

# Empirical Modelling approach to a Hydro Power Plant

Charith I Fernando (0957728)  
The University Warwick  
c.i.fernando@warwick.ac.uk

## Abstract

This paper will build a model to demonstrate the work-flow of a hydro power plant. It discusses how Empirical Modelling concepts could help to create a mathematical model of this nature and what benefits the users could gain from it compared to other traditional software concepts. Here the paper builds a hydro power plant model to estimate the monthly power generation and water outflow depending on the parameters which are used to build and demonstrate it. The parameters would be the ones which an engineer would look at when building a hydro power plant such as the reservoir capacity, location of the dam, monthly rain fall and water input to the reservoir. Tkeden will be used as the tool for development and EDEN, DoNaLD and SCOUT will be used as definitive notations. The paper will evaluate the usage of the notations and their usability in the current domain.

## 1 Introduction

The main goal of this paper is to give an understanding to the readers that Empirical Modelling (EM) principles could be used along with modelling tools to build a mathematical model of an empirical study. It will discuss building of a Hydro Power Plants (HPP) to show the application of EM principles.

Hydro power is "the production of power through use of the gravitational force of falling or flowing water", Wikipedia. During this paper the author will be only taking into consideration the gravitational force of falling water.

Why did the author chose to apply the EM principles into HPP? This is especially because of the limited knowledge of the author in this area at the time of starting the paper. Which helps to show how iterative requirements are adapted into the system. EM is a way of modelling real life experience in computerized models which gives the user a better understanding of the subject and provides the ability to manipulate it along with the growth of experience. This is one of the advantages of building models on EM tools. After the current model is explained, the section on further enhancements will show how this is possible.

In this paper the next section on EM concepts and application will discuss EM concepts in detail and how it relates to building a model of a Hydro Power Plant. This will be followed by an overview of various EM tools and notations available. Section 4 will show how the model has being developed followed by an evaluation on the tools used to build the model.

Section 5 will describe further enhancements to this model followed by the conclusion and the acknowledgement.

## 2 EM Concepts and Application

In this section the author will discuss the definitions of EM concepts, how they are used by previous work and how they are applied to the current context of building the model for the HPP.

In "Uncovering Empirical Modelling" by King (2006), he discusses the "State-as-experienced" as the most fundamental concept of EM. He describes that the personal experience helps to build something in mind when a scenario is given. This is explained by taking the example of two glass tumblers, where he shows what kind of different interpretations there can be. In the Hydro Power Plant modelling project, the author sees the HPP with limited knowledge and it will be different to an experienced engineer's view or the state. The state understood by a person of a given scenario is defined as a construal in EM principles, King (2006). A construal is further described as a model of an understanding of a situation.

The construal which the author has on modelling the HPP is shown in figure 1. Here the author see a reservoir which is built by building a dam across a river and a power station below it with the turbines and the generators. In this paper, the storage of the generated electricity or how it travels will not be taken into consideration. The scope of the paper will only consider the factors shown in figure 1 .

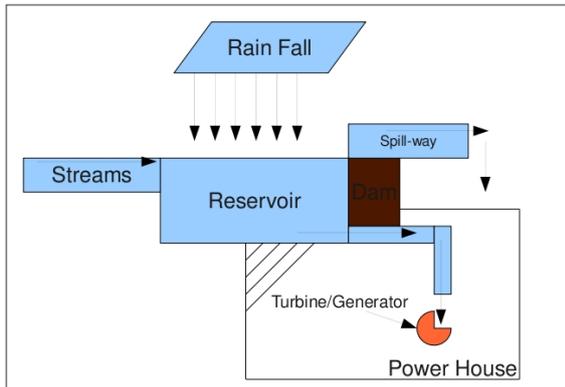


Figure 1: Construal of the HPP at design stage

A construal is developed by taking three more EM concepts into consideration, which are observables, dependencies and agency. As shown in King (2006), observables and the dependencies are closely related to each other. He explains that an observable is "something which has a value or status to which an identity can be ascribed", King (2006) and that a dependency is the relationship between these observables. An agency is something that may affect these observables to change.

The following is a list of observables initially identified in the system along with their relationships which depicts the dependencies. The formulas show how each value is derived.

