

Abstract

The motto of this thesis is to achieve “*write once persist anywhere*” in distributed environment. Distributed applications development using model based methods naturally starts with functional requirements given by the application. The model capturing these requirements is then refined by taking more aspects of the Technology / Platforms into account. The Model Driven Architecture (MDA) has been introduced by the Object Management Group (OMG) in order to model systems independently of the (middleware) platform used. For this, Platform Independent Models (PIMs) and Platform Specific Models (PSMs) are defined in MDA. PIMs model a system independently of a set of possible target platforms and PSMs model a system in terms of a particular platform. When the transformation between PIMs and PSMs are well understood, the development of distributed applications can become less costly.

In order to adapt the MDA for software developments, some obstacles like acceptance of modeling for software development, Platform / Technology dependence of pervasive Services etc have to be removed. This work aims to promote the persistence service of pervasive service category from platform dependency to independent form with automated round trip engineering.

The first investigation is to form MDA based reverse engineering for legacy persistence service to construct platform independent persistence service model for a given application context. The second investigation is to devise tool specific persistence service design using Platform Independent Model, Platform Specific Model and mapping between them. The third investigation is based on the realization of tool chain for MDA based software development. MDA components are the best way

to realize tool interoperability in tool chain with reengineering, reverse engineering and round trip engineering aspects. This is validated by adding QoS aspect at various levels of Tool chain.

This study provides an experimental backdrop for the development and design approaches each holding lessons for persistence service design, its preservation and round trip engineering. Abstraction levels are necessary for the design of any pervasive service like persistence service. Tool dependency and past investment are two major deciding factors in using MDA for persistence service development. Implementation of experiments serves to validate the approach with encouraging results.