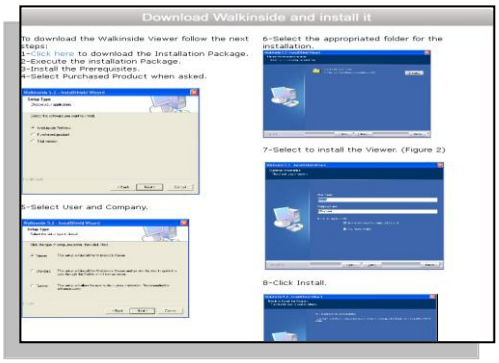
Model Converter Service

Rich Model to Transactional Model Conversion

Executive Summary

This case study is all about conversion of Rich Models (rich with CAD / CAM data) to Transactional Models (a light weight version of Rich model file). Our Belgium client was looking for a software development firm who can provide a solution wherein, conversion of Rich Models (heavy files) to its corresponding light weight version is done easily and at a faster pace keeping in mind that the light weight version to be created should contain the same behavior as that of the Rich Model file.



When Mindfire Solutions was approached for this project, they analyzed the client's business and technical requirements before proposing a solution. Mindfire finally developed a new standard called Transactional Model that met client's objective. There were few challenges faced during the course of this project, but the team did well to produce a very robust multi-threaded windows service to convert Rich Model to Transactional Model. The current model has SQL Server 2005 as the database. But, as required by the client the model also has Oracle serving as the backend for users who would prefer Oracle rather than SQL Server. The project also involved some SharePoint development work of creating a log-in page.

About our Client

Client Solution Provider to Heavy Industry | **Location** Belgium | **Industry** Heavy Industry

Business Situation

In heavy industries engineering users often create the model / blue print of the sites and they need software with rich functionality to process engineering data to prepare end user information. "Rich Model" is a standard developed by our client that provides engineering user a facile way to create, read, update and delete engineering data and present them efficaciously.

The client was curious in having something in place that would, be of lighter weight and have the same behavior as that of the Rich Model, with a faster approach of converting the heavy weight file to its lighter version. The requirement for such a solution was required due to the problems stated below:

- As Rich model files are self-contained, they are heavy-weight in nature (Millions of engineering data are present for a part of Rich Model)
- All engineering data reside in local file system, hence sharing data among multiple users was tedious
- Multiple users can't work on same model at the same time

The client also had couple of other requirements. They required a log-in page to be created for users in SharePoint. Also, they needed Oracle as an alternate database apart from SQL Server so that users can have their own choice of database.

Solution Details

The Mindfire Solution

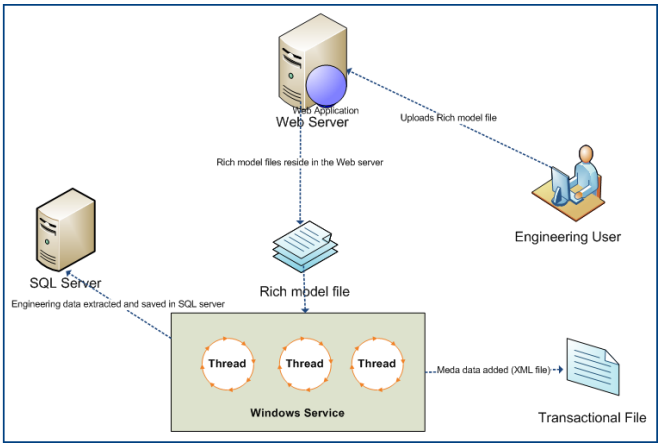
After a long search for the right development firm, the client came up to Mindfire in search of a solution. Before proposing a solution to the client, Mindfire's team analyzed the clients business and technical requirements. The final proposal from Mindfire turned out to be satisfactory and fruitful.

