International Journal of
Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-4, Issue-12)

Research Article | **December 2015**

Identification of Common Variants in Enterprise Business Applications

Manish Peshkar, Dr. Vinay Chavan
Associate Professor and Head, Department of Computer Science,
S. K. Porwal College, Kamptee, Nagpur, Maharashtra,
India

Abstract:

In this paper an attempt has been made to identify common variants found in enterprise business applications. Previous attempts have been made to find the commonality/super set for any specific application domain resulting in a Product for that application domain , whereas this paper attempts to find the commonality across the domains resulting in creation of platform or the framework for enterprise application development. Common variants have been found out by studying the 14 Enterprise Business Applications in different sectors viz. Corporate , Private and local Urban bodies.

Keywords: Common Variants, Enterprise Business, commonality, Information System, Enterprise Business Components , Functional Requirements.

I. INTRODUCTION

The emergence of enterprise application has become most relevant in large business enterprises. The existing Enterprise Business Application suppliers seek to establish themselves as primary provider of this business and the IT backbone that supports the enterprise's operations. In fact large business enterprises applications are complex in nature. They are scalable, distributed and component-based and are also mission-critical. They may be deployed on a variety of platform across corporate networks. The networks may be intranet or internet and data-centric. They are necessarily to be user-friendly and should meet the stringent requirements from administration and maintenance aspects.

A common requirement while developing business applications is to support variations in the behavior of business applications based on factors like business structure, lines of business (LOB), customer segment/cluster, delivery channel, user location and many more while deploying the application. The application designed for a domain does not suffice the factors considered but different business processes are required to have different behavioral pattern for different activities and operations of business enterprises..

The different requirements were being addressed by having separately developed applications in silos for the different LOBs, marketing channels, locations etc. The traditional systems thus adopted for having silos to develop multiple applications for different LOBs, marketing channels, geographies etc. have now become redundant, as the silos of applications/systems does not give seamless flow of information across the enterprise, particularly after opening of number of new avenues and out-lets of foreign investors and multi-national undertakings. The silos system strategy has resultant effect of duplication and inconsistencies and unsatisfactory end-user experience-results, particularly in rigid Information Technology systems resulting in time-lag which in turn loses business opportunities.

The basic attempt of this paper is to find out all such common cross cutting business concernsi.e variants.

The development of single application which will address the whole gamut of all requirements and all contexts. The variations in this for the various contexts are inter-spread across various locations of application implementations resulting in spaghetti of variations. This would make the application fragile and hard both to analyze what has been varied with a view to have adequate and right further variations as need be.

Thus there is a need for better strategy to look at these requirements for variations based on usage context like LOBs, channels, and geographies etc as cross cutting business concerns and accordingly define strategies for architecting enterprise applications with well defined mechanisms for handling the variations for these concerns.

On screening Enterprise Business Application ; it is clear that the variants of PRS devised for particular domain are the attempts from one perspective and therefore the main thrust of the study has been that some common variants from other perspective can be clubbed to address the complexity.

II. RELATED WORK

Stuart R. Faulk, University of Oregon have given the " Product-Line Requirement Specifications (PRS) : an Approach and Case Study" , a systematic approach to developing a Product-line Requirement Specifications. The Product-line Requirement Specifications explicitly represents the family's common requirements specifications(SRS) as well as allowed variations that distinguished family members.

Deepak Dahiya, Pooja Jain in their paper "Enterprise Systems Development: Impact of Various Software Development Methodologies" [1]. have compared various software development methodologies used for developing enterprise business applications. They have concluded that identifying one stop solution in terms of a software development methodology for enterprise wide business application development whose various sub-components or sub-stages can be best used to describe a software development scenario is still an evolving domain.

KemaforAnyanwu, Amit Sheth, Jorge Cardoso, John Miller, KrysKochutm of LSDIS Lab, Department of Computer Science ,University of Georgia, Athens GA 30602-7404, USA in their paper "Healthcare Enterprise Process Development and Integration"[2]. explained the role of enterprise application in improving the efficiency of enterprise application with special reference to healthcare.

