

A size measurement method for Enterprise Applications

Neslihan Küçükateş Ömüral¹ and Onur Demirörs²

¹ Middle East Technical University, Üniversiteler, Çankaya, Ankara, 06800, Türkiye

² Izmir Institute of Technology, Gülbahçe, Urla, Izmir, 35430, Türkiye

Abstract

Enterprise Applications are known as one of the best practices of software reuse. They are complex applications, including most of the business processes. In this domain, size measurements and effort predictions are mostly performed in an ad-hoc fashion, and they frequently suffer from schedule and budget overruns. We developed a size measurement method for Enterprise Applications and explained this novel method in this paper. We categorized transactions as “unchanged”, “changed”, and “new” in this method. We defined a size measurement unit, Data Transaction Point (DTP), and measured size as DTP in these categories. We conducted a sample size measurement with a well-known business process to demonstrate the implementation of the method.

Keywords

enterprise applications, enterprise systems, enterprise resource planning, software size measurement, reuse, data transaction point

1. Introduction

Enterprise Applications are complex and effort-intensive applications that are implemented by most organizations. These applications initially emerged as MRP (Manufacturing Resource Planning) Systems, and evolved into ERP (Enterprise Resource Planning) Systems as a result of the need to run different types of transaction-based back office operations [1], [2]. Later, with the addition of processes including front office and inter-organizational operations, the scope expanded with applications such as SCM (Supply Chain Management) and CRM (Customer Relationship Management) [1], [3], [4]. Enterprise Applications are defined as “commercial software packages that enable the integration of transactions-oriented data and business processes throughout an organization (and perhaps eventually throughout the entire inter-organizational supply chain)” [4].

Enterprise Applications are referred to by different terms such as “Enterprise Systems”, “Enterprise Software”, and “Enterprise Software Applications”. In this paper, we use the term “Enterprise Applications” and define EA projects as the implementation of Enterprise Applications in specific organizations.

Markus and Tanis [4] defined the following critical characteristics of Enterprise Applications:

- *Packages*: Enterprise Applications are commercial software packages; purchased or leased from software vendors instead of developed from scratch.
- *Software Life Cycle*: Rather than designing a new software to meet the organization’s needs, the adopters of Enterprise Applications often try to adjust organization’s business processes to the EA package.

Proceedings Name, Month XX–XX, YYYY, City, Country
EMAIL: email1@mail.com (A. 1); email2@mail.com (A. 2);
email3@mail.com (A. 3)

ORCID: XXXX-XXXX-XXXX-XXXX (A. 1); XXXX-XXXX-XXXX-XXXX (A. 2); XXXX-XXXX-XXXX-XXXX (A. 3)



© 2020 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

- *Best Practices:* Enterprise Applications are designed to handle generic business processes. Organizations redesign their business processes to adopt the EA package's best practices. Business process reengineering is an essential part of EA projects.
- *Integration:* In many cases, EA adopters need to interface the EA package to the organization's existing software.
- *Configuration:* During the EA project, the configuration is done by setting software parameters based on the business processes to be implemented. The process of configuring an EA package for an organization is significantly different from software programming.
- *Evolving:* Enterprise Applications are rapidly changing based on industry expectations.

Enterprise Applications include all processes, interactions, and financial transactions of different organizations on the same platform, which makes their structure complex. Daneva and Wieringa [5] stated that EA projects, unlike other types of software projects, include thousands of business activities, require diverse configuration and modification activities, and do not have a master architecture.

Enterprise Applications are distinguished from other software with their high reuse rates by reusing many of the pre-built functions to meet customer requirements. "Software reuse" is defined as the "use of existing software or software knowledge to construct new software" [6]. By allowing adopters to handle various business requirements by configuration and customization, EA projects are considered as one of the most successful reuse implementations in Software Engineering [7].

In EA projects, applying required changes to the package is performed by using configuration and customization. "Configuration" is defined as setting necessary business-specific parameters in the system to run the EA package; configuration does not comprise source code changes. On the other hand, customization comprises source code changes. Customization is a term in EA that is used for modifications made to the software to handle the customer's requirements that are not supported by pre-built functionalities of the software [8], [9], [10]. Customization is implemented by modifying the source code of the software. The relationship between customization and reuse is inverse; as customization increases in an EA project, the reuse rate decreases. Due to the structure of EA projects, customization is inevitable in these projects. Results of the EA research [11] showed that 64.3% of organizations participating in the research performed customization in their projects.

Daneva [12] performed an empirical study to explore reuse, and she measured the levels of reuse in three SAP projects in the same business sector. According to the results of the study, reuse rate was up to 80% at best, and reuse levels varied based on the type of implemented module. With these findings, she argued that the reusability of the ERP projects should not be overvalued. She claimed that companies adopting ERP should be ready to the minimum of 20% (in some cases presenting 40%) of pre-built functionality would not fit their business requirements, and customization would be inevitable.

In EA projects, size measurements and effort estimations are often performed in an ad-hoc fashion. They often suffer from time and budget overruns, as demonstrated by many reports such as [11] published in the field. After Stensrud [13] published the first study claiming that conventional size measurement and effort estimation methods are not suitable for such complex projects, several research studies have been conducted handling the EA size measurement problem. Function points-based size measurement methods were mostly evaluated in these studies, and "Business Blueprint" is the most used resource for sizing [14].

