# Implement and profiling the MPTT technique in WordPress

## Software Architecture Document

**Version 1.0**

# Revision History

| Date | Version | Description | Author |
| --- | --- | --- | --- |
| 22/10/2009 | 1.0 | Initial | Ajith |
| | | | |
| | | | |
| | | | |

# Table of Contents

# Software Architecture Document

## 1. Introduction

### 1.1 Purpose

This document provides a comprehensive architectural overview of the system, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions which have been made on the system.

### 1.2 Scope

This document gives an architectural overview of MPTT technique in wordpress. And most of the things are directly derived from wordpress architecture because the part that I selected as a project is a small module compare to whole wordpress architecture.
Here I used 4+1 view model which is already available in Software Architecture Document template that provided by RUP.

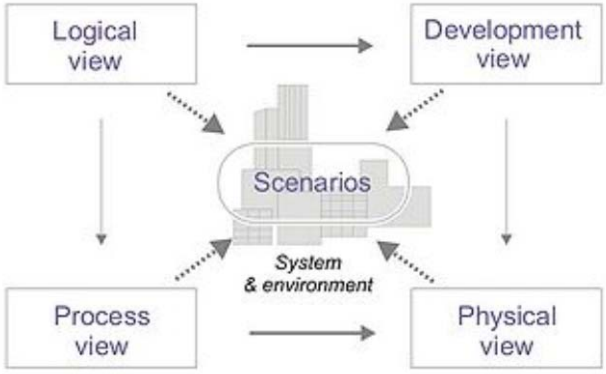
### 1.3 References

WordPress API
RUP document
World Wide Web
http://gsoc2009wp.wordpress.com/daniel-mptt/
http://www.ts.mah.se/RUP/wyliecollegeexample/courseregistrationproject/artifacts/analysisndesign/sadoc.htm

## 2. Architectural Representation

This document represents the architecture in a series of views; use case view, logical view and physical view. Use case view is represented using UML. If I consider the architecture of my part, it is nothing compares to wordpress architecture and it will be same as the wordpress architecture. So the architecture views that I used here are derived from wordpress architecture.

In this document designed according to the 4+1 Architecture view model,



# 3. Architectural Goals and Constraints

### 3.1 Architectural Goals

According to the hierarchical data available in database make an arrangement using MPTT technique that will sort hierarchical data and show it quickly. So when we need the data we can get it quickly. And this technique reduces the traffic on the server and utilizes the server performance.

### 3.2 Architectural Constraints

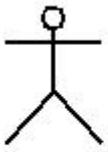
- Time
- Accessibility of the database server.
- Knowledge of the particular language that I 'm going to use.
- Speed of the server where the data stored.

# 4. Use-Case View

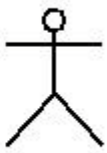
The Use Case View is important input to the selection of the set of scenarios and/or use cases that are the focus of iteration. It describes the set of scenarios and/or use cases that represent some significant, central functionality. It also describes the set of scenarios and/or use cases that have a substantial architectural coverage (that exercise many architectural elements) or that stress or illustrate a specific, delicate point of the architecture.

### 4.1 Use-Case Realizations

Here I used two use-case diagrams which are already available in software requirement document.

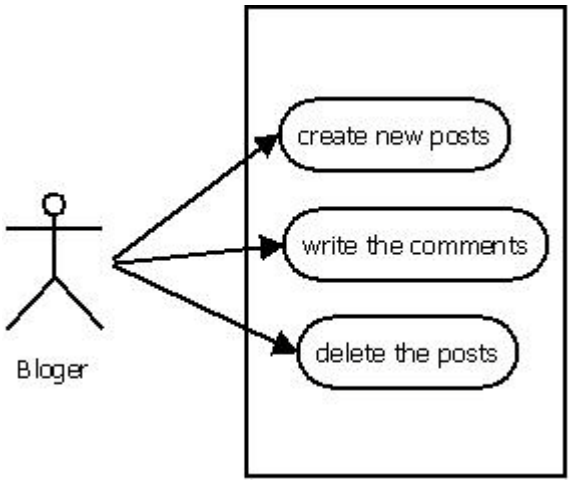


Blogget/Visitor



Developer

#### *4.1.1 Blogger*

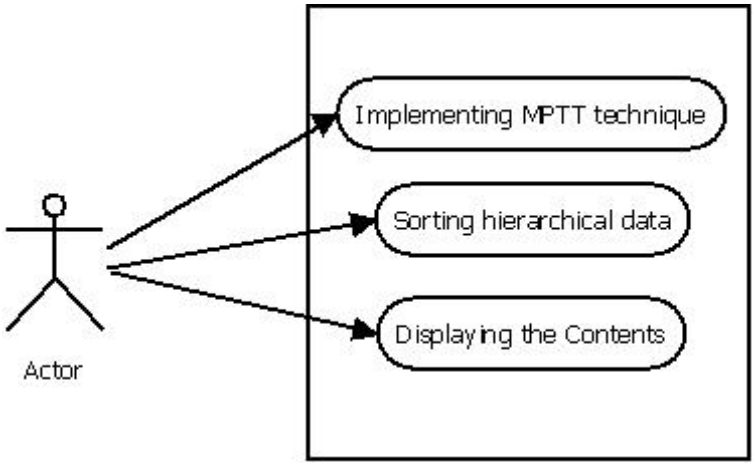



- Creating new posts
  This use case occurs when the blogger create new post and publish it.