They have also inferred that the administrative tasks , large volume of data and large number of personnel is handled by variety of business software applications and information systems which are very often heterogeneous, autonomous, and distributed. They have also stated standards used for various features of enterprise business applications.

The comparison made by the author of four different healthcare enterprise applications shows that each of the enterprise application developer has used some standard for development of each of the feature, process or sub-process such as application security, workflow automation etc.

As per the white paper published by BMC software [3]. In respect of their enterprise business applications for mobiles, they have elaborated seven laws for mobile enterprise application. Important enterprise application requirements are –

a. Role Based and User Configurable
b. Device Independence
c. Interoperable
d. Should Not be Required to Change the Underlying core Application
e. Capable of Being Administered without Domain Expertise.

Kyo C. Kang of Department of Computer Science and Engineering, Pohang University of Science and Technology (POSTECH), Pohang, Korea in their paper "Issues in Component-Based Software Engineering" explained how to increase the productivity and quality of software by reusing software components. They have also suggested paradigm shift from specific business application "development" to that of "integration."[4].

Peter Fettke and Peter Loos of Johannes Gutenberg-University Mainz, Chair of Information Systems & Management, in their paper "Specification of Business Components" explains holistic approach to specify a business component consisting of seven specification levels which address both technical and business aspects. [5]

RajaaSaidi, Agnès Front, Dominique Rieu, MouniaFredj, Salma Mouline in their paper "From a Business Component to a Functional Component using a Multi-View Variability Modeling "introduced the concept of Functional Component (FC) that captures similarities and variations between BCs that share common behaviors in order to increase their reusability in different business domains by using a multiview variability.[6]

Arvinder Kaur, Student, GNDEC and Kulvinder Singh Mann, LDH Assistant Professor, GNDEC, LDH in their paper "Component Selection for Component based Software Engineering" [7]. explains an approach for defining evaluation criteria for reusable software components. They also elaborated problems faced during component selection.

He Jifeng, Xiaoshan Li and Zhiming Liu in their paper "Component-Based Software Engineering- the Need to Link Methods and their Theories"[8] explains and defines some important concepts of component software development including, interfaces, contracts, interaction protocols, components, component compositions, component Publication and refinement.

III. OBJECTIVES

The following objectives are set for identifying the common variants and its use in enterprise business applications for paradigm shift from traditional business application development to the integrated enterprise business application development. –

i. Identification of common variants for component base software used in enterprisewide business and identifying sub-components, sub stages used in different domains of enterprise business applications of information system.
ii. Evolve the basic components of enterprise business application and various alternatives available for each component, and evaluate the selected components andanalyse its impact on business application.
iii. Studying the common variants used in heterogeneous autonomous and distributed environment of business applications in information system.
iv. Identify the functional areas and domains which can be used for similarities to facilitate different common functionalities used in enterprise business application and its behaviour .
v. Find all functionality requirements of internal and external interfaces, attributes and defining the performance standard which are used by common variants in real life situations of enterprise business applications.
vi. Automation of the use of common variants in business information systems for efficiency and cost effectiveness in enterprise business applications.

Simplifyingcomplexity, increasing reliability, reusability and evaluation of common variants used in enterprise business application for improving productivity and quality of applications developed and designing the strategies used for resolving in conciseness in business application opportunities.