Software size is used as a major input for project time scheduling, effort estimation, quality management, productivity measurement, risk management, and outsource management processes. It could also be used as a base unit for software acquisition, scope changes, and normalization of base project measures [15]. For software size measurement, various methods have been developed over the years. Function point-based methods such as COSMIC, IFPUG, and NESMA are widely used in software engineering [16]. These functional size measurement methods have been evaluated as potential methods for EA projects. Although many size measurement methods have been studied for EA projects, none of the methods is widely accepted and validated as a suitable method for EA size measurement [14].

Considering the features that distinguish EA projects from other software projects, we developed a size measurement method for EA projects. In this method, "changes", where pre-built functionality was insufficient for the customer requirements, are measured. We defined a new size measurement unit,

namely Data Transaction Point (DTP), for EA projects. We counted “data groups” in transactions and measured “changes” at the transaction level. This paper presents this proposed size measurement method with a detailed size measurement sample.

The rest of the paper is organized as follows: Section 2 reviews related research regarding software size measurement and effort estimation for EA projects. Section 3 describes the proposed size measurement method. In Section 4, a sample size measurement is explained in detail. The main conclusions are presented in Section 5. In Section 6, possible directions for future work is presented.

2. Background and related work

Size measurement is accepted as one of the most critical processes of software project management. Ozkan et al. [15] defined functional size as the major input for effort, cost, and schedule prediction, and they also stated that project scope changes could be measured using functional size. They listed many other contributions of functional size to software project management, being used as a software acquisition unit, and normalizing base project measures, such as performance and quality [15].

The first measure for sizing software was the Lines of Code (LOC), where size measurement is done by counting the lines of source codes of the software. It is an easily applicable and automated size measurement method, but it cannot be used for measurement at the requirements elicitation phases of the projects. In order to handle size measurement based project management problems, various size measurement/estimation approaches have been developed over the years, and the most used ones were approaches based on measuring functionality [16].

The first study in the literature on this issue was the study of Stensrud [13]. He claimed that existing effort estimation methods are unsuitable for comprehensive projects such as EA projects. He argued that specific metrics should be defined for size measurement/effort estimation of these projects and identified potential metrics, namely modules, software interfaces, business units, users, sites, EDI interfaces, custom-developed reports, data conversions, and modified screens [13].

Daneva and Wieringa [5] evaluated if existing size measurement and effort estimation methods were applicable to the EA domain in their study. Daneva defined the existing state-of-art for size and effort estimation of cross-organizational ERP projects [17]. In this study, asynchronous focus groups, including representatives from different stakeholders of ERP projects, were constructed; according to the results, there was no consensus among representatives about how to define and measure size, and whenever size is used in a project, it was mostly functional size [17]. Daneva [18] proposed to integrate COCOMO II, Monte Carlo simulation, and portfolio management methods for effort estimation and validated this approach by a case study. In this study, it was concluded that such an approach increases the probability of success in projects with high uncertainty.

RICE (Reports, Interfaces, Conversions, Enhancements) objects of EA projects have also been examined for measuring size. Wilson [19] developed a statistical model to predict size and effort with RICE objects and conducted a case study. Case study results showed a strong correlation between the RICE objects and the development effort [19].

Erasmus and Daneva [20] indicated that new in-memory database technologies such as SAP HANA enabled all ERP solutions to run on the same platform, and this increased the level of uncertainties and customizations in the projects. They analyzed if the Expert Judgment method could be applicable in these conditions. Erasmus and Daneva continued their studies [21], [22] with an integrated method including Function Points and Expert Judgments.

The applicability of Function Point based size measurement methods in EA projects has been investigated in many studies. Vogelezang [23] measured size with COSMIC-FFP and estimated effort by using historical productivity rates. Téllez [24] developed and explored a new size measurement method, namely “eEPC-COSMIC” in his thesis study. Erasmus [25] proposed another function points-based method, “COSMIC EPC”, in which COSMIC function points were calculated based on business process models.

The “reuse” concept in EA projects has also been examined in the studies. Daneva [12], [26] explored how to measure reuse in EA projects, where she used IFPUG function points for measuring size. In the study [27], an approach to measuring reuse reflective size of EA projects was proposed.

This approach was implemented in an SAP project, and results showed that reuse reflective size led to more accurate effort estimations.

An SLR (Systematic Literature Review) study [14] was published for size measurement and effort estimation in this domain. 41 primary studies were reviewed in this study; it was concluded that various size measurement/effort estimation methods were explored in this domain, and the most used ones were function point-based ones. The main implication of the study was no consensus occurred on how to measure the size of these kinds of projects.

Recently, cloud technology has been leading SaaS (Software as a Service) models for Enterprise Applications as well. In this way, smaller-scale organizations have a chance to implement the EA. EA has already a big share in the software industry, and by offering cloud-based implementation models, the market share of EA is expected to increase. One of the most significant shortcomings of the EA domain is the lack of a valid and effective size measurement method for these projects and the resulting budget and schedule overruns. As the EA market share grows, this problem will become more visible.

