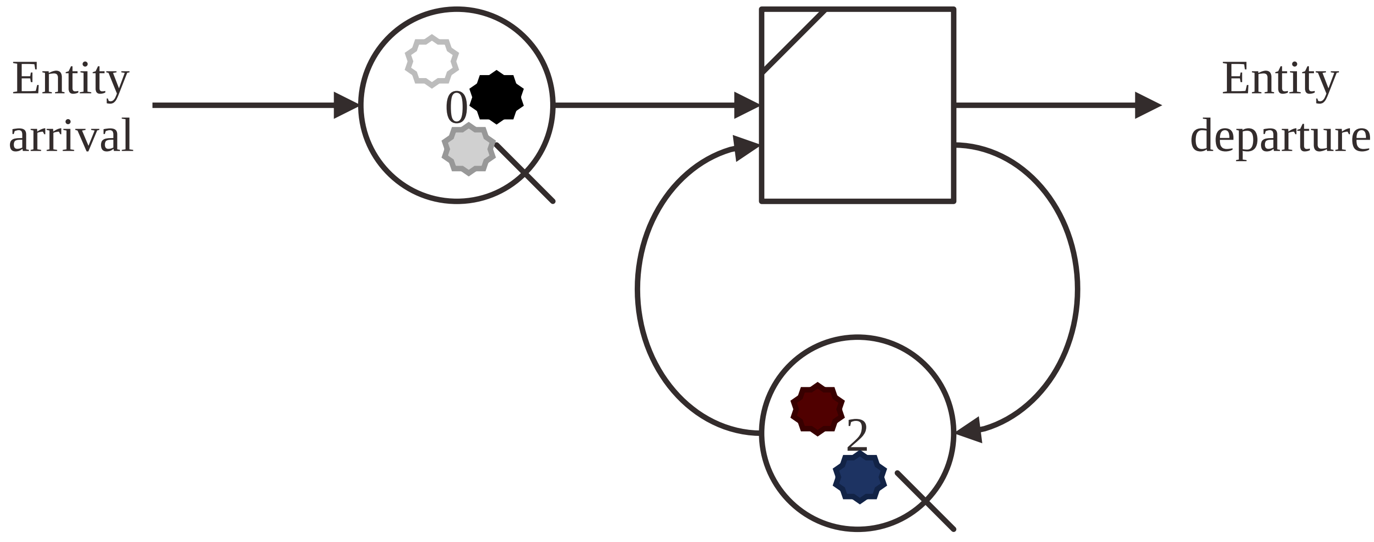
# The QUEUE Element




A QUEUE acts as a storage location for entities in the idle state.

QUEUEs always precede COMBIs, and the set of QUEUEs associated with a COMBI establishes its starting logic.

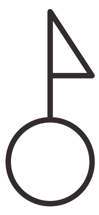
# Example: Constrained Task



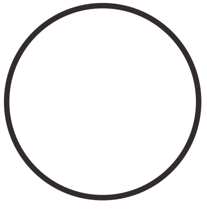

This demonstrates how to model an activity in which at most two entities can participate at any time.