Here is the outcome of the solution:

- A new standard developed i.e. Transactional Model
- A multi-threaded Windows Service developed to recognize Rich Models and converting them into Transactional Files
- With this new standard all engineering data are extracted and pushed into a centralized SQL server database, thereby making Transactional model lighter than Rich model and to help sharing data among multiple users
- Meta data are added to the Transactional Model to effective use of centralized data source

The project saw a great deal of challenges. Some of the major ones have been listed below:

- Correctly identifying engineering data in Rich Model
- Mapping of engineering data to SQL server data
- Developing a robust multi-threaded windows service to convert Rich Model to Transactional File



Achievements

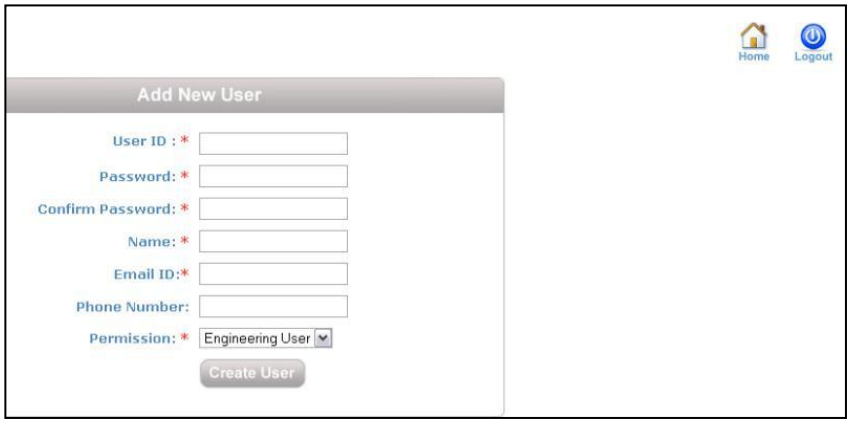
The new standard helped defining the new properties of the Transactional Model files and made it very easy for applications accessing Transactional Model files to maintain uniformity. The other major achievement of the project was the Windows service with the multi-threading capability which was robust enough to convert the Rich models to Transactional model, resulted in a high-speed process for conversion.

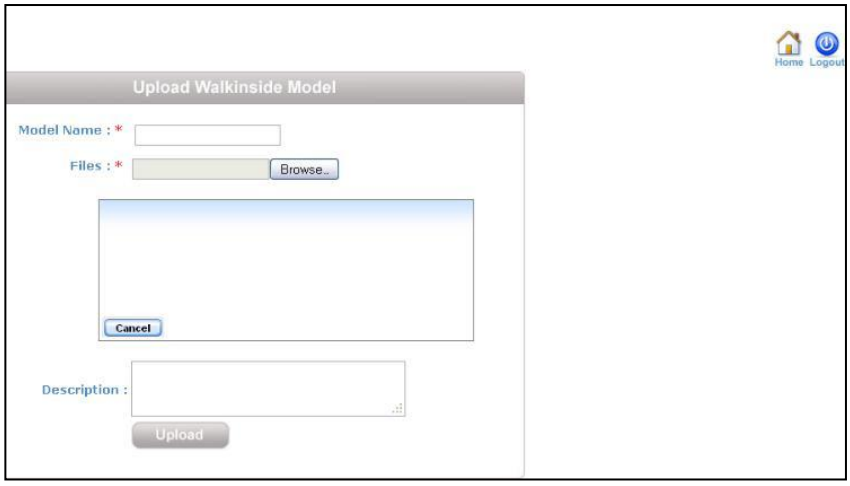
Technologies

.Net Framework 2.0, C#, SQL Server 2005, Oracle, SharePoint 2007

Final Results

Software System





Customer Benefits

On developing the product, client was benefitted with numerous ways. Some of these benefits are listed below:

- The client started using the Windows service which was robust and faster to handle conversion of Rich Model to Transactional Model.
- Achieved multi-user access to the same model.
- Sharing models with their customer made easy.

Future relationship

This has resulted in with the Client retaining Mindfire to work on their other set of products as well. Also the client is considering starting on some new product developments as well to be done exclusively by the Mindfire team in India.