

RESEARCH PAPER

HOW WORKS DYNAMIC SOFTWARE PRODUCT LINE ENGINEERING

Amir Ansari

Software Engineer

CTB Solutions

Puchong, Selangor, Malaysia

Abstract: *There was a time when technology came from big companies and a large number of products were developed and gave the birth to a new world which was a technology world. Each product was unique and it was developed by its own company. They did not want to share anything with other companies because they did not have proper guidance that how to implement the existing products with a new one so users have to wait for long time to get new products because it took much time to develop each and every component of the product. Today technology has reached its peak and there are many existing products available which can be used by the developers to create new products. The time has changed now developers have learnt many things from the past and they can easily integrate old technologies to new one.*

The dynamic software product line engineering is playing a vital role in our lives which one is teaching that how two products can be integrated if they are loose coupling status after introducing of this, it has solved many problems and also saving much money and time. Now big companies are ready to give some small task to the small companies and can ask the time of delivering products once its deliver big companies can integrate with their products and user can get new technology's product in short time.

Key Terms: *Dynamic Software Product Line Engineering, Software Component, Reusability, Loose coupling, tight coupling etc.*

1. Introduction

Program is a collection of instructions and software is a collection of programs. It means software is a building block of instructions which includes the combination of different type of instructions and if the instructions change then software will be changed and again have to write thousands of line code. A Dynamic Software Product Line Engineering (DSPL) is a methodology where many developers or programmers are using the functionality, tools and techniques from another product at run time to making dynamic software (Mike Hinchey S. P., 2012). Developers are used some additional number line of code during making software because they know that in the future some changes will be needed. This is a good technique where many software companies are sharing their products features with other companies and whatever companies have requirements they can buy from another company. Now a days, open source technology is available which is heavily used by programmers because it provides many functions and tools which help to invent a new product. Sometime this technique fails because of the functionality of the new software does not allow existing software. So, sharing is possible only loose coupling software which has characteristics to make connection with software (Muthu Ramachandran , 2008). The concept of updating is based on this technique because same code reuse with some modification of code and users get new version software.

Now a days there are some small companies available which are developing small products and delivering the big companies. And these big companies are ready to integrate these products with existing products and deliver the project to the users in less time and they do not use only products but sometime they also use methods, components and architecture of another product (Mahdi Bashari, 2013).

They are not only limited to use these features at static level but also used during the run time to make dynamic software which can achieve changes at run time and also there is no need to write more code. Dynamic software Product line engineering has given a new direction in the field of technology and changed the developers 'mind. Developers are doing research on the existing technologies and how can they implement these technologies to build a new product and how can use these features at run time. In dynamic software, there is no need to write a more code in such a manner that it will be adaptable at run time. Dynamic product line engineering concepts is based every type of product which can share their features with another one. (Mike Hinchey S. P., 2012), (Mahdi Bashari, 2013).

2. Literature Review

2.1. Software component architecture

This is a technique which reuses the software components and makes new software with some additional features. The main concept of software component architecture is that how software can make coupling between the existing components and also check that coupling will be possible or not between to different products also collect the all the requirements to make it successful. Dynamic software product line engineering is a part of software component architecture (SCA) because here developers use existing products of components to make new products and also this methodology is used at run time (Muthu Ramachandran , 2008).

2.2. Features of Dynamic Software Product Line Engineering

In the present time, world's technologies have changed as well as users' mind also changed because they are using a lot of products and expecting new products which should be better than these existing products and they are thinking like a developer because they are also concerned with new products which should be unique and fast. Users are not only concerned about software but they also want dynamic software which should change accordingly when it's needed (Mahdi Bashari, 2013).

2.2.1. Domain of Dynamic Software Product Line Engineering

Software coupling is not possible with every components of software because sometimes it creates problems for other software and also do not allow any changes. There are only few products which have the capability to share the things with another one. A domain is defined “Components or methods which can be shared only for loose coupling” because they have property to share tools with other products (Muthu Ramachandran , 2008).

2.2.2. Reuse methodology using Dynamic Software Product Line Engineering (DSPL)

Dynamic software product line engineering concept is based on Software component architecture which is complete set of reusability concepts and tells about that how can apply one product feature to another one and after defining a domain it tells that how can reuse the components and this coupling will be possible between two products or not before inventing the new products. This domain is not only static product as well as dynamic product which can change their configuration at run time (Muthu Ramachandran , 2008).

2.2.3. Self-adaptive system

The most important characteristics of dynamic software product line engineering are self –adaptive which is similar to human being and can adapt some new features at run time without taking time. Some software changes are reflected at run time because software needs some additional work. Dynamic software is self-adaptive and they change their configuration according to the situation and demand of the product. Self-adaptive system is based on the two characteristics out of which the first one is the application logic, which only focuses the functionality of the application. The second one features is adaption logic, how it will adapt new features at run time (Mahdi Bashari, 2013).