- $\text{PowerGenerated} = \text{Head} \times \text{FlowRate} \times \text{Gravity}$  (Watts)
- $\text{ReservoirCapacity} = \text{Width} \times \text{Height} \times \text{Depth}$  ( $\text{M}^3$ )
- MonthlyRainFall (Litres)
- MonthlyStreamInput (Litres)
- $\text{MonthlyReservoirContent} = \text{MonthlyRainFall} + \text{MonthlyStreamInout}$  (Litres)
- $\text{SpillWay} = (\text{MonthlyRainFall} + \text{MonthlyStreamInput}) - (\text{ReservoirCapacity} * 1000)$  Litres

The agents will be the factors affecting the change of these observables. "An agent is an entity in the domain being modelled that is perceived as capable of initiating state-change", EM-Group. In the current domain the monthly rain fall and the monthly inputs from the streams are categorized as agents, because these affects the water overflowing from the reservoir and the power generated at the plant.

### 3 EM Tools and Notations

As described in King (2006), in EM development, tkeden has been the principal tool used. EDEN (Evaluator of DEfinitive Notation) supports many other definitive notations for various purposes such as DoNaLD (Definitive Notation for Line Drawing) for line drawing, SCOUT for (Screen LayOUT), ARCA for diagrams, vertices and indices, SASAMI for 3D modelling and EDDI for relational database tables and views. These definitive notations are available in the latest version of tkeden. Apart from the above mentioned notations, there is another notation called DOSTE which is being developed at the University of Warwick. This notation is mainly based on an object based framework.

In the current paper the author will be mainly using EDEN, DoNaLD and SCOUT for modelling the HPP.

#### 3.1 LSD Notation

LSD Notation is built for Domain analysis and descriptions. When understanding an existing EM model, it won't be clear of the interpretations the modeller had in mind when building it EM-Lecture-Notes. Hence a notation such as this will help readers understand a model already created. The notes on LSD explains that "An LSD account is orientated towards understanding the interaction between agents in a system" EM-Lecture-Notes. It further describes the four kinds of observables it has which are state, oracle, handle and derivate. In applying this notation to the HPP model which the author has in mind will be as follows. Here the author has taken the Input from Water streams and the Rain as agents. In this paper, the user who may change parameters manually in the system will not be considered as an agent.

**AGENT** MonthlyRainFall() // Monthly Rain fall Agent/Observable

**state**

*integer* current\_month = 1 // Jan

**handle**

*integer* Jan\_rain = 100 // Rain fall of January

*integer* Feb\_rain = 200 // Rain fall of February

:

:

**AGENT** MonthlyStreamInput() // Monthly Water input from streams

**state**

*integer* current\_month = 1 // Jan

**handle**

*integer* Jan\_input = 100 // January Stream input

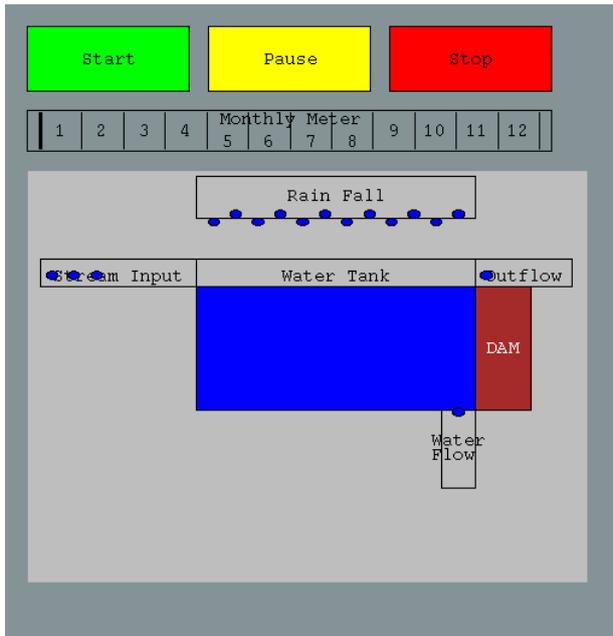


Figure 2: Hydro Power Plant Model

```
integer Feb_input = 200 // February Stream input
:
```

## 4 Hydro Power Plant Model

The model which is built in the current paper will mainly focus on how much power will be generated from the available water at a given period of time. The period is taken in months and the water availability is calculated by adding up the monthly rain fall and the monthly input from the streams to the current content.

The following are the list of observables in the system:

- Tank\_Width : Integer
- Tank\_Height : Integer
- Tank\_Depth : Integer
- Monthly\_RainFall : List with 12 elements
- Monthly\_StreamInput : List with 12 elements
- Monthly\_FlowRate : List with 12 elements, flow rate shows the water intake to the generators decided by the engineers at each month.

