International Journal of Electrical and Computer Engineering (IJECE)
Vol. 4, No. 1, Feburary 2014, pp. 127~137
ISSN: 2088-8708

Extractability Effectiveness on Software Product Line

Mohammad Ali Torkamani
R&D Department, Iranian Telecommunication Manufacturing Company, Shiraz, Iran

Article Info

Article history:

Received Aug 9, 2013
Revised Dec 8, 2013
Accepted Jan 1, 2014

Keyword:

Extractability
Software metrics
Software Product line
Software quality attribute

ABSTRACT

A software product line consists of a family of software systems. Most of quality attributes are defined for single systems. When we are facing a family of products instead of a single system, some aspects of architecture evaluation, such as cost, time, and reusability of available assets, become more highlighted. In this paper a new quality attribute for software product line, which we called it extractability, is introduced. Also extractability measuring method and relationship between extractability with some quality attributes is presented. At the end, Extractability Effectiveness on Software Product Line is evaluated in practice.

Corresponding Author:

Mohammad Ali Torkamani
R&D Department
Iranian Telecommunication Manufacturing Company, Shiraz, Iran
Email: torkamani_ali@yahoo.com

1. INTRODUCTION

A software product line is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way [1]. As architecture enjoys a special status in other fields of software engineering, it also plays an important key role in the software product line. Generally, one of the most important concerns of the software product line architecture is to achieve the stockholders' desired quality attributes. Most of the existing quality attributes are defined for single systems. While when we are confronted with a family of products instead of a single system, some of the aspects of evaluating the architecture such as cost, time and reusability of the existing products become more prominent [2]. So, a special quality attribute for the software product line is needed that architects can consider it in designing architecture and also stockholders can evaluate the quality of product line with its use. In this paper, a new quality attribute for software product line, named "Extractability", is presented and the Extractability effectiveness on Software Product Line is evaluated. The structure of the continuation of the paper is in this way that first the related works are studied in the second part of the paper. Then in the third part of the paper, definition of the extractability quality attribute, its general scenario, the measuring method and its relation with some of the quality attributes is presented. Part four, five and six of the paper are devoted to case study, conclusion, and further works, respectively.

2. RELATED WORKS

Some of the researchers in papers like [3] have considered common quality attributes in software engineering such as reusability and have presented methods for the increase of quality in software product line. Reference [4] also has presented the quality model of reusability for software product line. In this model, the quality attribute of reusability includes the six quality attributes of flexibility, maintainability, portability, scope coverage, understandability, and variability. References [5, 6] have considered variability as a quality attribute in the software product line. Reference [7] has also presented a general scenario for variability. Some of the references such as [8-14] have presented metrics for measuring the quality in software product line, which the most important ones are Structure Similarity Coefficient (SSC), Component reuse rate (CRR), Reuse benefit rate (RBT), Product-related Reusability (PrR), Size of Commonality (SOC), Percent reuse. References [15-20] have also presented metrics for assessment of the feature model. Reference [21] has also presented EATAM method for assessment of architecture of product line which is actually a development of ATAM method. Reference [22] has also examined quality attributes for a family of medical products. One of the prominent points of this paper is that among the presented quality attributes, two quality attribute of serviceability and safety can also be seen. Reference [23] has also examined the quality attribute for embedded software product lines. Moreover, due to the importance of cost in product line, models such as SIMPLE [24] and InCoME [25, 26] have been presented for the product line engineering. Some of the references like [27-29] have also presented quality models for software product line. Reference [30] has also examined remarkable aspects of quality necessities in software product line. Considering this fact that the major concern of this paper is related to MAP and OAR techniques, further we will describe these two methods.

2.1. MAP and OAR Techniques

MAP and OAR Methods are techniques for developing core assets for software product lines from existing assets. Major goals of MAP Method include:

- Analysis of several legacy systems to determine the commonalities and variability across these systems.
- To determine whether, from a technical viewpoint, it should undertake a software product line approach [31].

MAP can be applied along with Options Analysis for Reengineering (OAR), which is a method for measuring economic feasibility of mining the existing components for the product line. In fact, after performing MAP method and designing the architecture of new product line and after reaching to the conclusion that we should have product line, “analysis of make/buy/mine/commission” should be done. That means, the organization should decide whether to construct or to buy the required components or to extract them from the components of the existing system or to commit them to a contractor. If mining is chosen, OAR method can be applied for identification of components which have the necessary potential for reusability. OAR method specifies the mining options, cost, efforts and risks of each connection (joint). OAR method is applied to specify whether some components of the existing systems can be reused in the new product line or not? MAP method in the architecture level and OAR method in the components level looks for the assets [31].