**Note:** The numbers inside the QUEUEs specify the number of entities the QUEUE contains at the start of simulation.

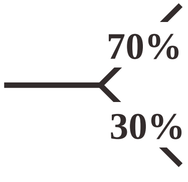
# Secondary Modelling Elements



**COUNTER:** Counts passing entities. Primarily used to track production in a system.

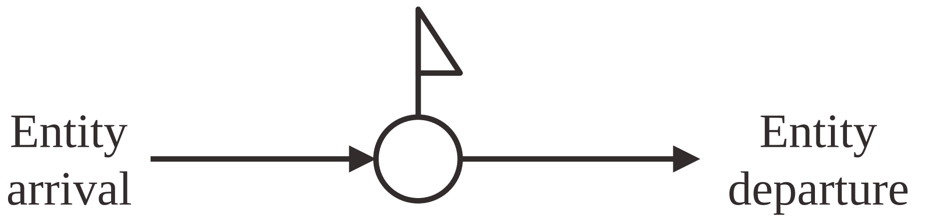


**FUNCTION:** Generates or consolidates entities.




**BRANCH:** Routes entities probabilistically.

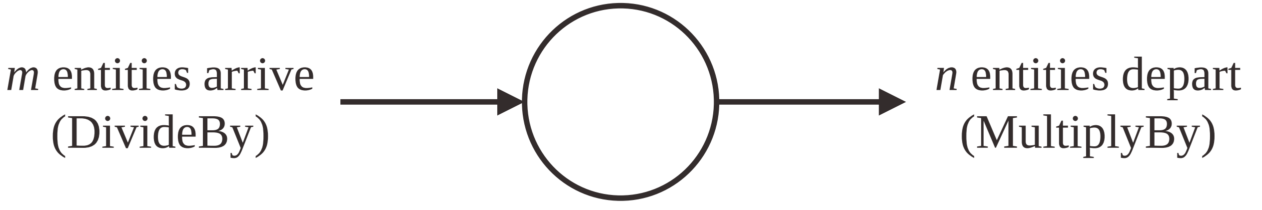
# The COUNTER Element




The COUTER element counts passing entities. Entities are not delayed in a counter—they pass through instantaneously.

A COUNTER can be configured to terminate simulation after a specified number of entities have been observed.

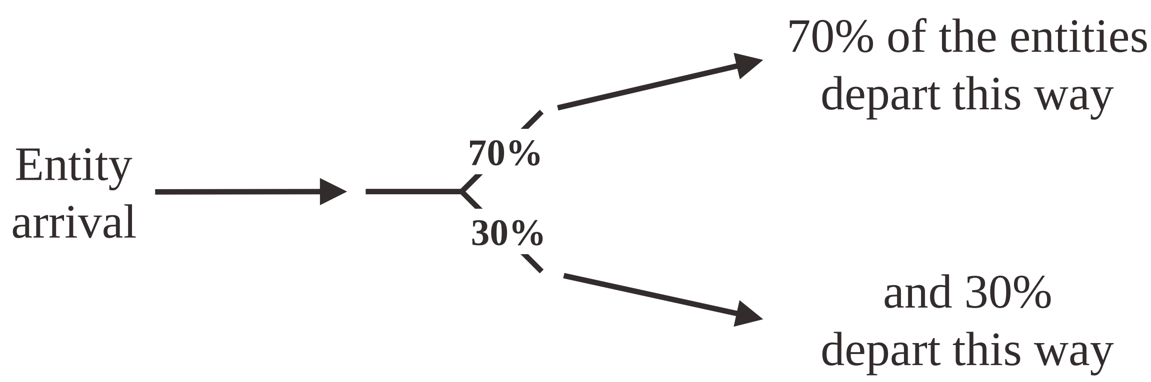
# The FUNCTION Element




The FUNCTION element generates or consolidates entities.

The departing entities will not depart until the full complement of entities have arrived.

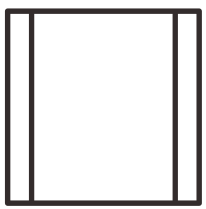
# The BRANCH Element



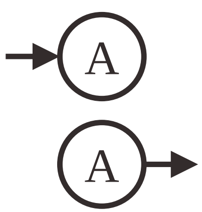

The BRANCH element routes a stream of entities probabilistically. A certain percentage will follow one branch, while the remainder will follow via the other.

Entities are not delayed by a BRANCH element.