3. The size measurement method

Considering the implications of the SLR [14] and exploratory case studies [27], [28] we conducted, we developed a size measurement method for EA projects. We concluded that the most critical feature of these projects that affecting sizing was “high reuse rates”. Reuse rates determines also customization levels; if reuse rate is low, this means there would be more customization. There could be cases where pre-built functionality is not sufficient for the customer requirements, even with customization. In these cases, new transactions could be developed using EA’s programming language. Thus, we need to measure “changes” where customization is required and “new” transactions are developed.

For measuring “changes” and “new” transactions, we propose to count “data groups” used in each “transaction”. We defined a novel size unit, namely “Data Transaction Point (DTP)”, as a measure of EA projects’ size. DTP is a size unit showing the number of data groups executed by transactions; in other words, it shows data points of a transaction. We claim that a change in a transaction containing multiple data groups has more effect on size than a change in a transaction containing one data group. We argue that if a transaction has a large number of data groups, it will have a greater impact on size since the change in that transaction will be reflected across all data groups.

We list data groups and transactions based on business processes. We defined three categories and measured the size based on these categories:

- *Unchanged*: For the business process, data groups in “no change required/used as is transactions” are counted
- *Changed*: For the business process, data groups in “change/customization required transactions” are counted
- *New*: For the business process, data groups in “new developed transactions” are counted

We calculate size as DTP for all the transactions in these categories and take to reach the size of the business processes.

This proposed size measurement method has five main steps, which are presented in Figure 1.

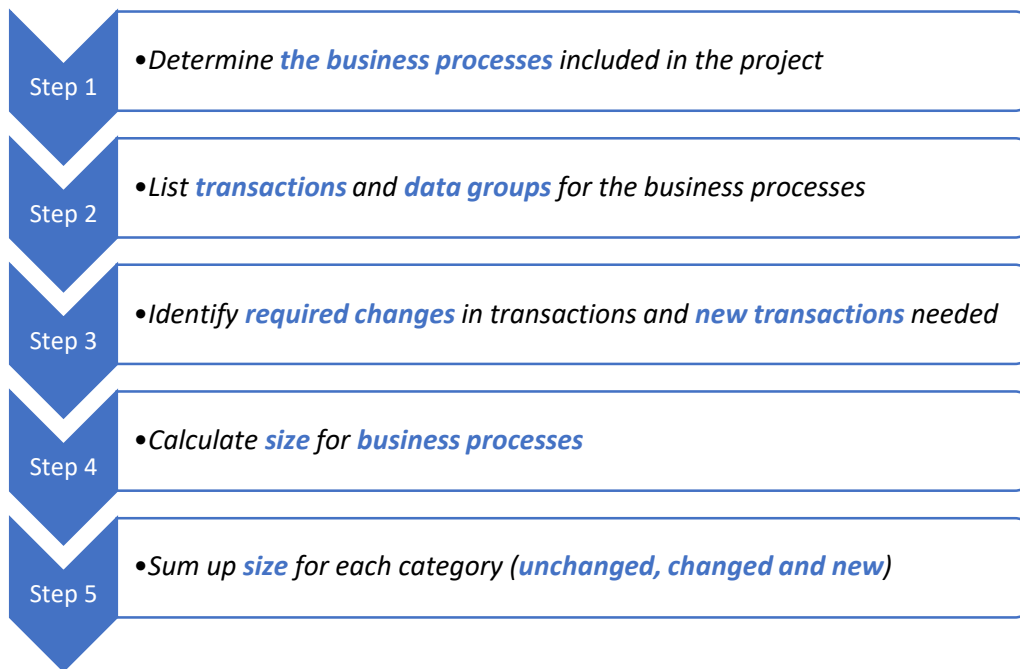


Figure 1. Steps of the size measurement method

3.1. Determining business processes

In an EA project, based on the customer requirements, several business scenarios are implemented. A business scenario includes many business processes. A “business process” is defined as a collection of operations that takes a particular business input and convert it into a valuable business output through a series of transactions [29].

In the scope determination phase of the EA projects, the business scenarios and corresponding business processes are determined. In this first step of size measurement, the “*Project Scope Document*” is reviewed, and “*Business Scenarios*” and “*Business Processes*” are listed for the project, as presented in Figure 2.

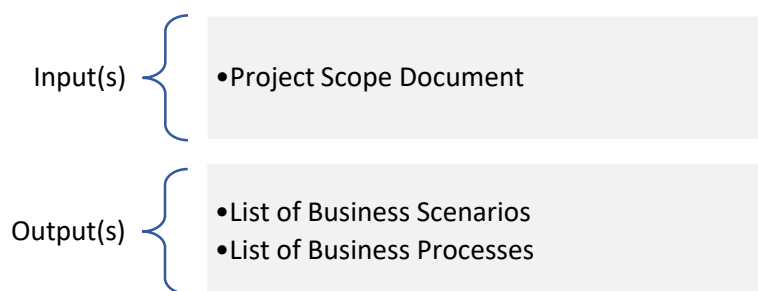


Figure 2. Determining business processes

3.2. Defining transactions and data groups

The second step of the size measurement is listing “*Transactions*” and “*Data Groups*” included in the business processes. For this step, the “*Business Process Repository*” of the EA could be used as the primary resource. Inputs and outputs of this step are presented in Figure 3.