3. EXTRACTABILITY

Extractability means the ability of finding the commonality, analysing the benefit and cost, and ease of rehabilitating the products, resources, or some pieces of the existing software systems for giving service in the software product line. The necessary condition for the system to have extractability is that assets extraction cost in it be less than a new development cost. Cost is indicated with dollar ($) unit. Table 1 indicates the general scenario for core assets extraction. In the software product line several assets are used. The value of these assets and the benefit made by the organization through each asset reuse are different. Extractability of the weight value of the assets is considered in metric. So, before we describe the way of calculating the extractability, we present the way of calculating the weight value of the assets.

3.1. Determining the Weight Value of the Assets

In past years, the focus of experts was on the reusability of fine grain assets like reusability in code level. Due to this approach, we have seen fewer successes in reusability field. Currently, the concentrations have been changed towards coarse grain assets which are being uniformed by software architecture. This approach has some advantages: a) the assets would be more appropriate for offering in market, b) it increases productivity and c) it saves time [17]. Moreover, the SEI framework of product line [32] considers product line as an attempt for employing strategic plans for coarse grain reuse. For this, larger grain assets are more valuable for reusing in software product line. In order to determine the weight value, we should convert assets and artifacts to a common measurement unit such as “Line of code” [33].

Table 1. General Scenario for core Assets Extraction

Scenario Part	Values
Source of stimulus	Development Organization
Stimulus	Wishes to finding, analyzing and rehabilitating core Assets for use in software product line
Artifact	Analysis, Architecture, Design, Part of code, Sub system, Component, Service, External Software (Such as COTS)
Environment	Before/After Product line Installation
Response	Locates in Architecture to be modified, Wrapping, Adapting components, Create/Update CONOPS, Create/Update Production plan
Response measure	Extraction Cost, Extraction Effort, Time, Level of Risk

If the number of code lines of the software assets is not available (like a situation in which an organization has purchased a commercial of the shelf (COTS)) or it is difficult to us to convert non software assets to the number of code lines, the following method can be used for determining the weight value. Suppose that among the assets, a_k asset has made the least necessary effort for development. The necessary effort for development of this asset is shown with E_k. In this case, weight value of the asset a_i is calculated with relation 1:

$$\mathrm{W_i} = \frac{\mathrm{E_i}}{\mathrm{E_k}} \quad (1)$$

It is evident that weight value of the asset a_k equals 1. The more the necessary effort for development of an asset, the more its cost would be; so in relation 1 development cost can be replaced with effort, i.e.:

$$W_i = \frac{\mathrm{Cost_i}}{\mathrm{Cost}_k} \quad (2)$$

3.2. The Method of Measuring Extractability

According to the definition of extractability, this quality attribute should be a function of the three parameters of finding commonality, analysis of cost and benefit, and ease of rehabilitating. Finding is what is done in MAP and OAR methods, i.e., determination of commonality and variability in the product line. In order to evaluate this part of work, we can divide the weight of common components in product line architecture over the weight of all the components of the product line. After commonality and variability were specified, analysis of make/buy/mine/commission should be done so that to specify which one of the choices of make, buy, commission or rehabilitating of the existing components is suitable for satisfying the new architecture. Mining and rehabilitation of the existing component to serve in a new system are chosen when mining cost is less than the cost of a new development. It's better to exert the ratio of benefit to cost i.e. the Return of Investment (ROI) in the formula instead of only considering the benefit alone. According to the component table in the OAR method [31] we evaluate the ease of rehabilitating with the amount of changes of the component for comparison with the new architecture. So we have:

$$E = \frac{1}{w_{spl}} \sum_{i=1}^{n} w_i \times ROI_i \times \left(1 - \frac{M_i}{100}\right) \quad (3)$$

In this relation, n is the number of common assets in the architecture of the product line, E is the extractability, w_i is the weight value of the asset i, w_{spl} is the weight of all assets of the product line, ROI_i is the asset i return of investment, and M_i is the percentage of changes of the component i to adapted with the new architecture. Weight of the assets of the product line is calculated using the following relation:

$$w_{spl} = \sum_{k=1}^{m} w_k \quad (4)$$

In which m is the total number of the product line assets. If after the component i the black box is reused, $1 - \frac{M_i}{100}$ equals 1. It is evident that $1 - \frac{M_i}{100}$ would be an integer, more than zero and less than one. In order to calculate ROI_i one should divide the asset extraction benefit over the asset i cost of extraction. Extraction benefit and cost of extraction are indicated with BE_i and CE_i, respectively (relation 5).

$$\mathrm{ROI_i} = \frac{\mathrm{B_{Ei}}}{\mathrm{C_{Ei}}} \quad (5)$$

The extraction benefit of asset i is calculated using relation 6. In this relation, CD_i is the new development cost of the asset i, and CE_i is the cost of extraction of the asset i.

$$B_{Ei} = C_{Di} - C_{Ei} \quad (6)$$

We consider a unit for extractability and indicate it with eu (extraction unit).