During the process of making a software, if programmers want to make any changes at run time two things are very important, “when to adapt” and “how to adapt”. How to adapt, they already added some functionality in the application but when it will adapt this is the big problem for the developer because it depends on the individual software and their functionality. This is the dynamic variability of software (Mahdi Bashari, 2013).

2.2.4. Flexible variability mechanism

Flexibility is another most important feature of dynamic software product line engineering because if system is flexible then coupling is possible between the products and they can share their methods, tools and technique with software. This kind of methodology happens at run time because they are applicable to adapt any changes. Sometime this technique is fail because systems are not very flexible and they do not allow to implement any changes at run time it is important for a coupling system should be flexible (Mike Hinchey S. P., 2012), (Mahdi Bashari, 2013).

2.2.5. Dynamic reconfiguration

Static software do not have ability to change their configuration at run time because during making the software everything is already defined also it is expected that in the future this configuration will not change these are static software and can run on the desktop but dynamic software has ability to adapt any changes with new features and configuration at run time (Mahdi Bashari, 2013).

2.2.6. Motivation

This is an important step which motivates the developers to establish a connection between two products and use existing features at run time. It is not an easy task therefore developers should work under the guidance of the experts and who can motivate the developers. (Mahdi Bashari, 2013).

2.3. Goal of Dynamic Software Product Line Engineering

The aim of this project is that use the existing products and makes new products and provides these products to the users as soon as possible with good features.

2.4. Benefits of Dynamic Software Product Line Engineering

There are many advantages of these types of products but most important is that every product can be used to build new products if they have some characteristics to share the things. Developer saves time and also investors save their money to use the existing products and also use these characteristics at run time and make a dynamic product (Mahdi Bashari, 2013).

2.5. Disadvantages of Dynamic software Product Line Engineering

Every coin has two faces so dynamic software product line engineering also have some disadvantages. First of all this methodology is only applicable for loose coupling product so if the developers do not have proper knowledge about the products and start working on it definitely these products will be failed. Some investors also do not want to invest their money on this because they think it will be a failure due to lack of guidance. This same thing happens at run time when the software changes are required but old software does not accept any changes at run time then it creates problem for other software. Therefore, research about the same is the most important thing to implement this technique (Mahdi Bashari, 2013).

3. Research Design and Methodology

I. Software versus Dynamic Software

Dynamic software concept is similar to software but there are huge demands of the dynamic software which differentiate some features between the software.

a) Feature Comparison

Software	Dynamic Software
is Collection of many programs	is Collection of less programs
No changes at run time	Adapt new features at run time.
It works whatever program has fixed during implementation of software.	There is no more code require because at run time it is applicable to adapt new code.
It takes More time	It takes less time
It does not follow reuse concepts	Code can be used many times at run time with some additional features.

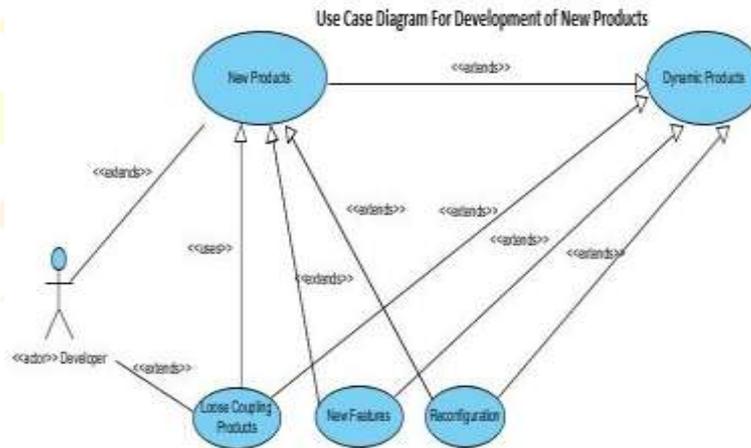
II. Requirements of Component

Dynamic software product line requirements are existing components of a technology which have capability to share the property with another products. It should be able to adapt a new architecture with some additional features at run time and products should have loose coupling concepts because only those products have capability to adapt new things (Mahdi Bashari, 2013).

III. Analysis and Planning

Dynamic software required changes at run time before making these types of software analysis is essential and this is also its most important feature because it can save the time or money both. First of all that one should know that system is ready to adapt new features at run time or not. Dynamic software product line engineering can take much time in analysis because here developers have to analyze more than one product and how can use these products at run time with new products (Mahdi Bashari, 2013) . Planning is the strategy to develop new products and which component should use first to start the work (Mahdi Bashari, 2013).

Use Case Diagram



From the above graph it can be seen that how the process will work to develop a new product. Dynamic software product line concepts tells us that only loose coupling products features can be used to make a new product but it is not possible that it adapts all the features at run time. This use case diagram displays each and every step.