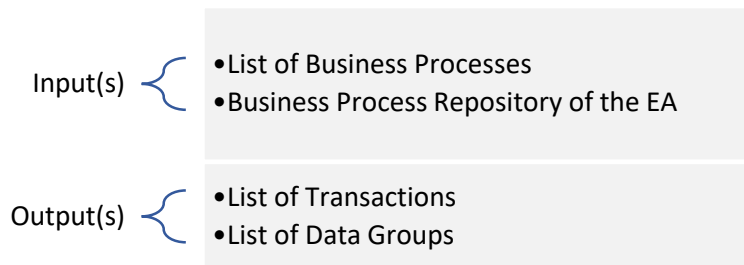


Figure 3. Defining transactions and data groups

A “transaction” is basically defined as executing a program in EA terminology, whether transactions are called via system-defined/user-specific transaction codes or invoked by other programs [30]. Many programs in EA are run by users with transaction codes, but also some programs are run by periodic jobs or called by integration programs. For example, a report can be created and sent to users via e-mail by a periodic program, or the included raw data can be transmitted to integrated systems by a program. Because in all of these programs, data groups are processed to comply business process requirements, we consider them as transactions, regardless of whether they are run by a user with a transaction code or not.

In EA, each transaction uses a number of data groups to fulfill the business request. A “data group” is defined as “a unique, non-empty, non-ordered group of data attributes, explaining the same one object of interest” [31]. In an EA, there exist two main data categories: master data and transactional data. Master data is permanent data that contains information for customers, suppliers, materials, etc., whereas transactional data is temporary data that contains information for purchase orders, invoices, etc. [29]. We include both categories in the definition of “data group”.

It may be helpful to use an example to explain the concept of a data group for an EA. In the “sales order processing” business process, “create sales order”, “change sales order”, and “display sales order” are the main transaction codes. We can list four main data groups as “Customer”, “Material”, “Pricing” and “Sales Order” for these transactions. “Customer”, “Material” and “Pricing” are master data containing the required information to create a sales order. “Sales Order” is the transactional data that occurred during the execution of the transactions. Consider a business requirement such as “If the customer is a local customer and the price of the material is higher than 100 \$, set the status of the sales order as to be approved”. In order to apply the required customization to fulfill this requirement, these four different data groups would be handled. Thus, we count all of these data groups in the size calculation.

3.3. Identifying required changes and new transactions

Depending on the customer requirements, some transactions are used as is, and some transactions are used by applying customization or code changes. If customer requirements cannot be met by the pre-built functionality of the EA, new transactions can be developed using the programming language of the EA.

In this step of the size measurement, change required transactions, new transaction needs, and related data groups should be figured out. The main input of this step is “The Business Blueprint/Functional Requirement Document” of the EA project. By examining in detail the Business Blueprint / Functional Requirement Document of the project, it should be determined which category (unchanged, changed, new) each transaction belongs to. Inputs and outputs of this step are presented in Figure 4.

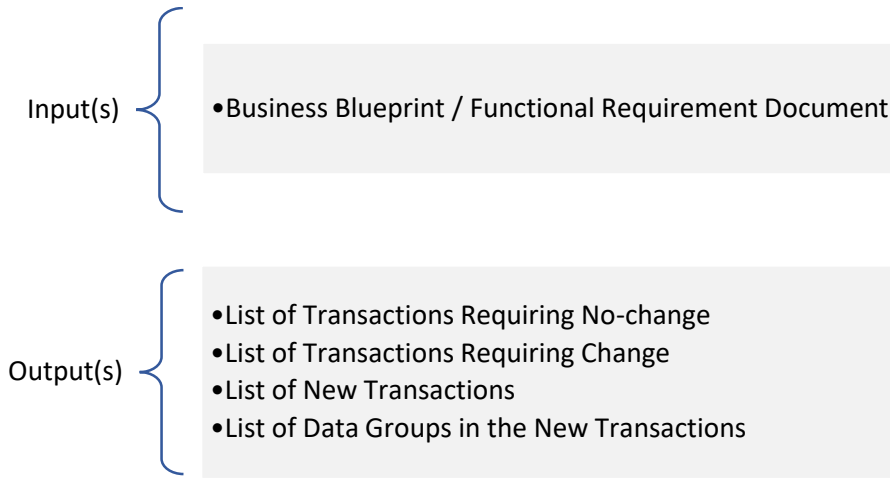


Figure 4. Identifying required changes and new transactions

3.4. Calculating size for business processes

For each business process, the size for each category is calculated as DTP by counting data groups in the transactions of that business process. The size as DTP for each category is calculated by using the following equations in which DG_i represents the total number of data groups in transaction i .

$$Size_{unchanged} = \sum_{i=1}^N u_i \cdot DG_i \quad (1)$$

$$Size_{changed} = \sum_{i=1}^N c_i \cdot DG_i$$

$$Size_{new} = \sum_{i=1}^N n_i \cdot DG_i$$

$$u_i = \begin{cases} \mathbf{1}, & \text{if transaction } i \text{ is an unchanged transaction} \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

$$c_i = \begin{cases} \mathbf{1}, & \text{if transaction } i \text{ is a changed transaction} \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

$$n_i = \begin{cases} \mathbf{1}, & \text{if transaction } i \text{ is a new transaction} \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