3.3. Extended Extractability

In relation 3, if structure similarity coefficient (SSC) [8] be replaced by ratio weight of common part of architecture assets to weight of product line assets (i.e., for evaluating the commonality, we use the number of assets instead of their weight) the obtained metric will be called “Extended Extractability”. Extended extractability is indicated with Ex and is calculated using the relation 7:

$$Ex = SSC \times \sum_{i=1}^{n} ROI_i \times \left(1 - \frac{M_i}{100}\right) \quad (7)$$

We can rewrite the SSC metric [8] as follows:

$$\mathrm{SSC} = \frac{|C|}{|SPL|} \quad (8)$$

In this relation, $|C|$is the number of common components of the product line architecture and $|SPL|$is the total number of the product line components. Other parameters of the relation 7 are similar to those of the relation 3.

In the case study we will examine the difference between extractability and the extended extractability.

3.4. Extractability Advantages for the Organization

- It is a method for evaluating the mining process in the organization
- It helps optimizing the scope of the product line. Contrary to “structure similarity coefficient” [8] metric, it also considers cost.
- It can be regarded as a basis for stockholders in order to evaluate the product line.
- It is a quality attribute that should be considered by the architect during architecture planning.

3.5. Extractability Correlation with Other Quality Attributes

According to the quality model ISO/IEC – 9126, each one of the quality attributes is dependent on a number of factors or sub-characteristics. For example, according to the quality model ISO/IEC – 9126, usability is dependent on factors such as understandability, learnability, operability, attractiveness, and usability compliance [34]. To specify the relation between extractability and other quality attributes, we have compared it with other quality attributes. In this comparison, factors related with every quality attribute are considered. Tables 2, 3, and 4 indicate positive and negative Correlation of extractability with system, business, and architecture quality attributes, respectively. Empty cells are indicator of ineffectiveness of dependence.

Table 2. Correlation between extractability and system quality attributes

System Quality Attributes	Extractability
Availability	
Modifiability	+
Performance	-
Security	
Testability	+
Usability	
Reliability	-
Portability	+
Reusability	+

Table 3. Correlation between extractability and business quality attributes

Business Qualities	Extractability
Time to market	+
Cost and benefit	+
Projected lifetime of the system	+
Targeted market	+
Rollout schedule	
Integration with legacy systems	+

Table 4. Correlation between extractability and architecture quality attributes

Architecture Qualities	Extractability
Conceptual integrity	+
Correctness and completeness	
Buildability	+

3.6. The Method of Improving the Architecture of the Product Line after Calculating Extractability

If the extractability of a system is low, some suggestions can be offered to the developer:

- Compare the weight value of the existing assets in common part of architecture with total weight value of the product line assets. Maybe some of the following conditions have been happened:
  - One of the products has a little commonality with other products and it is better not to be put in the product line.
  - Many exclusive features have been considered for one or several of the products. This matter can cause the product line to lose its economic advantage.
  - Weight value of the assets is not specified correctly. Regarding the fact that weight value is done on the basis of estimating the development time, it is possible that an appropriate estimation has not been done of the development time of the assets.
- Examine the cost and percent of necessary changes for adaptation once more. It is possible that an appropriate estimation of the percent of necessary changes for rehabilitating the components has not been done. So, these matters should be reassessed.

3.7. Evaluation of Extractability in Practice

In this section, the extractability effectiveness on Software Product Line is evaluated in practice in Iranian Telecommunication Manufacturing Company (ITMC). ITMC is a company operating in Electrical engineering and ICT areas. Beside some products in electrical and communication area, ITMC is developing some software. In order to take advantage of Software Product Line, R&D department of ITMC has developed five Software product lines:

- SPL1: Software Product Line for Mobile Sets
- SPL2: Software product line for Telecommunication Centers
- SPL3: Software Product line for ECU (and Smart control systems for cars)
- SPL4: Software Product line for ATM and Banking systems
- SPL5: ERP Software Product line