- Write the comments
  This use case occurs when the blogger or guest put some comments under the post.

- Delete the posts
  This use case occurs when the blogger delete the post which is written by blogger and only the blogger has the privilege to delete it.

#### *4.1.2 Developer*



- Implementing MPTT technique
  this use case occurs when we need the fast performance that provide quick sorting mechanism for hierarchical data.

- Sorting hierarchical data
  this use case occurs when the hierarchical data sorted using MPTT technique.

- Displaying the contents
  this use case occurs when the hierarchical data displayed as a web page and the MPTT technique provides a fast profiling mechanism to display the contents.

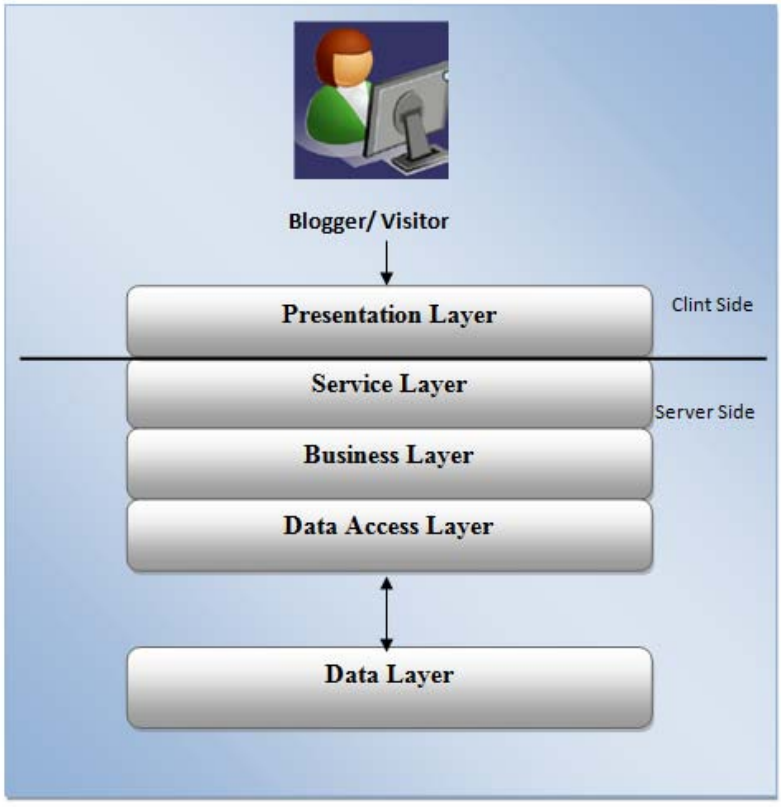
## 5. Logical View

It provides a basis for understanding the structure and organization of the design of the system.

### 5.1 Overview

This is concerned with the functionalities which are provided by the system to the end user.

### 5.2 Architecturally Significant Design Packages



- Presentation view
  it provide a function that used to make a communication between end user and system. Through this layer the end user able to put the data and get the data. For that purpose the end user will use mouse, key board and monitors.

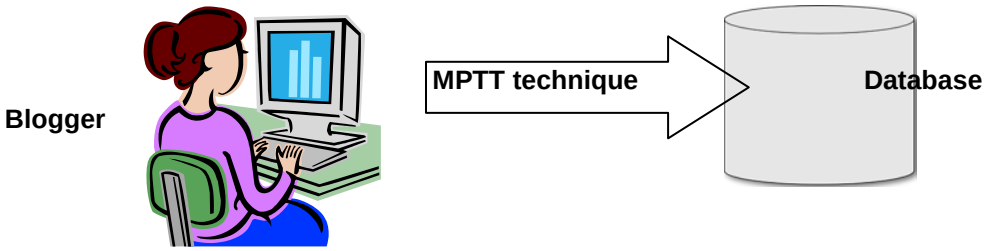
- Business layer
  business logic layer provides support for the application of specific business processes, as well as, on the application and implementation of trade rules and data integrity.

- Data Access Layer
  data access layer provides support for continued access to data and in conjunction with a relational database and WordPress

- Data Layer
  this layer deals with the data that are available in database like storing and reading the data from database.

## 6. Physical View


## 7. Data View



## 8. Size and Performance

When the no of users, no of posts and comments in a page increases, the time take to display the all contents also increases with the current technique that used to sort and profile the hierarchical data in wordpress database. Usually the time taken to perform a function increase with the size of the data, but providing a better solution (implementing MPTT technique) may reduce the increasing rate of time.

- Response time

  When the MPTT technique applied to the system, the response time taken to displaying or sorting the hierarchical data is become less compare to other techniques.

- Throughput

  The MPTT technique shows more posts for a given time compare to other technique. The time taken to sort the hierarchical data is also become less

- Capacity

  How many users visit to a page (pages in a particular blog) is varying according to the blogs category or demand. The system must have the ability to handle the capacity of the request.

- Resource utilization

  By performing this technique, the resources can be utilized, because it provides a fast and better solution to sort and display the data.

## 9. Quality

- The developed function won't make harmful things inside the main system(wordpress)
- Providing fastest performance compare to previous technique.
- Reduce the work load on the server – usually it happens when the user visit to the page every time.