3.5. Calculating total size

The total size of an EA project is the total size of the business processes included in that project. Using the following equations, the total size of the project is calculated as DTP units in this step. BP used in the equations represents the business processes.

$$Size_{unchanged} = \sum_{i=1}^n Size_{unchanged} (BP_i) \quad (2)$$

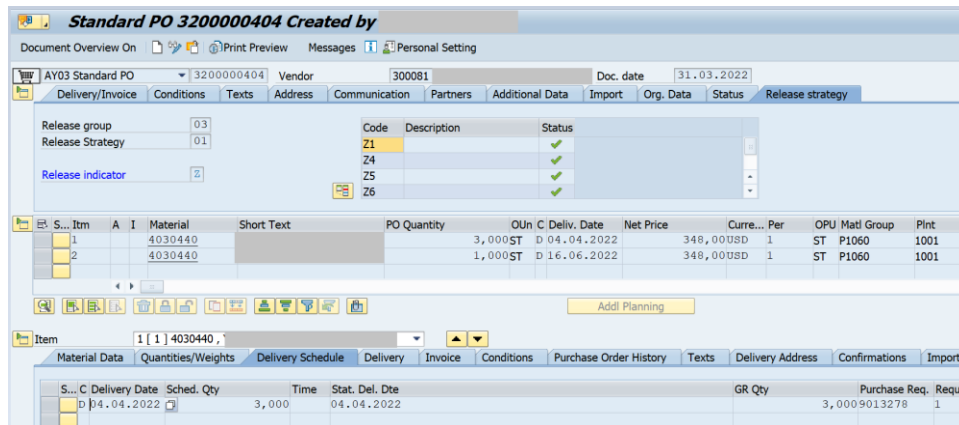
$$Size_{changed} = \sum_{i=1}^n Size_{changed} (BP_i)$$

$$Size_{new} = \sum_{i=1}^n Size_{new} (BP_i)$$

4. Sample size measurement

For a better understanding of the method, we applied the method on a well-known business process, the “Purchase Order” process. A “Purchase Order (PO)” is an official document showing the description, quantity, price, and purchase conditions of the ordered products or services. It is created by the buyer and forwarded to the vendor to start the purchasing process officially. The purchase order process is included in the MM (Material Management) Module of SAP. The main transactions run for this process are presented in Figure 7 in Appendix A.

In this process, the PO is created, changed, displayed, reported, and in most of cases an approval (release) step is applied. A sample view for the PO is presented in Figure 5.



The screenshot shows the SAP interface for a Standard Purchase Order (PO) with document number 320000404. The header includes fields for Vendor (300081), Document Date (31.03.2022), and Release Strategy (01). Below the header, there are tabs for Release group, Release Strategy, and Release indicator. A table lists release conditions with codes Z1, Z4, Z5, and Z6, all marked with green checkmarks. The main table displays two items: Item 1 (Material 4030440) with a quantity of 3,000 and a delivery date of 04.04.2022, and Item 2 (Material 4030440) with a quantity of 1,000 and a delivery date of 16.06.2022. The bottom section shows the 'Delivery Schedule' tab for Item 1, with a delivery date of 04.04.2022 and a quantity of 3,000.

Figure 5. SAP sample view for purchase order display

4.1. Transactions and data groups

We use the Project Scope Document to understand that a process would be implemented within the scope of that project. A sample section from the Scope Document showing that the PO process is implemented in the project could be as in Figure 6.

The purchasing process will be carried out through the system for both investment and non-investment purchases. MRP will be carried out through the system, minimum stock quantities for materials will be defined, and Purchase Requests will be opened automatically according to the needs. Purchase Orders will be created for purchase requisitions, and the approval process for both purchase requisitions and purchase orders will be carried out through the system. The printout of the purchase orders can be taken over the system or, if desired, sent to the seller by e-mail.

Figure 6. The scope definition for purchase order process

In order to list “Transactions” and “Data Groups” included in the business process, we use the “Business Process Repository (BPR)” of the EA and EA itself as the resource. As described in the method, we consider both master data and transactional data when determining Data Groups.

We firstly checked the master data list for the MM module defined in BPR of SAP, as presented in Table 4 in Appendix B, and determined the master data of the PO process as “Buyer”, “Vendor”, “Material”, “Conditions”, “Supplement” and “Release Strategy”. “Purchase Order” is the main transactional data of this process. As described in the scope, PO is created depending on the Purchase Requisition, so we determined “Purchase Requisition” as another transactional data.

We listed related “transactions” and “data groups” of the PO process as presented in Table 1.

Table 1

Data groups and transactions for the PO process

Business Process	Process Step	Data Groups	Transactions
Purchase Order Processing	Maintain Purchase Order	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	Create, Change, Maintain Supplement, Mass Maintenance, Display
Purchase Order Processing	Release Purchase Order	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	Individual Release, Collective Release
Purchase Order Processing	Report Purchase Order	Purchase Order, Buyer, Vendor, Material, Conditions	By Vendor, By Material, General, By Purchase Order

4.2. Required changes and new transactions

After listing “transactions” and “data groups”; change required transactions, new transaction needs and related data groups should be figured out. For this purpose, we use the “Business Blueprint Document” as the resource. The “Business Blueprint Document” includes both configuration and customization details of the project, and new transaction requirements are also defined in this document.