The main criterions for evaluating this case study are Structure Similarity Coefficient (SSC), Component reuse rate (CRR), Reuse benefit rate (RBR), Product-related Reusability (PrR), Size of Commonality or SOC [8, 10]. Other criterions used include: the number of rehabilitated components, the number of reuse in architecture common parts, the number of products, average weight of the assets of the product line, weight of members of the product line , weight of the components of the architecture common parts, weight of components of the product line, ROI, ease of rehabilitating the assets for adaptation and reuse in a product (AoR_p), ease of rehabilitating the assets for adaptation and reuse in the product line (AoR_{SPL}), impact of the reusability of an asset on developing a product (I_{ij}), the impact of reusability on developing a product (Ii), impact of reusability on developing all products (I), weight percent of products reusability ($Wt\%R_P$), and weight percent of reusability (Wt%R). AoRp, AoRSPL, I_{ij}, I_i, I, Wt%Rp and Wt%R are presented in [35]. Tables 5 to 8 indicate the data related to the product line 1. In table 5, list of common assets of the architecture of the product line has been shown. In table 6, list of other assets which are reused in some of the products of the product line 1 is indicated. In table 7, list of other assets (new development) is presented. In table 8, products of the product line 1 is presented along with the data related to each product. These items include the name of the product, name of the used assets in the product, weight value of any asset, type of asset (new development or mining), mining ROI, percent of exerting the necessary changes for comparing them with the new architecture and weight of the product (W_P). In these tables, calculation of the Wt%Rp, AoRp metrics, the impact of the reusability of an asset on developing a product (I_{ij}),), the impact of reusability on developing a product (Ii), impact of reusability on developing all products (I) and products related reusability (PrR) is also presented.

In order to perform evaluation, we calculated the evaluation criterions, extractability and extended extractability and recorded the results in table 9. In table 10, for every aspects of comparison, we specified the rank achieved by product lines 1 to 5. This table has been prepared using the recorded results in table 10.

Table 5. List of architecture common assets of the product line 1

no	asset	wi	type	ROI_M	Percept of Changes	CRR
1	a1	1	Mining	10	20	100
2	a2	2	Mining	5	BB	100
3	a3	2	Mining	7	BB	100
4	a4	1	Mining	21	BB	100
5	a5	7	Mining	50	40	100
6	a6	3	Mining	4	40	100
7	a7	5	Develop.			100
8	a8	2	Develop.			100

Table 6. List of other assets which are reused in some of the products of the product line 1

no	asset	wi	CRR
1	a9	2	40
2	a10	1	40
3	a11	1	40
4	a12	2	40
5	a13	1	20
6	a14	1	20
7	a15	1	20
8	a16	2	20
9	a17	2	40
10	a18	2	20
11	a19	2	20
12	a20	2	20

The reusability potential in product line is usually measured with SOC and SSC. According to these metrics, the more the common components of the product line, the more the similarity of architecture of product line members. So, the product line will obtain more benefit of reuse. Examination of the results of the case study indicates that product line 4 has the biggest SSC (row 7 of table 10). So this product line should have more benefit of reuse. The results of metrics of RBR and "the effect of reuse in products (I)" also confirms this matter (rows 13 and 19 of table 10). But these metrics cannot be a good criterion for evaluating the reuse in product line. The obtained results of the case study show that product line 4 has the least amount of ROI (rows 16 and 17 of table 10).

On the other hand, according to table 10, product line 2 has the first rank of weight percent of reuse (Wt%R) and AoR_{SPL} and also the second rank of ROI, the number of reuse in architecture common parts, the number of products, size of commonality (SOC), average of CRR, weight of components of the architecture common parts, reuse benefit rate (RBR), and the effect of reuse in products. According to the instances and regarding the fact that extractability of product line 2 is maximum, it can be said that this metric has provided an appropriate view about product line 2. (Remember that extractability metric is a multivariate metric and is not dependant on one factor).

But in comparison with extractability of product line 4 and 1, it is remarkable that despite the fact that product line 4 has the least rate of ROI, in respect of the number of rehabilitated components, the number of reuse in architecture common parts, the number of products, size of commonality (SOC), structure similarity coefficient (SSC), weight of components of the architecture common parts, reuse benefit rate (RBR), average of PrR, impact of reusability on developing all products (I), it has been the best product line. This is while product line 1 in most of these criterions has gained the last ranks (fourth or fifth) among these product lines. So, it can be said in respect of this product line, extractability metric has not behaved fairly. But this problem does not exist in the metric of extended extractability. If we compare extractability with extended extractability, you will see that the rank of product line 1 and 4 are substituted with each other. We can conclude from this matter that when W_{SPL} is more than a certain extent, metric of extended extractability would respond better. It is necessary to mention that product line 4 has the biggest weight of the components of the product line (WSPL), the most number of products and weight of the product line members. It also holds the second rank of the total number of the components. So we can state that this product line has the biggest product line among the product lines of 1-5. After the product line 4, product line 3 is regarded as the biggest product line, because it holds the second rank in respect of the weight of the product line components, the number of products, the weight of the product line members and holds the third rank in respect of the total number of the components. Product line 5 is also considered as the smallest product line. Comparison of the rows 7, 8, 10, 16, 17, 20, and 21 of table 10 for product line 3 and 5 indicates that in respect of this product line, extractability has provided an acceptable comment, but extended extractability has not behaved so fairly because since the structure similarity coefficient (SSC) is higher we expect the potential of the extractability reuse in product line 5 to be more and because of the better quality of ROI we expect that this product line be more economical in respect of the costs (cost-effective). Now if you suppose that product line 3 is better than product line 5 on the basis of other criterions (other rows of the table 10 except the rows of 7, 8, 10, 16, 17, 20, and 21), we can understand that extractability in respect of the product line 3 which has been a big product line, has not behaved fairly. But extended extractability has presented an acceptable comment in this regard. We can conclude from this discussion that extractability is more appropriate for small product lines and extended extractability must be used for bigger product lines. Also in comparison with metrics of size of commonality (SOC) and structure similarity coefficient (SSC) for product lines 2 and 3, we will notice that these metrics are equal and as a result they have no specific suggestion in respect of product lines 2 and 3. So, extractability can help optimization of the scope of product line beside structure similarity coefficient.