4. Results and Discussion

I. Artificial Intelligence versus Dynamic Software Project

Artificial Intelligence is the technology which works like human being it knows very well about the user’s action. For example, game programming is based on artificial intelligence concepts because one side human beings are involved in the game and another side computer play the same and system knows that what is going on in human’s ‘mind and so the computer wants to oppose all the action of humans. In game programming, programmer writes more code and every step to play the game. Dynamic software is similar to the artificial but there are some differences between two. Dynamic software is a collection of some lines of code and programmer can provide other codes at run but in the desktop game it is not possible because everything is already fixed.

II. Server Side Technology

There are many existing technologies available which are server side and change themselves when they connected to the server and it provides many facilities whatever users want without taking much time and gave the accurate output. Server side technology is used in dynamic software product line engineering because without server no dynamic software can be implemented. Here developer provides some new features day by day to reuse the components of same products or other products.

III. Net beans

This is software which is a building block of the existing components and used for different purpose according to the products’ demand. There are many inbuilt components are available which provides services to make dynamic software.



IV. Servlet

This is a server side technology which is used to deploy the code on the server with the help of existing technology like NetBeans, Eclipse etc. Mostly it is used to deploy the website on the server with the combination of HTML.

V. HTML

It is a client side language and used to create a static page and output can be seen on your system and there is no need for a server but it behaves as a dynamic website when it makes a coupling with other components and code is deployed on the server. This is the best example of dynamic software product line engineering because client side language became server side and also can change itself when it will be needed.

VI. XML

This is a latest technology which is popular among developers because it has different roles as well as uses. Most of the important role is that it provides a service to make a coupling between two components and deploy it on a server within an existing technology. It is used to store data and transport the data. It is also used in Android development and also in web development to create a web.xml file to establish a connection between two technologies.

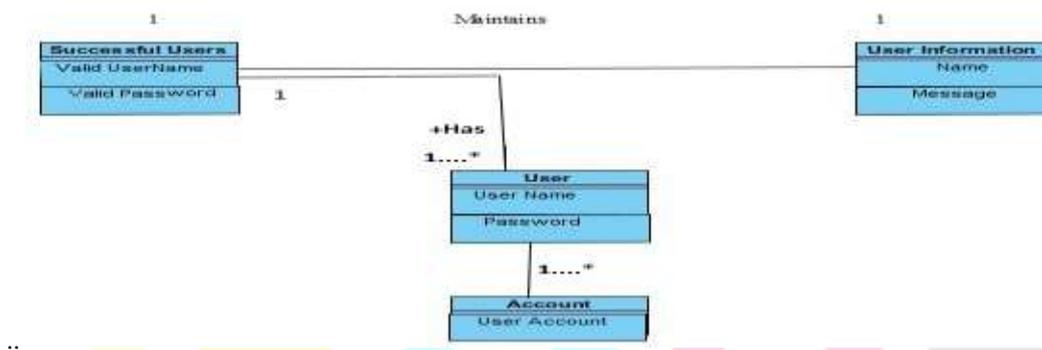
VII. Dynamic Product Lines Using NetBeans

The actual meaning of a dynamic software product line is that it reuses existing components and applies these features to a new software at run time. NetBeans is also a collection of many components and developers reuse these components according to their products' demand. There are several steps to complete this project and it will also prove that how a static page running on the server and how the answer is changing according to the users. This is a portal and only authorized users can access this account.

a. Class Diagram

It is a class component which shows that only valid users can access the information about the portal. It may be possible that there are many valid users who are working for the same organization.

Class Diagram For User Account



5. Conclusion

Dynamic software product line is a good methodology to reuse existing components to develop new software at run time and if existing product has capacity to adapt the new configuration at run time then it will change according to the requirements else it will fail. It is used to save the time of the developer and also save the money of the businessmen who are ready to invest money on the products. It also proves that every product has important only necessary is that how can use these product with another one.

REFERENCES

- [1] Mahdi Bashari. (2013, August 27th). Toronto: Laboratory for Systems, Software and Semantics (LS3).
- [2] Mike Hinchey, K. S. (2015). Self-Adaptive and Self-Organizing Systems. *Cambridge, MA, 21-25 September, 2015*, (p. 1).
- [3] Mike Hinchey, S. P. (2012, 10 10). *Building Dynamic Software Product Lines*. Retrieved 10 29, 2015, from <http://www.computer.org/http://www.computer.org/csdl/mags/co/2012/10/mco2012100022-abs.html>
- [4] Muthu Ramachandran . (2008). *Software Reuse: Components, Architectures and Product Lines*. Caedmon Hall 216, Ext 24743: M.Ramachandran@leedsmet.ac.uk.