Model Driven Architecture (MDA) a Myth or Reality: A Case Study-OptimalJ

II. Description of the project:
Software Development has not achieved any significant improvement on abstraction and hence productivity since the appearance of Third Generation Language. Indeed there is not much difference in productivity between old languages like Basic or FORTRAN and modern languages like C Sharp, C++ Java. All of them are based on coding concepts. However, if we based our development on higher level concepts like UML models a higher levels of abstraction and productivity will be achieved.

Several approaches are being developed lately to promote model as a first class citizen in the software development arena (as apposed to code) one of them Model Driven Architecture (MDA) is fostered by the Object Management Group known by its CORBA specification.

The Model-Driven Architecture starts with the well-known and long established idea of separating the specification of the operation of a system from the details of the way that system uses the capabilities of its platform.
MDA provides an approach for, and enables tools to be provided for:  
- specifying a system independently of the platform that supports it,  
- specifying platforms,  
- choosing a particular platform for the system, and  
- transforming the system specification into one for a particular platform.

One of the best known MDA implementation is Compuware OptimalJ that is the technology we are going to use to explore by implementing a modern size library management system for the J2EE platform.

III. Expected outcome:
   a- Implementation of a moderate size Library Management System in J2EE Platform using OptimalJ.  
   b- A report describing the benefits and limitation or using this technology to developed the Library Management System  
   c- A set of metrics applied to the above project comparing with the same system being developed without any MDA support