IV. IDENTIFICATION METHOD AND COMMON VARIANTS

Number of learned scholars have worked but it has been inferred that from enterprise software development perspective instead of writing the system requirement specification (SRS), it is suggested to start from Product

Requirement Specification (PRS) although common variants and its variations for effective handling. But, while perusing various organizations of public & private enterprises along with urban development institutions and their respective activities and methodologies in practice supported by software formats used by them, it has been construed that common variants, discussed hereunder, have also been found useful crafted for particular domain are suitable for other activities. Therefore endeavor has been made to crystalize them by micro-analysis which shows that following steps are apt to further streamline the findings of common variants.

i. The users are working at different places and are busy in sales, marketing and competing with contemporary market players. Therefore, the system developed for particular domain, need not be handy for other ones. Even in one industry like banking, different banks resort to different versions for banks, universities, insurance , octroi check post,airlines , health insurance , infrastructure. They address different types of epidemics spread over the different climatic zones. They also have to rush to sites for urgent medical assistance as a part of disaster management. They have complex role to play. Branded linen industries are also facing acute competition particularly after opening of global outlets.
ii. Each operation units runs the complete functional module to manage their day - to - day operations. This aspect has a unique perspective. The module need not be useful at a particular operating unit if the functional area is not handled there, for instance one man location may not need HR, time attendance, however, they may need fully operational inventory module.
iii. Each functional area generates few documents in the process, may it be for external or internal use. Industrial units may generate gate pass, receipts, indents, purchase order, reports, payment receipts, sheets, tabulations of varied entities generating monthly bills based on different plans floated by different vendors where uniform aspects for all is worth considering.
iv. Each document generated by enterprise business application have been perused and cross-section of users have been interviewed. On minute and careful perusal, initially logical common variants have been identified. Further, it identifies the actual common variants. These identified common variants have been further segregated and each variant has been codified by one identity to the logical unit of functionality. The whole exercise undertaken for the process of identification has been taken rather than extending the line drawn by PRS designers.
v. All enterprise business applications are developed and deployed on some technology platform which consists of operating system, application server data-base and programming languages. To effectively and for optimum usage some technical features are also to be considered. The steps taken to analyze the approach which are illustrated hereunder:
    a. Analytical exercise:Use of independent function unitin enterprise business application which deals with mainly on finance, human resources, material management, manufacturing, octroi application, rahadari, cess collection, etc. In the process of these modules, each one generates lots of documents for internal or external use, purchase orders (external), purchase returns, goods receipt notes, receipt vouchers and issues vouchers. These documents if further crystallized, each one of them states further detailing like belongingness, creator's information, owners responsibility, uniqueness, supporting or documents viz. digitized signature of authorized personnel, document data subscribing PO number, PO date, item details, item number, rate, quantity, delivery terms, payment terms etc.
    b. Upon analyzing these attributes of the documents, each document has in common belongingness, uniqueness, owner/responsibility, reference documents and event timings. These characteristics are found common whether the document is indent in inventory module or it is an investigation result from pathology laboratory and octroi payment receipt in an octroi application. Upon critically studying the aspect of belongingness of the document, it has been found that it shows belongingness towards the particular office of organization (Operating unit)

V. COMMON VARIANTS IDENTIFIED

It has also been observed that each organization works under different and multiple legal entities. Hence it is crystallized that '**Multiple Organization Support'** in the identification process.

Each document is uniquely identified in its life- time; this uniqueness is achieved by numbering the documents user-wise, office-wise, date-wise etc. Essentially this process ensures that each and every document can be referred uniquely and precisely by the number is **'Document Number Generation'.**

No document has an authenticity unless it is manually or digitally signed which defines the ownership and the responsibility of the organization on issuing or generating the document. To implement a mechanism in any module, it has been identified variant as **'Workflow'.** Extending the discussion on document further each document contains the reference of other document and in the event of any necessity arises, these reference documents are also required to be preserved along with the document **'Document Preservation'**.

In a large enterprise, the functional modules used in different offices behave marginally different to adopt the local requirements & hence the application behavior should be absolutely parameter driven to achieve these dynamics. Thereby one of the common variants identified is **'Application Set-up Parameter'**.