Table 7. List of other assets of the product line 1 (new development)

no	Asset	wi	*CRR*	no	Asset	wi	CRR
1	a21	1	20	21	a41	1	20
2	a22	1	20	22	a42	1	20
3	a23	1	20	23	a43	1	20
4	a24	1	20	24	a44	1	20
5	a25	1	20	25	a45	1	20
6	a26	1	20	26	a46	1	20
7	a27	1	20	27	a47	1	20
8	a28	1	20	28	a48	1	20
9	a29	3	20	29	a49	1	20
10	a30	1	20	30	a50	1	20
11	a31	1	20	31	a51	1	20
12	a32	1	20	32	a52	1	20
13	a33	1	20	33	a53	1	20
14	a34	1	20	34	a54	1	20
15	a35	1	20	35	a55	1	20
16	a36	1	20	36	a56	1	20
17	a37	2	20	37	a57	1	20
18	a38	5	20	38	a58	1	20
19	a39	1	20	39	a59	1	20
20	a40	1	20	40	a60	1	20

Table 8. The products of the product line 1

no	Product	asset	wi	Type (Mining or developed)	ROI_M	Percent of Changes (BB=Black Box)	Wp	$\sum_{i=1}^{n} w_i$	$Wt\%R_p$	AoR_p	I_{ij}	I_i	$\sum_{i=1}^{n} I_i$	PrR
1	p1	a1	1	M.	10	20	29	22	75.86206897	0.854545455	0.027586207	0.593103448	2.143200067	0.615384615
2		a2	2	M.	5	BB					0.068965517			
3		a3	2	M.	7	BB					0.068965517			
4		a4	1	M.	21	BB					0.034482759			
5		a5	7	M.	50	40					0.144827586			
6		a6	3	M.	4	40					0.062068966			
7		a7	5	D.										
8		a8	2	D.										
9		a11	1	M.	3	20					0.027586207			
10		a12	2	M.	2	BB					0.068965517			
11		a13	1	M.	5	BB					0.034482759			
12		a14	1	M.	2.1	BB					0.034482759			
13		a15	1	M.	1.1	40					0.020689655			
1		a1	1	M.	10	20			84.44444444	0.745454545	0.017777778	0.404444444		0.347826087
2		a2	2	M.	5	BB					0.044444444			
3		a3	2	M.	7	BB					0.044444444			
4		a4	1	M.	21	BB					0.022222222			
5		a5	7	M.	50	40					0.093333333			
6		a6	3	M.	4	40					0.04			
7		a7	5	D.										
8		a8	2	D.										