- Pipe\_Head : The distance from the water falling point to the turbines
- Gravity : Floating Point Constant
- Tank\_Content : Integer
- Current\_Month : Integer

As shown in Figure 2, there is a monthly meter which will show the animation of which month the model is evaluating at a certain given time. It shows streams connected (combined as one input), rain fall, water overflow output and the power generation output flow. The water tank simulates how much water is available in the tank initially and how it changes when the water is been used for power generation and is added by monthly inputs. 'Monthly\_FlowRate' observable is used here to check how much water is flowing out each month decided by the engineers. Rain and input streams are only simulated if the user has specified that it is going to rain in a given month or if monthly stream input is specified respectively. If at any given month the capacity of the tank is exceeded by the content of the water, it makes the water overflow. Hence the engineers gets an idea here when the gates should be opened to let go the excess water at a given time according to the inputs given by the engineer.

The figure 2 has three main buttons to control the simulation, Start, Pause and Stop. Start button will start the simulation, Pause will pause it at a given time and Stop button will stop the simulation and let the user start the 12 month cycle again with the current state of the model.

At the end of the simulation of the model the system gives out how much power is generated with the available water in the tank and gives out a monthly overflow table which highlights when the gates of the dam should be opened to make the extra water flow out. Apart from giving these information at the end, while the simulation is in progress it gives each month's power generation as an when it is calculated.

During the development of this model, SCOUT was used to make the windows to display the model. DONALD was used to draw the components of the model. As an example, the water tank, rain drops and dam are some of the components drawn using shapes available. While SCOUT and DONALD was used to draw the model on the screen EDEN notation was used to maintain the observables and the dependencies in the system.

## 4.1 Changing Observable (Agency)

The current model created does not support an interactive way of changing the observable values. But EM tools are powerful to take inputs from the users while the simulation is in progress. The monthly rain input and the stream in flow are represented as two lists in the model. Hence the users can use the code segments below to change the observables of the model.

Please note that '%eden' notation should be used for these. The numbers given here for the array can be changed and applied to the model at any time which would change the behaviour of it.

```
%eden  
rain_fall = [120, 240, 1024, 240, 202, 420, 120,  
4456730, 6785120, 530, 530, 250];  
stream_input = [120, 420, 440, 1110, 2220, 304, 0, 0,  
530, 120, 330, 5550];
```

Apart from the rain and the stream inputs, the users can change the dimensions of the tank. This is not reflected by a change in the size of the tank in the model. But the size change will be shown by adjusting the water level of the tank because now it will be able to hold more or less water according to the given values.

## 4.2 How is EM tools and notations useful here?

The author only aimed to show the monthly outflow of water and the power generated by the model for the given properties. The model that has been created satisfies this aim, hence this model can be defined as completed. The power of EM tools comes into play when this model needs to be extended with more complex dependencies and results. The section on "Further Enhancements", a later section will describe what other criteria could have been included in this model and how easy it is to do so with the tools currently used. Another advantage of using such tools is the ability of it to interpret statements immediately. Whenever the user feels like changing a statement, dependency or a function implementation, the programmer just have to write the code and apply it.

As a simple example, if the user wants to start the process from month 4, the current\_month observable needs to be changed to 4. As soon as you apply the change it will be reflected in the model. And when you start the model at that point, it start processing values from the month April.

The model in Figure 2 could have been easily built by using a programming language such as Java, but

changing the model according to how the user gathers knowledge about the domain is a tedious task. This is due to the fact that a program in Java needs to be compiled and run before results are seen and there is no way to save the status of a program at a given time and change values of variables at runtime. But in EM tools, it allows the program to stop the process at any given time and change the values of observables so that it will behave in a different way from that point onwards. Using EM tools is similar to using a Debug option on a programming language like Java. But there are some limitations in a debug mode where the user cannot declare new observables at runtime. It will not be fair for EM tools to be compared with high end modelling tools. It should be observed that the power of EM tools are on the flexibility it gives in changing the model while the developer or user gains knowledge which should be taken into consideration comparing with other modelling tools.

## 5 Evaluation of EM Tools and Notations

In this project, the model development was done using SCOUT, DONALD and EDEN notations. From the available notations in EM, SCOUT is used to create screen layouts. This notation provides features such as creating sensitive screens which can capture mouse and keyboard interactions. In this project SCOUT notation has been used to build the layout in different components (Windows) and display them on a screen. The buttons in the model are separate windows which have different background colours. The Monthly meter and the main model with the animations are also two different windows. Buttons are represented as separate components or windows to capture the mouse interactions with them separately. Monthly meter and the main model have been divided into two components for the clarity of the program. Models should be divided into several windows for clarity of the model according to whether they have any dependency over another window or not.