If an enterprise has multiple locations and numbers of users are also varied, then the user management comes in as a key aspect as it would extend controlled management but will also have user preferences. Therefore, the common variant identified in this regard would be termed as **'System Administration'**. On further amplification of the social

administration there crops up a need to analyze and maintain user-wise statistics. This common variant can therefore be termed and coded as **'Application Statistics'**.

In number of enterprise application, there is always existence of second colloquial language and therefore common but vital variant identified in the exercise as **'Multilingual Support'**.

In the case of multiple database support: the cost effectiveness is prime object. It would be prudent to consider the acquirement of database support depending on available work force which can be increased or decreased as the necessity arises. Therefore another common variant identified is **'Multiple Database Support'**.

As the enterprise business application has traditionally been viewed as reactive source as it is predominantly makes available reports of various aspects, which end-users have to search for themselves. This has to be transformed as pro-active system to make the information available to end-users at their door step. This common variant identified thus can be termed as **'Notification'**.

The results of system administration coupled with multiple organization support, have brought effective and efficient common variant in all enterprise business application and is therefore termed as **'User Management'.**

The other common variant is identified and termed as **'Calendar'** providing ease of collaboration, availability of resources allowing to calculate the efficacy of the process and which helps in deciding the service level agreements for stakeholders.

The number of users in enterprise business applications being high, the **'User Preferences'** have been left to the discretion to configure user preferences of their choice but maintainence of secrecy has been ensured.

The aspect of '**Security**' at every level in operationalization of enterprise business application , it could be at a server level , network level and database level. In certain cases security measures are implemented by having access to IT resources. This feature is preferred for enforcing the governing laws.

There exists a requirement for interfacing some or other device to automate the process. It is also necessary in view of minimizing errors in manual data capturing and to avoid any scope for malpractice. The attendance recorder or EPABX are necessarily to be interfaced with enterprise business applications. Certain other devices , particularly in medical side , are also to be interfaced. This variant termed as '**Interfacing**' has been considered.

Lastly, a few more technical variants have been encountered which demand the security of IT resources, inter-facing various devices with the application and which are respectively termed as 'security & interface' variant. In certain instances, it has also been observed that data needs to be synchronized from one location to other or variant it termed as **'Data Synchronization'**. The changes likely to be cropped up in future depending on external outcomes or changes in applicable rules or laws amended from time to time, also has to be considered while designing an application & this aspect is also an important common variant which is termed as '**Changed Management'**

VI. FINDING AND RESULTS

Average use of identified common variants in organization

Sr. No.	Feature No	Purpose	Average
1	1.1	Identification in database	100.00
2	1.1.1	Level of operating unit	85.71
3	1.2	Functional level of operating units	68.57
4	1.2.1	Special information of Operating unit.	78.57
5	1.3	Single parent- multiple child or tree structure of organization	33.33
6	1.4	Cross-referencing of inter operating unit transactions	52.38
7	1.5	Cross-referencing of intra operating unit transactions	60.12
8	2.1	Definition level	71.43
9	2.2	Global level	
10	2.3	Organization level	
11	2.4	Operating unit level	
12	3.1	Manage user rolewise	90.48
13	3.2	Operational object level user rights	95.24
14	3.3	Operational object's level user rights	97.62
15	3.4	Additional/exceptional user rights	38.81
16	4.1	Users at legal entity level	52.38

17	4.2	Users at operating units level	55.48
18	4.3	Users roles at operating units level	52.38
19	4.4	Adhoc rights user requirement	5.95
20	5.1	Calendar management	63.21
21	5.2	Multiple entity calendar	47.38
22	6.1	Notification requirement	71.55
23	6.2	Notification Modes	
24	6.3	Notification log maintainence	21.91
25	7.1	Look and feel settings	50.00
26	7.2	Password change	95.24
27	7.3	Switching operating units	35.12
28	8.1	Uniqueness	98.81
29	8.2	Traceability	73.81
30	8.3	Audit	95.24
31	8.4	Consolidation	92.86
32	9.1	Tag with transactions	48.57
33	10.1	Transaction audit	100.00
34	10.2	Transaction statistics	92.86
35	11.1	Documents	50.12
36	11.2	Approval Nature viz. Dynamic or Static	
37	12.1	Requirement	44.05
38	12.2	Level viz. Label or data or both.	
39	13.1	compatibility and operability	50.00
40	14.1	Application Usage	
41	15.1	Awareness	63.21
42	15.2	Application server	60.71
43	15.3	Transaction	28.57
44	15.4	Network	80.95
45	15.5	Information technology resources	35.71
46	16.1	Need	16.67
47	16.2	Devices	1.19
48	17.1	Need	14.29
49	17.2	Areas	7.14
50	18.1	Effectiveness	86.91
51	18.2	Methods	35.71