9		a16	2	M.	1	10					0.04		
10	p2	a17		M.	2	60	45	38			0.017777778		
11		a18		M.	1	25					0.033333333		
12		a19		M.	1	40					0.026666667		
13		a20		M.	2	45					0.024444444		
14		a21		D.									
15		a22		D.									
16		a23		D.									
17		a24		D.									
18		a25	1	D.									
19		a26	1	D.									
20		a27	1	D.									
21		a28	1	D.									
22		a29	3	D.									
23		a30	1	D.									
1		a1	1	M.	10	20					0.026666667		
2		a2	2	M.	5	BB					0.066666667		
3		a3	2	M.	7	BB					0.066666667		
4		a4	1	M.	21	BB					0.033333333		
5		a5	7	M.	50	40					0.14		
6		a6	3	M.	4	40					0.06		
7	p3	a7	5	D.			30	18	60	0.95			0.571428571
8		a8	2	D.									
9		a9	2	M.	2	15					0.056666667		
10		a31	1	D.									
11		a32	1	D.									
12		a33	1	D.									
13		a34	1	D.									
14		a35	1	D.									
1		a1	1	M.	10	20					0.021052632		
2		a2	2	M.	5	BB					0.052631579		
3		a3	2	M.	7	BB					0.052631579		
4		a4	1	M.	21	BB					0.026315789		
5		a5	7	M.	50	40					0.110526316		
6		a6	3	M.	4	40					0.047368421		
7		a7	5	D.									
8		a8	2	D.									
9		a9	2	M.	5	40					0.031578947		
10	p4	a10	1	M.	3	40	38	21	55.26315789	0.78	0.015789474	0.4	0.470588235
11		a11	1	M.	4	30					0.018421053		
12		a12	1	M.	1.7	10					0.023684211		
13		a36	1	D.									
14		a37	2	D.									
15		a38	5	D.									
16		a39	1	D.									
17		a40	1	D.									
1		a1	1	M.	10	20					0.017391304		
2		a2	2	M.	5	BB					0.043478261		
3		a3	2	M.	7	BB					0.043478261		
4		a4	1	M.	21	BB					0.02173913		
5		a5	7	M.	50	40					0.091304348		
6		a6	3	M.	4	40					0.039130435		
7		a7	5	D.									
8		a8	2	D.									
9		a10	1	M.	2	40	46	19			0.013043478		
10	p5	a17	2	M.	7	40					0.026086957		
11		a41	1	D.									
12		a42	1	D.									
13		a43	1	D.					41.30434783	0.775		0.295652174	0.266666667
14		a44	1	D.									
15		a45	1	D.									
16		a46	1	D.									
17		a47	1	D.									
18		a48	1	D.									
19		a49	1	D.									
20		a50	1	D.									
21		a51	1	D.									
22		a52	1	D.									
23		a53	1	D.									
24		a54	1	D.									
25		a55	1	D.									
26		a56	1	D.									
27		a57	1	D.									

28	a58	1	D.	
29	a59	1	D.	
30	a60	1	D.	

Table 9. Calculation of the metrics and comparing them for the product lines 1, 2, 3, 4, and 5

No.	Comparison aspect	SPL1	SPL2	SPL3	SPL4	SPL5
1	Total number of the components	60	40	40	44	16
2	Total number of the new development	42	25	25	14	6
3	Number of rehabilitated components	18	15	15	30	10
4	No. of reusability in common part of architecture	6	10	10	15	10
5	Number of products	5	6	6	10	5
6	Size of commonality (SOC)	8	10	10	15	5
7	Structure similarity coefficient (SSC)	0.133333333	0.25	0.25	0.340909091	0.3125
8	Average of CRR	32.33	38.49206349	38.49206349	35.68181818	47.5
9	Weight of product line components (Wspl)	89	72	96	105	44
10	Average weight of product line assets	1.483333333	1.8	2.4	2.386363636	2.75
11	Weight of product line members	188	206	233	725	104
12	Components' weight of the architecture common part	23	31	29	61	13
13	RBR	2.112359551	2.861111111	2.427083333	6.904761905	2.363636364
14	PrR average	0.454378835	0.657020358	0.657020358	0.727545548	0.660714286
15	AoR_p average	0.821	0.706400544	0.841656954	0.744730649	0.658333333
16	Mining ROI average for architecture common parts	16.16666667	9.625	4.625	4.483333333	6.6
17	ROI average for all the products	11.27446809	8.739130435	4.391304348	4.089847716	5.90625
18	AoR_{SPL}	0.625	0.875	0.725	0.716666667	0.66
19	impact of reusability on developing all products (I)	2.143200067	3.788222304	3.566379307	7.007818122	2.941345347
20	Wt%Rp	63.37480383	82.67909057	76.14769688	96.50385028	83.95562275
21	Wt%R	12.23404255	15.04854369	12.44635193	8.413793103	12.5
22	Extractability	3.035955056	4.219618056	0.935546875	1.78297619	1.268181818
23	Extended extractability	9.786666667	18.309375	7.784375	15.45596591	6.5625

Table 10. Rank of the product lines 1-5 of different aspects

No.	Comparison aspect	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
1	Total number of the components	1	4	2,3	5	
2	Total number of the new development	1	2,3	4	5	
3	Number of rehabilitated components	4	1	2,3	5	
4	No. of reusability in common part of architecture	4	2,3	1	5	
5	Number of products	4	2,3	1,5		
6	Size of commonality (SOC)	4	2,3	1	5	
7	Structure similarity coefficient (SSC)	4	5	2,3	1	
8	Average of CRR	5	2,3	4	1	
9	Weight of product line components (Wspl)	4	3	1	2	5
10	Average weight of product line assets	5	3	4	2	1
11	Weight of product line members	4	3	2	1	5
12	Components' weight of the architecture common part	4	2	3	1	5
13	RBR	4	2	3	5	1
14	PrR average	4	5	2,3	1	
15	AoR_p average	3	1	4	2	5
16	Mining ROI average for architecture common parts	1	2	5	3	4
17	ROI average for all the products	1	2	5	3	4
18	AoR_{SPL}	2	3	4	5	1
19	impact of reusability on developing all products (I)	4	2	3	5	1
20	Wt%Rp	4	5	2	3	1
21	Wt%R	2	5	3	1	4
22	Extractability	2	1	4	5	3
23	Extended extractability	2	4	1	3	5