# Structural Modelling Elements

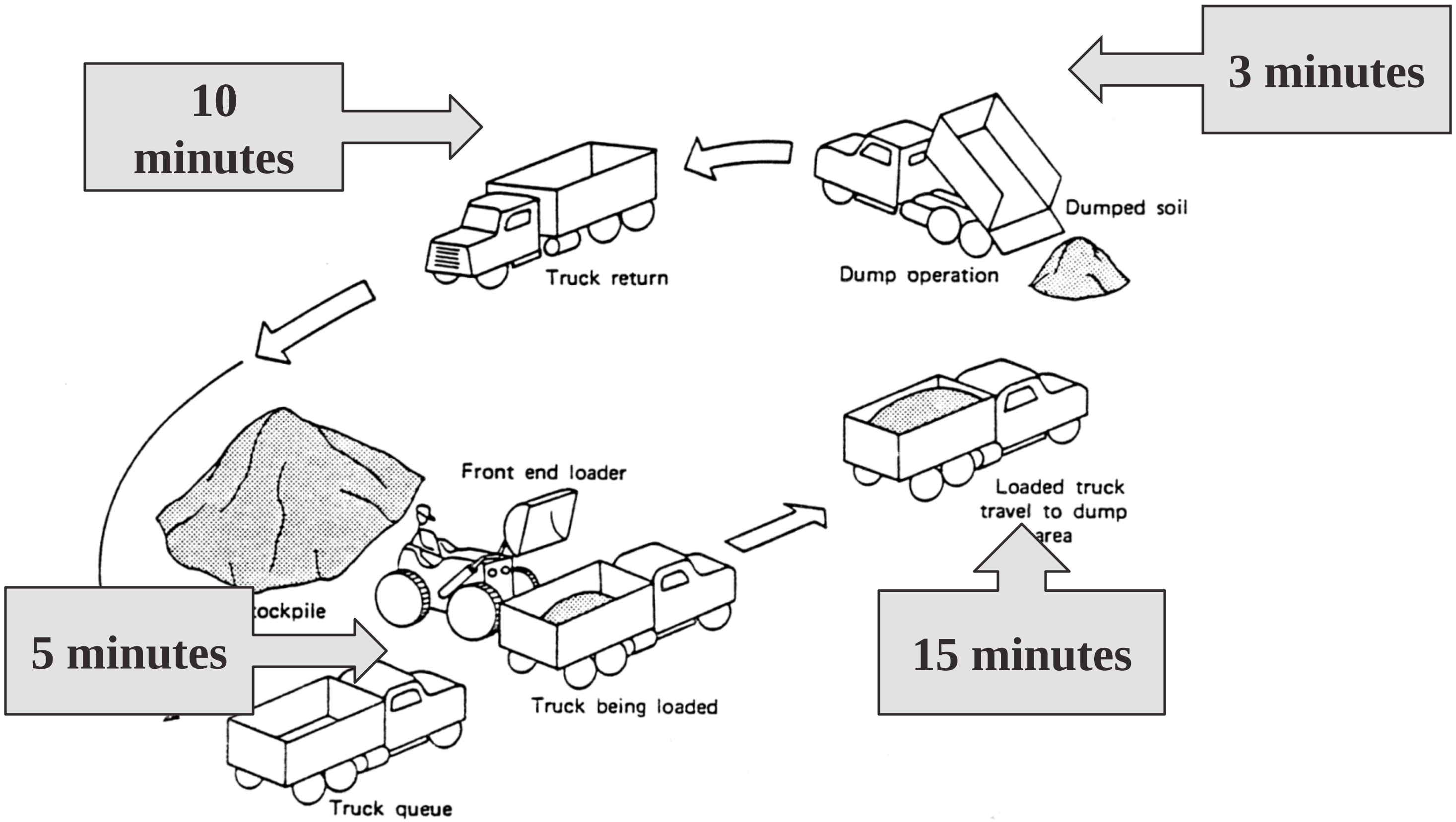


**COMPOSITE:** Container element for sub-models. Used to add hierarchy to a complex model.

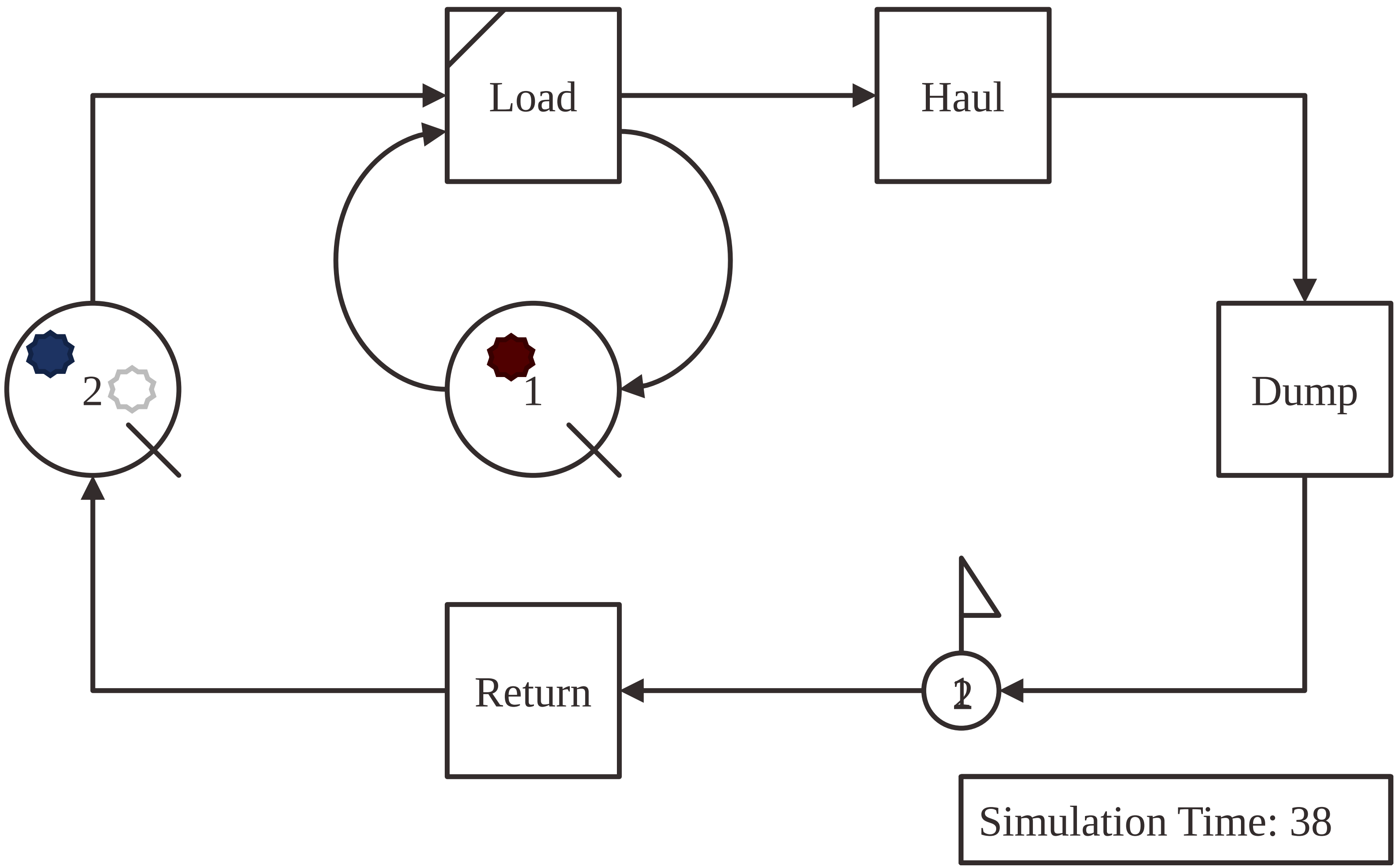



**SEND/RECEIVE:** Generates or consolidates entities.

# Example: Earthmoving



# Example: Earthmoving



# Example: Earthmoving

## When does this model terminate?

There are three ways for a simulation model to terminate:

1. When there are no more events to process.
2. When a maximum simulation time is reached;
   e.g. we simulate for 8 hours (480 minutes).
3. When the simulation is explicitly halted;
   e.g. the counter could terminate the simulation after it has observed 100 trucks pass.

# In-class Exercise

**Please refer to the assignment on ETL**

# Research Methodology (1)
# Introduction to Simphony

# How Does the General Template Differ from CYCLONE?

- Entities have attributes.
- Sophisticated resource modelling.
- Custom statistics.
- User-written code.
- Continuous modelling.

# Entity Attributes

## Local Attributes

| Attribute | Description |
|---|---|
| LN($i$) | Integer attribute $i$. |
| LX($i$) | Floating point attribute $i$. |
| LS($i$) | Text attribute $i$. |
| LV($a$) | Vector entry $a$. |
| LM($a,b$) | Matrix entry $(a,b)$. |

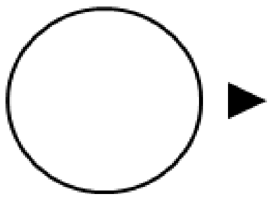
- Private to each entity.
- Each entity has its own set.
- A change made by one entity is not visible to another.