DONALD, on the other hand provide the necessary notation to draw 2-d lines along with some geometric functions. In the current project, this notation has been used to draw all the shapes on the screen. It also defines some observables which will be used by EDEN to manipulate the locations of the components and to animate the model when it is started.

EDEN does most of the work on the animation. The monthly meter animates the time cycle where as the main model animates the rain fall, stream input-

output and power generation. All of these animations are done by movement of the shapes. For the animation, eden clock is been used to identify the moving time. Monthly meter's starting point and ending point denotes the start and the end of the animation and each line denotes the boundaries of the months. Eden procedures has been used to monitor observables. When the time is ticked by the eden clock it moves the line of the monthly meter animating that the time is moving. Everything else in the main model and all calculations depends on the movements of this line. The animation is done by changing the observables which defines the 'X' or 'Y' coordinates of the shapes. Eden definitions of dependencies are done to calculate power generation. As an example, the eden definition calculates the power generated per month as soon as the current month has been changed.

## 6 Further Enhancements to the model

There is a wide range of enhancements which could be applied to this project. The model was built just by the construal which the author had about a hydro power plant. Going beyond this, there is a vast area in this domain which can be explored so that this model can be enhanced. This section will be dedicated to discuss few example of such enhancements and to show how it is possible to do so since the modelling is done on EM notations.

When the rain fall is given in to the system it does not take into consideration the rain which would fall on the ground around the tank. The current model assumes that every drop of water is fallen in to the tank, which is not practical. A possible enhancement is to find out about the soil condition of the nearby grounds to see how much water it absorbs and how much will be drained into streams around. At the same time the stream input that is currently specified by the user can be also calculated according to the rain fall. In the project, only the current geographical location's rain fall is taken into consideration. But if the user can enhance this by building a formula which will calculate the stream input from each region according to their rain fall in those areas we could get an accurate reading on the stream input to the reservoir which will also depend on the rain fall values inserted.

Another enhancement that this model could accommodate is to take the surface level evaporation of water. For this the velocity of air above the water, the humidity ratio of the air and the heat can be taken

into consideration.

This project only aims to deliver the power generation outputs and the monthly overflow of water. Since we have a tank with proper dimensions we could also calculate the water pressure in the tank which can be useful to identify the strength of the dam wall. Current knowledge of the author is insufficient to make a formula of this nature, but if an engineer supports this model, it can be improved to include that formula to calculate the expected strength of the dam wall. Another hard coded input which the users has to give in this model is the water flow rate travelling to the turbines. This can be represented by a formula which will find the flow rate according to the properties of the channel that is used transfer water to the turbines

## 7 Conclusion

The main objective of this paper was to build a model of a Hydro Power Plant using EM tools and notations. This was mainly to show that using EM concepts to build a mathematical model of this nature is beneficial given the fact that how the requirements may change in the future and how convenient it would be to reflect these requirement changes. Especially to show how a model could be built by a person with limited knowledge in the domain and not worrying about the actual scenario in the domain. The power of EM lies in the aspect of its flexibility of changing the model immediately as an when knowledge is acquired compared to normal programming languages where it will be another cycle of analysis and design. This paper has discussed the EM concepts and demonstrated how they are applied to a working model which has been built by using EM tools and notations. It also discusses the advantages of using EM concepts to build models of this nature compared to the use of a high-level language such as Java. Finally the paper gives some enhancements that can be built on top of the current model with some additional professional information.

## Acknowledgements

Herewith, I would like to appreciate the support rendered by Dr M. Beynon and Dr. S. Russ in giving me an insight into the Empirical Modelling concepts and the feedback given along with the development of the paper and the model. Also appreciate the guidance given by Nicolas Pope for installing the development tools.

## References

EM-Group. A glossary for empirical modelling.  
<http://www2.warwick.ac.uk/fac/sci/dcs/research/em/intro/glossary/>.  
Accessed 21st December 2009.

EM-Lecture-Notes. The lsd notation  
for domain analysis and description.  
<http://www2.warwick.ac.uk/fac/sci/dcs/research/em/teaching/cs405-0708/concsys/concsyslecture4>. Accessed 21st  
December 2009.

K G King. Uncovering empirical modelling. Master's  
thesis, The University of Warwick, 2006.

Wikipedia. Hydro electricity.  
<http://en.wikipedia.org/wiki/Hydroelectricity>.  
Accessed 20th December 2009.