4. CONCLUSION

In this paper, a new quality attribute called "extractability" was introduced and the general scenario and the way of measuring it were described. Then, we paid to the examination of its correlation with some quality attributes. We observed that this quality attribute has positive correlation with many important quality attributes. Therefore, extractability is an important quality attribute that the architect should consider it when designing architecture. Because the increase of this quality attribute leads to the increase of other important quality attributes in the product line and as a result the increase of the product line quality.

Measuring this quality attribute is done on the basis of ROI, weight value of commonality, and the degree of the ease of rehabilitating the assets extracted from the existing systems in the organization. We have defined this quality attribute for the software product line. So, against most of the quality attributes which are generally defined in software engineering, extractability is defined specifically for the software product line. We can mention the following instances as the most important advantages of measuring extractability for the organization:

- It is a method for evaluating the mining process in the organization.
- It helps optimizing the scope of software product line.
- It can be regarded as the basis of stockholders' work for evaluating the product lines.
- It is a quality attribute that the architect should consider when designing architecture.
- It can be considered as a method for evaluating the maturity of the product line.
- The measuring method of the extractability is easy.

At the end of this paper, Extractability effectiveness on software product Line is evaluated by a case study. The obtained results of the case study shows that the more the amount of reuse, the more the obtained number for extractability. Also due to the role of ROI in calculating extractability, we can say that the bigger the extractability, the more successful the product line would be in respect of saving the costs and gaining benefit. The results of the case study also show that extractability is more suitable for small product lines and extended extractability is used for bigger product lines. The other result obtained from the case study is that in some situations the obtained amount for one metric may be equal in different product lines (e.g., metrics of size of commonality (SOC) and structure similarity coefficient for product lines 2 and 3 in case study). It is evident that when these metrics are equal with each other, it has no special suggestion in respect of the product lines. So, in these conditions calculating the extractability metric can be helpful.

As a final adding up we can hint to this point that regarding this fact that extractability matches with many other metrics and also has direct relationship with many important quality attributes, generally we expect a product line which is successful in many aspects, posses also an appropriate extractability.

5. FURTHER WORK

In the case study we said that when the number of the components of the product line or their weight is more than a certain limit, extended extractability formula responds better than the formula presented for extractability. Research and investigation of this subject in a paper, would be our further work.

REFERENCES

[1] Bass L, Clements P and Kazman R. Software Architecture in Practice. Second Edition, Addison Wesley. 2003.

[2] Mirakhorli M. "Assessment of Software Product line Reliability". MSC thesis, Shahid Beheshti University, Tehran. 2007.

[3] Geertsema B and Jansen S. "Increasing software product reusability and variability using active components: a software product line infrastructure". *ACM*. 2010.

[4] Amin F, Mahmood A and Oxley A. "A Proposed Reusability Attribute Model for Aspect Oriented Software Product Line Components". *IEEE*. 2010.

[5] Myll¨arniemi V, M¨annist¨o and Raatikainen M. "Quality Attribute Variability within a Software Product Family Architecture". *Short paper in Conference on the Quality of Software Architectures (QoSA)*. 2006.

[6] Etxeberria L and Sagardui G. "Evaluation of Quality Attribute Variability in Software Product Families". *15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems.* 2008.

[7] Bachmann F and Clements P. "*Variability in Software Product Lines*". Technical Report, Carnegie Mellon University. 2006.

[8] Zhang T, Deng L, Wu J, Zhou Q and Ma C. "Some Metrics for Accessing Quality of Product Line Architecture". *IEEE*. 2008.

[9] Berger C and Busse C. "*Product Line Metrics for Legacy Software in Practice*". The 14th International Software Product Line. 2010.

[10] Berger C, Rendel H and Rumpe B. "Measuring the Ability to Form a Product Line from Existing Products". *Variability Modelling of Software-intensive Systems (VaMos).* 2010.

[11] van der Hoek A, Dincel E and Medvidoviü N. "*Using Service Utilization Metrics to Assess the Structure of Product Line Architectures*". Proceedings of the Ninth International Software Metrics Symposium (METRICS'03), IEEE. 2003.

[12] Aldekoa G, Trujillo S, Sagardui G and Díaz O. "Experience measuring maintainabilityin software product lines". *XV Jornadas de Ingeniería del Software y Bases de Datos (JISBD).* 2006.