## Global Attributes

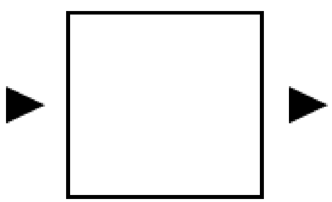
| Attribute | Description |
|---|---|
| GN($i$) | Integer attribute $i$. |
| GX($i$) | Floating point attribute $i$. |
| GS($i$) | Text attribute $i$. |
| GV($a$) | Vector entry $a$. |
| GM($a,b$) | Matrix entry $(a,b)$. |

- Shared between entities.
- Only one set in the model.
- A change made by one entity is visible to all others.

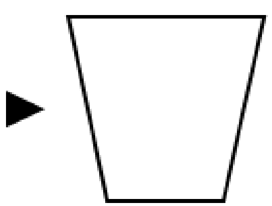
# General Template Elements I



**CREATE:** Introduces entities into the model.

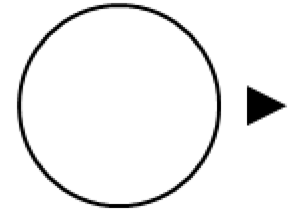


**TASK:** Represents an activity. Can be constrained or unconstrained.

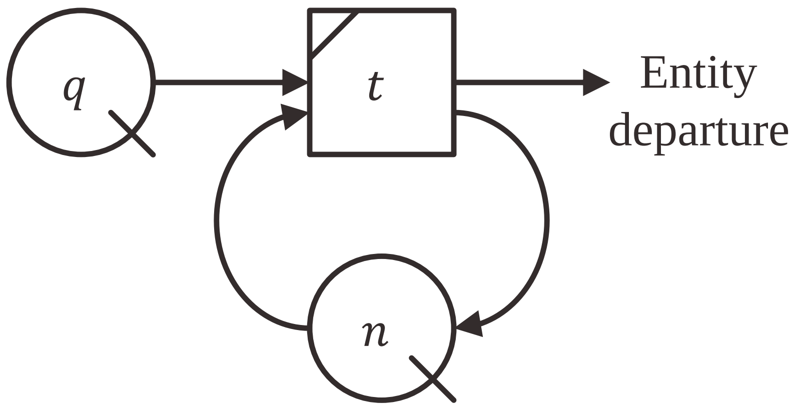


**DESTROY:** Removes entities from the model.

# The Create Element

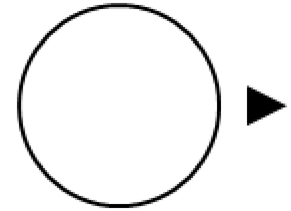


- Encapsulates the following CYCLONE model:




- The following values must be specified:
  - The total quantity to create $q$.
  - The inter-arrival time $t$ (possibly 0).
  - The quantity to create per interval $n$ (usually 1).

# The Create Element

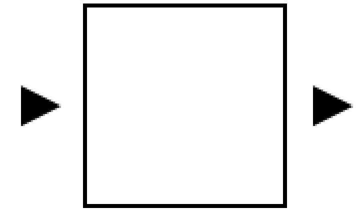


- The creation time of the first entity is specified independently of the inter-arrival time ($t$ in the previous slide).

- Local attributes can be initialized.

- By default, LX(0) is initialized to the simulation time at which the entity was created.