For this sample calculation, we listed “customer requirements”, “related transactions”, and “new transactions and their data groups” for the PO process in Table 2.

Table 2

List of the change requests for the PO process

PO – Required Changes & New Transactions	
CR.1	<p><i>“The purchase order will be only created by entering a purchase requisition as a reference document. If the entered purchase requisition’s approvals are not complete, the system will not allow the purchase order creation, and an error message will return stating that approvals are not complete”</i></p> <p>change in purchase order creation -transactions (create)</p>
CR.2	<p><i>“When a user tries to change a PO, the purchase order, the system will check the PO approval status. If any approval is given for the PO, a pop-up will appear saying, “PO is approved; if you save changes, SAS approvals will be initialized”. If the user approves, the change will be saved, and approvals will be initialized”</i></p> <p>change in purchase order change -transactions (change, mass maintenance)</p>
CR.3	<p><i>“When an approval for the purchase order is given, the next approver will be notified by e-mail”</i></p> <p>new transaction for PO release, same data groups as individual & mass release -transactions (e-mail)</p>
CR.4	<p><i>“A new PO good receipt report is required for PO showing PO details, good receipt document, batch numbers”</i></p> <p>new transaction for PO good receipt report -transactions (report PO good receipt) -data groups (purchase order, buyer, vendor, material, conditions, good receipt document, batch)</p>

4.3. Calculating Size

For the business process, we calculated the size of each category by using the size calculation equations. The calculation sheet is presented in Table 3.

Table 3
Size calculation for the PO process

Transaction	Data Groups	#Data Groups	u_i	c_i	n_i	Size unchanged (DTP)	Size changed (DTP)	Size new (DTP)
Create	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	0	1	0	0	7	0
Change	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	0	1	0	0	7	0
Maintain Supplement	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	1	0	0	7	0	0
Mass Maintenance	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	0	1	0	0	7	0
Display	Purchase Requisition, Purchase Order, Buyer, Vendor, Material, Conditions, Supplement	7	1	0	0	7	0	0
Individual Release	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	6	1	0	0	6	0	0
Collective Release	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	6	1	0	0	6	0	0
E-mail	Purchase Order, Buyer, Vendor, Material, Conditions, Release Strategy	6	0	0	1	0	0	6
Report by Vendor	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0
Report by Material	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0
Report General	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0

Transaction	Data Groups	#Data Groups	u_i	c_i	n_i	Size	Size	Size
						unchanged (DTP)	changed (DTP)	new (DTP)
Report by Purchase Order	Purchase Order, Buyer, Vendor, Material, Conditions	5	1	0	0	5	0	0
Report PO GR	Purchase Order, Buyer, Vendor, Material, Conditions, Good Receipt Document, Batch	7	0	0	1	0	0	7
						46	21	13

Based on these calculations, size of the PO process is as follows:

Size_{unchanged} = 46 DTP

Size_{changed} = 21 DTP

Size_{new} = 13 DTP

5. Conclusions

We developed a size measurement method for EA projects. In this method, we proposed measuring "changes" and "new" transactions where pre-built functionality does not fulfill customer requirements. We counted "data groups" in the categorized transactions. Claiming that a change in a transaction with one data group would not have the same effect on the size as a change in a transaction with multiple data groups, we defined a novel size measurement unit, namely Data Transaction Point (DTP), for EA projects. We measured size as DTP in three categories: "unchanged (no change required transactions)", "changed (customization required transactions)", and "new (new developed transactions)". We defined five main steps to apply this method as presented in Figure 1. We implemented the method on a well-known business process, "Purchase Order".

The main contribution of this study is the proposed size measurement method. This is a novel size measurement method developed by taking into account the unique characteristics of EA projects. We claim that this size measurement method can be applied by a novel user with general knowledge about the implemented EA. Using this method could lead to reducing size measurement variances and subjectivity. We believe that, as the method is used many times in an organization, productivity values will be more reliable, and subsequent effort estimations will be more accurate.

The second contribution is the novel size measurement unit DTP for EA projects. This specific size measurement unit can be used as a base unit for these types of projects. Since size measurement is performed at the business process level in the proposed method, size measurement unit DTP can be used for of the project management processes such as monitoring, planning, and control.

Another contribution of this thesis is the useful yet simple definition and application of the concepts "reuse" and "change" in EA projects. By making these definitions, we have revealed how EA projects can be evaluated.

6. Future Work

In order to observe the validity of the size measurement method, we plan to conduct a multiple case study with different EA projects. This size measurement method can also be applied in other change-intensive project types like software maintenance and upgrade projects. We also plan to extend our studies to evaluate the method's applicability for these types of projects.

There exist some EA project tools, such as SAP Solution Manager, which are also used to generate automatically Business Blueprint document. Since these tools comprise the required data for size measurement, the proposed method can be automated via these tools. This would reduce the overall time required for the DTP calculation.