[13] Rahman A. “Metrics for the Structural Assessment of Product Line Architecture”. MSC Thesis, School of Engineering at Blekinge Institu te of Technology, Sweden. 2004.
[14] Junior E, Gimenes I and Maldonado J. “A Metric Suite to Support Software Product Line Architecture Evaluation”. *XXXIV Conferencia Latinoamericana de Informática (CLEI 2008)*. 2008.
[15] Benavides D, Trinidad P and Ruiz-cortés A. “*Automated Reasoning on Feature Models*”. LNCS, Advanced Information Systems Engineering: 17th International Conference, CAISE. 2005.
[16] Mendonca M, Wasowski A, Czarnecki K and Cowan D. “*Efficient compilation techniques for large scale feature models*”. In Generative Programming and Component Engineering, 7th International Conference, GPCE, Proceedings. 2008: 13–22.
[17] Fernandez-Amoros D, Gil R and Somolinos J. “*Inferring Information from Feature Diagrams to Product Line Economic Models*”. ACM International Conference Proceeding Series Vol. 446, Proceedings of the 13th Internationsl Software Product Line Conference. 2009: 41-50.
[18] Czarnecki K and Kim P. “*Cardinality-based feature modeling and constraints: A progress report*”. In Proceedings of the International Workshop on Software Factories At OOPSLA 2005. 2005.
[19] Kasikci B and Bilgen S. “Scalable modeling of software product line variability”. 13th *International Software Product Line Conference, SPLC 2009*, SanFrancisco. 2009.
[20] Benavides D, Segura S and Ruiz-Cortés A. “Automated analysis of feature models 20 years later: A literature review”. *Information Systems*. 2010.
[21] Kim T, Ko I, Kang S and Lee D. Extending ATAM to Assess Product Line Architecture. *IEEE*. 2008.
[22] Wijnstra J. Quality Attributes and Aspects of a Medical Product Family. *IEEE*. 2001.
[23] Kolb R and Bayer J. Pattern-Based Architecture Analysis and Design of Embedded Software Product Lines. *EMPRESS consortium*. 2003.
[24] Clements P, McGregor J and Cohen S. The Structured Intuitive Model for Product Line Economics (SIMPLE). technical report, Carnegie Mellon University. 2005.
[25] Nóbrega JP. “An Integrated Cost Model for Product Line Engineering”. M.Sc. Dissertation, Universidade Federal de Pernambuco, 2008, Available: http://www.ivanmachado.com.br/research/rise/thesis/files/2008_JarleyNobrega_msc.pdf
[26] Nóbrega JP, de Almeida ES And Meira SRL. “InCoME: Integrated Cost Model for Product Line Engineering”. 34th *Euromicro Conference Software Engineering and Advanced Applications, IEEE*. 2008.
[27] Thörn C. A Quality Model for Evaluating Feature Models. *Software product line conference (SPLC)*. 2007.
[28] Trendowicz A and Punter T. Quality Modeling for Software Product Lines. 7th *ECOOP Workshop on Quantitative Approaches in Object-Oriented Software Engineering (QAOOSE’03)*. 2003.
[29] Bartholdt J, Medak M and Oberhauser R. Integrating Quality Modeling with Feature Modeling in Software Product Lines. *Fourth International Conference on Software Engineering Advances*. 2009.
[30] Etxeberria L, Sagardui G and Belateg L. “Quality aware Software Product Line Engineering”. *Journal of the Brazilian Computer Society*. 2008.
[31] OíBrien L and Smith D, MAP and OAR Methods. *Techniques for Developing Core Assets for Software Product Lines from Existing Assets*. Technical Report, Carnegie Mellon University. 2002.
[32] SEI, A Framework for Software Product Line Practice, Version 5.0, 2009, http://www.sei.cmu.edu/productlines/frame_report/miningEAs.htm
[33] Zubrow D and Chastek G. *Measures for Software Product Lines*. Technical Report, Carnegie Mellon University. 2003.
[34] Modiri N, Sadre Refiei K and Ahangari S. *Software Quality*. Mehregane Ghalam. 2010.
[35] Torkamani MA and Shams F. *Some metrics to evaluate reusability of Software product line Architecture*. AWER Procedia Information Technology and Computer Science. 2012; 1: 688-698.

BIBLIOGRAPHY OF AUTHOR

Mohammad Ali Torkamani was born in Iran, Shiraz City, in 1975. He received the M.S. degree in software engineering from the Shahid Beheshti University, in 2011. He is the author of 15 books (in Persian), more than 35 articles. His research interests include software architecture, Ultra Large Scale systems, cryptography and Network security and holds one patent.
He is working in R&D Department of Iranian Telecommunication Manufacturing Company now. Also, he is currently teaching at the University of Applied Science and Technology in Shiraz.