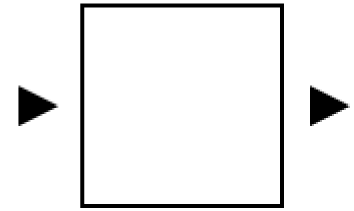
# The Task Element



## Unconstrained Task:

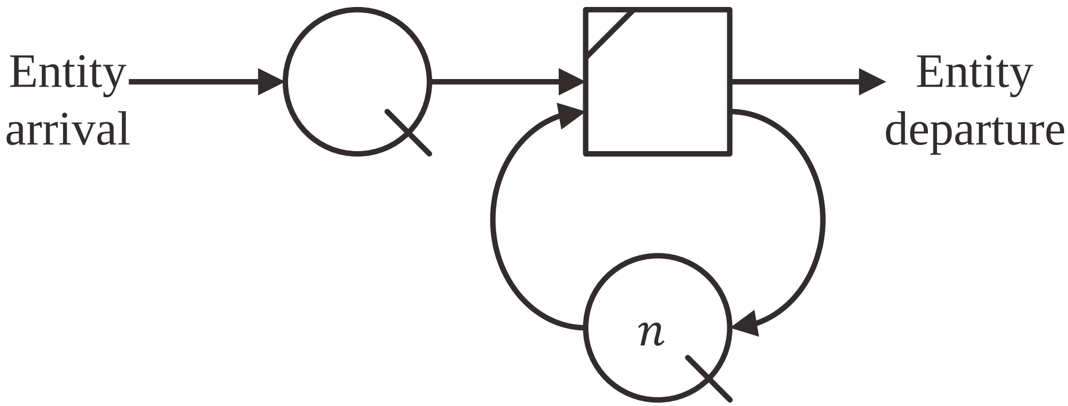
- Identical to the NORMAL element in CYCLONE.
- A duration must be specified.
- Entities are delayed by the specified duration.
- There is no constraint on the number of entities engaged in the activity.