7. Appendices

7.1. Appendix-A

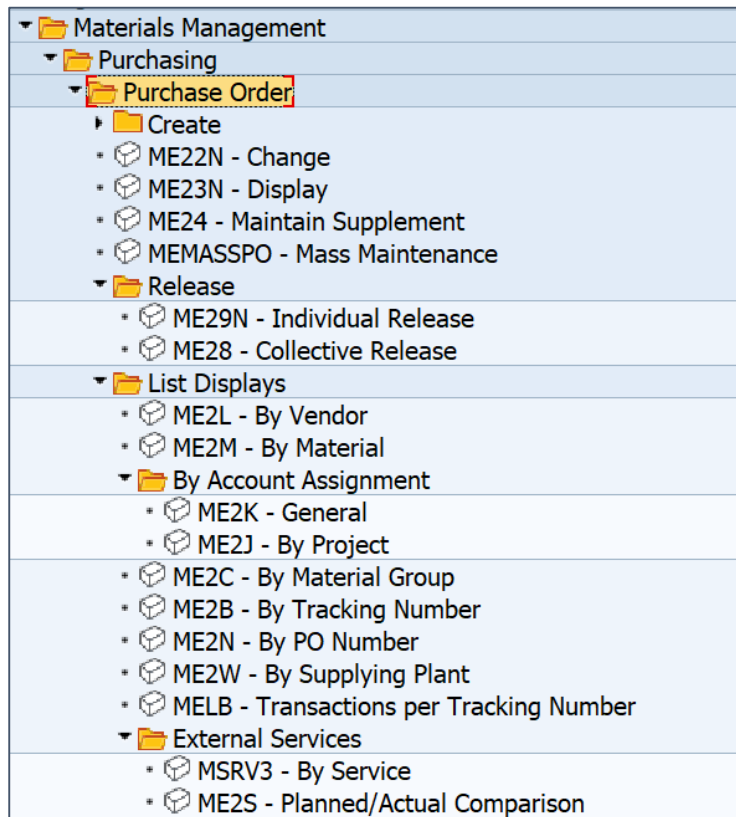


Figure 7. SAP sample view for transactions of the PO process

7.2. Appendix-B

Table 4

BPR master data list for the SAP MM module

Organizational Area	Object Type	Name	Package	Module
Procurement	Master Data	Buyer	BP_LIB_R3MM	MM
Procurement	Master Data	Conditions (Procurement)	BP_LIB_R3MM	MM
Procurement	Master Data	Contract (Purchasing)	BP_LIB_R3MM	MM
Procurement	Master Data	Delivery Address	BP_LIB_R3MM	MM
Procurement	Master Data	Manufacturer Part Number	BP_LIB_R3MM	MM

Organizational Area	Object Type	Name	Package	Module
Procurement	Master Data	Purchasing Info Record	BP_LIB_R3MM	MM
Procurement	Master Data	Quota Arrangement	BP_LIB_R3MM	MM
Procurement	Master Data	Release Strategy with Classification	BP_LIB_R3MM	MM
Procurement	Master Data	Service Condition	BP_LIB_R3MM	MM
Procurement	Master Data	Settlement Accounting for Conditions Requiring Subsequent Settlement	BP_LIB_R3MM	MM
Procurement	Master Data	Source List	BP_LIB_R3MM	MM
Procurement	Master Data	Standard Service Catalog	BP_LIB_R3MM	MM
Procurement	Master Data	Sustainability Information Record	BP_LIB_R3MM	MM
Procurement	Master Data	Vendor Evaluation	BP_LIB_R3MM	MM
Procurement	Master Data	Vendor Master Record	BP_LIB_R3	MM
Procurement	Master Data	Vendor Rebate Arrangements	BP_LIB_R3MM	MM
Procurement	Master Data	Vendor Sustainability Record	BP_LIB_R3MM	MM

8. Acknowledgements

We would like to thank the support of the Scientific and Technological Research Council of Turkey (TUBITAK) ARDEB 1001 [Project number: 121E389] program.

9. References

- [1] O. Volkoff, D. M. Strong, and M. B. Elmes, "Understanding enterprise systems-enabled integration," *Eur. J. Inf. Syst.*, vol. 14, no. 2, pp. 110–120, 2005.
- [2] B. Ramdani, D. Chevers, and D. A. Williams, "SMEs' adoption of enterprise applications: A technology-organisation-environment model," *J. Small Bus. Enterp. Dev.*, 2013.
- [3] T. H. Davenport, *Mission critical: realizing the promise of enterprise systems*. Harvard Business Press, 2000.
- [4] M. L. Markus and C. Tanis, "The enterprise systems experience-from adoption to success," *Fram. domains IT Res. Glimpsing Futur. through past*, vol. 173, no. 2000, pp. 173–207, 2000.
- [5] M. Daneva and R. Wieringa, "Cost estimation for cross-organizational ERP projects: Research perspectives," *Softw. Qual. J.*, vol. 16, no. 3, pp. 459–481, 2008, doi: 10.1007/s11219-008-9045-8.
- [6] W. B. Frakes and K. Kang, "Software reuse research: Status and future," *IEEE Trans. Softw. Eng.*, vol. 31, no. 7, pp. 529–536, 2005.
- [7] M. Mijač, R. Picek, and D. Andročec, "Determinants of ERP Systems as a Large-Scale Reuse Approach," in *MATEC Web of Conferences*, 2019, vol. 292, p. 3007.
- [8] L. Brehm, A. Heinzl, and M. L. Markus, "Tailoring ERP systems: a spectrum of choices and their implications," in *Proceedings of the 34th annual Hawaii international conference on*