(Source – complied by the researcher)

Table gives a comprehensive tabulation of entire data collected by common variant by organization. Organization wise averages of each variable have also been found out.

VII. RESULTS AND DISCUSSIONS

On surveying number of organizations implementing enterprise business applications ,studied by us have provided commonality in multiple business process or business functions; it has been found that identified common variants are reusable in software. While designing the architecture of common platform , the coherent business functionalities are used to define commonality. With the advent of application software as a service (SaaS) in cloud computing , number of organizations have preferred to subscribe . Each operating unit becomes separate organization within one legal entity. All common business components studied are enabling inte-company and intra-company implementation in enterprise business applications. The data being stored commonly , it not only offers flexibility for data sharing and manipulation but also provides inter-operability in other legal entity.

VIII. CONTRIBUTION

The paper has brought the fact to the fore that besides usual variants there are various common variants which are also applicable for compact mechanism which covers vast area of multiple organization support, application set up parameters, system administration , user management , calendar , notifications , user preferences , document preservation

, application statistics, workflow, multi-lingual support , multi-database support , interfacing , synchronization and change management. The other aspects like indent, purchase order , purchase return , receipt voucher, etc together with belongingness, creators information , responsibility , uniqueness, date and time of document enforcement have also brought under one umbrella so that a compact structure is developed which will be useful irrespective of line of activities. The exercise of putting all common variants in a structured manner will be handy to all even in those cases who do not require all the aspect at the initial stage but my need them over a period of time. Thus a compact structured enterprises business application framework , if evolved , can bring historical change in software environment.

ACKNOWLEDGEMENTS

I am extremely thankful to **Dr. Vinay Chavan**for his support and guidance at every stages of this work. I am also thankful to all the Directors and staff other organizations for their support for giving the data during the research and for their cooperation.

REFERENCES

[1] Deepak Dahiya and Pooja Jain, Enterprise Systems Development: Impact of Various Software Development Methodologies, International Journal of Advancements in Computing Technology Volume 2, Number 4, October 2010.
[2] KemaforAnyanwu, Amit Sheth, Jorge Cardoso, John Miller, KrysKochut, Healthcare Enterprise Process Development and Integration, Journal of Research and Practice in Information Technology, Vol. 35, No. 2, May 2003.
[3] bmcsoftware, The Seven Laws of Mobile Enterprise Applications, 2011.
[4] Kyo C. Kang, Issues in Component-Based Software Engineering. Computer based learning unit, University of Leeds, April 1999.
[5] Peter Fettke and Peter Loos, Specification of Business Components.
[6] RajaaSaidi, Agnès Front , Dominique Rieu, MouniaFredj, Salma Mouline, From a Business Component to a Functional Component using a Multi-View Variability Modelling, Proceedings of MoDISE-EUS 2008.
[7] Arvinder Kaur and Kulvinder Singh Mann, Component Selection for Component based Software Engineering, International Journal of Computer Applications (0975 – 8887) Volume 2 – No.1, May 2010.
[8] HeJifeng, Xiaoshan Li and Zhiming Liu, Component-Based Software Engineering.- the Need to Link Methods and their Theories, United Nations University International Institute for Software Technology, August 2005.