# The Task Element



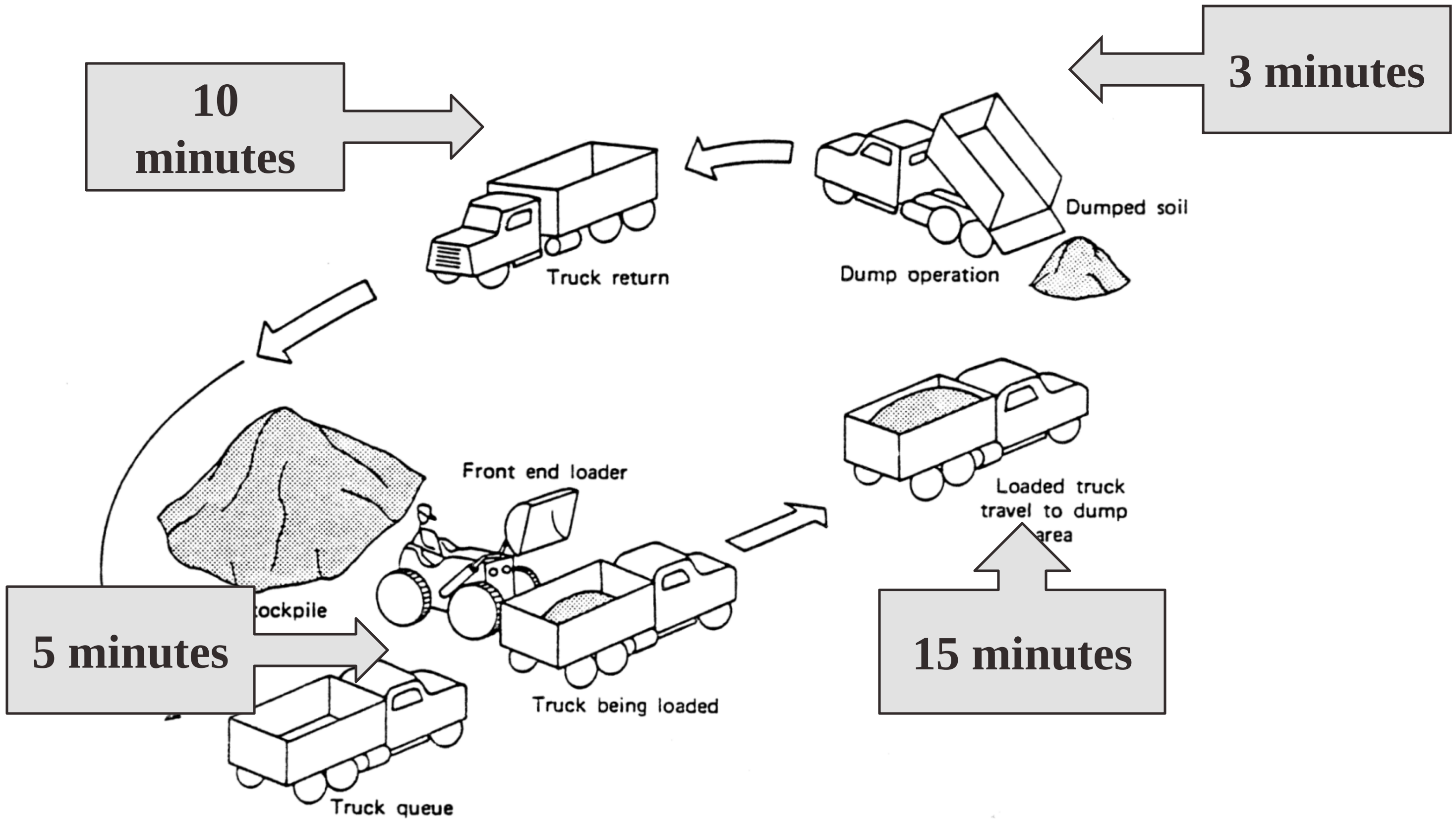
**Constrained Task:**

- Encapsulates the following CYCLONE model:

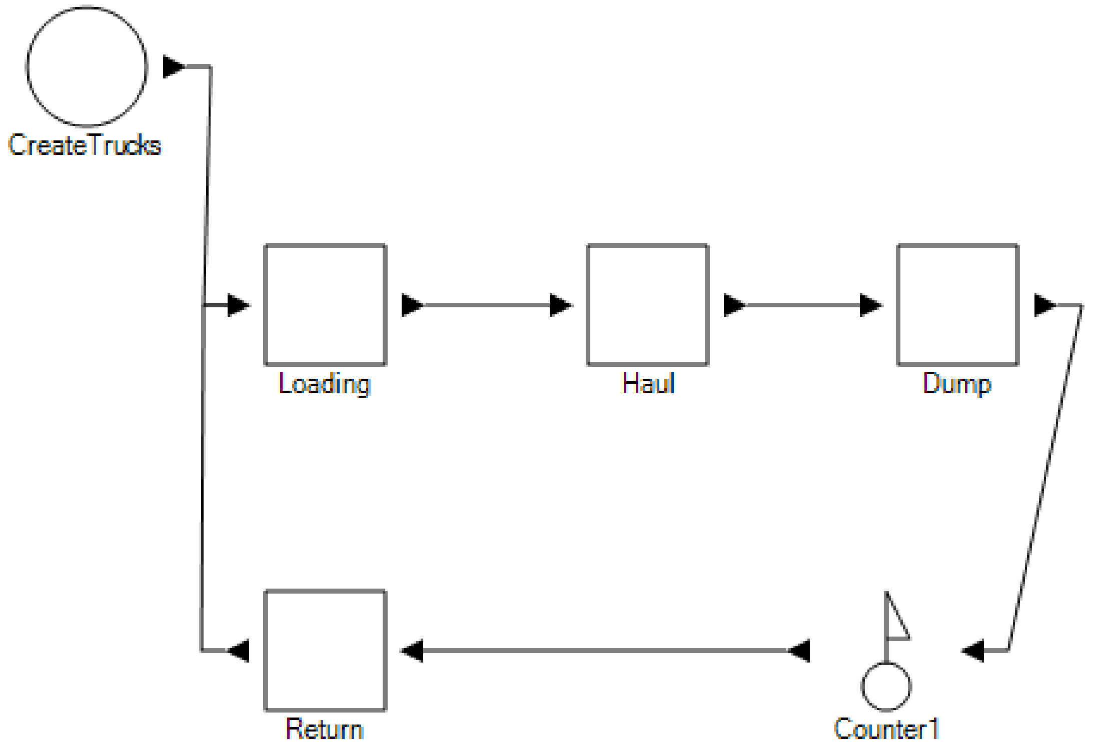



- The number of servers $n$ must be specified.

# Example: Earthmoving



# Example: Earthmoving



# In-class Discussion

(1) Which research/industry/life problems can be tackled by using Discrete Event Simulation (DES)? Why DES can be an appropriate tool for that problem?

:: Please think about one specific and simple problem and be ready to talk about it.

(2) If you design DES model for that problem, what would be input and output variables for your model? How would you collect data for input variables?

(3) How would you validate your model?

(4) (Optional) what will be the advantage of using DES over System Dynamics?