- system sciences*, 2001, pp. 9-pp.
- [9] B. Light, "The maintenance implications of the customization of ERP software," *J. Softw. Maint. Evol. Res. Pract.*, vol. 13, no. 6, pp. 415–429, 2001.
 - [10] M. Keil and A. Tiwana, "Relative importance of evaluation criteria for enterprise systems: a conjoint study," *Inf. Syst. J.*, vol. 16, no. 3, pp. 237–262, 2006.
 - [11] "The 2022 ERP Report," *Panorama Consulting Group*, 2022. <https://www.panorama-consulting.com/resource-center/erp-report/>.
 - [12] M. Daneva, "Understanding functional reuse of ERP requirements in the telecommunication sector: An empirical study," in *Proceedings - 2014 Joint Conference of the International Workshop on Software Measurement, IWSM 2014 and the International Conference on Software Process and Product Measurement, Mensura 2014*, 2014, pp. 216–221, doi: 10.1109/IWSM.Mensura.2014.42.
 - [13] E. Stensrud, "Alternative approaches to effort prediction of ERP projects," *Inf. Softw. Technol.*, vol. 43, no. 7, pp. 413–423, 2001, doi: 10.1016/S0950-5849(01)00147-1.
 - [14] N. K. Omural and O. Demirors, "Effort estimation for ERP projects - A systematic review," in *Proceedings - 43rd Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2017*, 2017, pp. 96–103, doi: 10.1109/SEAA.2017.68.
 - [15] B. Ozkan, O. Turetken, and O. Demirors, "Software functional size: For cost estimation and more," in *European Conference on Software Process Improvement*, 2008, pp. 59–69.
 - [16] C. Gencel and O. Demirors, "Functional size measurement revisited," *ACM Trans. Softw. Eng. Methodol.*, vol. 17, no. 3, pp. 1–36, 2008.
 - [17] M. Daneva, "Preliminary results in a multi-site empirical study on cross-organizational ERP size and effort estimation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4895 LNCS, pp. 60–71, 2008, doi: 10.1007/978-3-540-85553-8_5.
 - [18] M. Daneva, S. Wettflower, and S. De Boer, "Uncertainty in ERP effort estimation: A challenge or an asset?," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5338 LNCS, no. i, 2008, pp. 208–222.
 - [19] W. Rosa, "Empirical effort and schedule estimation for enterprise resource planning projects," in *Proceedings of the 2012 Joint Conf. of the 22nd Int. Workshop on Software Measurement and the 2012 7th Int. Conf. on Software Process and Product Measurement, IWSM-MENSURA 2012*, 2012, pp. 190–197, doi: 10.1109/IWSM-MENSURA.2012.35.
 - [20] I. P. Erasmus and M. Daneva, "ERP effort estimation based on expert judgments," in *Proceedings - Joint Conference of the 23rd International Workshop on Software Measurement and the 8th International Conference on Software Process and Product Measurement, IWSM-MENSURA 2013*, 2013, pp. 104–109, doi: 10.1109/IWSM-Mensura.2013.25.
 - [21] P. Erasmus and M. Daneva, "ERP services effort estimation strategies based on early requirements," *CEUR Workshop Proc.*, vol. 1342, no. 1, pp. 83–99, 2015.
 - [22] P. Erasmus and M. Daneva, "An experience report on ERP effort estimation driven by quality requirements," *CEUR Workshop Proc.*, vol. 1342, pp. 126–139, 2015.
 - [23] F. Vogelezang, "Using COSMIC-FFP for sizing, estimating and planning in an ERP environment," in *16th International Workshop on Software Measurement and DASMA Metrik Kongress*, 2006, pp. 327–342.
 - [24] F. M. Téllez, "Solving the size estimation problem in ERP project context : the eEPC- COSMIC approach," University of Twente, 2009.
 - [25] I. P. Erasmus, "The COSMIC EPC method An ERP functional size measurement method delivering time and cost estimates," University of Gothenburg, 2012.
 - [26] M. Daneva, "Integrating reuse measurement practices into the ERP requirements engineering process," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4034 LNCS, pp. 112–126, 2006, doi: 10.1007/11767718_12.
 - [27] O. Demirors and N. Küçükateş Ömüral, "Exploring reuse levels in ERP projects in search of an effort estimation approach," in *Proceedings - 44th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2018*, 2018, pp. 191–197, doi: 10.1109/SEAA.2018.00038.
 - [28] N. K. Ömüral and O. Demirors, "Effort estimation methods for ERP projects based on function

- points: A case study,” in *ACM International Conference Proceeding Series*, 2017, vol. Part F1319, pp. 199–206, doi: 10.1145/3143434.3143464.
- [29] E. Monk and B. Wagner, *Concepts in enterprise resource planning*. Cengage Learning, 2012.
- [30] T. Orosz, “Analysis of SAP Development tools and methods,” in *2011 15th IEEE International Conference on Intelligent Engineering Systems*, 2011, pp. 439–443.
- [31] “ISO - ISO/IEC 19761:2011 - Software engineering — COSMIC: a functional size measurement method.” <https://www.iso.org/standard/54849.html> (accessed Oct. 22, 2021).