International Journal of Mechanical, Industrial and Aerospace Sciences
ISSN: 2517-9950
Vol:6, No:3, 2012

Operational risks Classification for Information Systems with Service-Oriented Architecture (Including Loss Calculation Example)

Irina Pyrlina

***Abstract*—**This article presents the results of a study conducted to identify operational risks for information systems (IS) with service-oriented architecture (SOA). Analysis of current approaches to risk and system error classifications revealed that the system error classes were never used for SOA risk estimation. Additionally system error classes are not normallyexperimentally supported with real-enterprise error data. Through the study several categories of various existing error classifications systems are applied and three new error categories with sub-categories are identified. As a part of operational risks a new error classification scheme is proposed for SOA applications. It is based on errors of real information systems which are service providers for application with service-oriented architecture. The proposed classification approach has been used to classify SOA system errors for two different enterprises (oil and gas industry, metal and mining industry). In addition we have conducted a research to identify possible losses from operational risks.

***Keywords*—**Enterprise architecture, Error classification, Oil&Gas and Metal&Mining industries, Operational risks, Service-oriented architecture

I. Introduction

EACH year organizations look for ways to improve and optimize internal processes and procedures aimed at growth of sales revenues, improvement of quality and increased employee productivity. Having said this, such measures should still comply with the strategy of reducing operatingl costs. To reach these goals new technologies and systems are used. Manufacturing and engineering systems are modernized alongside systems for auxiliary processes (like finance, human recourses, accounting etc.) and platforms to support them like information system with service-oriented architecture (SOA).The task of efficiency assessment for new technology or system architecture (such as SOA) requires a method for cost/benefits analysis. Normally information system costs contain implementation and support expenses (including software and hardware costs, human resources expenditures etc.)[1].Additionally intangible expenses such as losses from risks occurrences should be taken into account. Such risks are known as operational risks. They can be met in every IS implementation project and the total sum of operational risk losses can comprise a significant part of IT budget.

Today modern information systems provide statistical data to get more accurate estimation of operational risk losses. Causes of such risks in SOA IS implementations could be a lack of qualified resources (for instance, to support data quality, or to avoid interpretation of the results in a wrong way), unauthorized personnel actions, application errors, lack of memory, network or servers failures.

In this article we propose a method to define types of operational risks for information systems with service-oriented architecture. The results are applied to two enterprises with the help of IS error messages statistics. Over a 5-year period of research an oil&gas company gathered 820 error messages in different system modules convered by SOA applications. Similarly, a metal&mining company gathered 916 error messages during a 4-year period (2007-2010). Based on this data a method for SOA system errors classification was designed. The main objective of this activity is to propose the method to classify operational risks of service-oriented architecture implementation.

In this article operational risk[2] is considered to be potential losses from SOA information system errors such as software and hardware errors or technical resources failures.

The core element of service-oriented architecture is a service (business or technical). In this case a business service is an area of enterprise activities where SOA IS implementation can have a highest positive impact. Let's define a business service as an aggregation of

1. Functions consolidated by business critical criteria;
2. System functions wrapped up with web-services;
3. Technical resources which are critical for system work and required for web-services implementation (for instance, platform or server applications).

A technical service, according to its classical definition, is an autonomous, modularized, «self-describing» application, which combines a number of executable functions provided for every system-consumer [9]. In our approach a technical service is seen as application and technical components, which support work of business service with web-service concept.

Taking this into account the operational risks of service-oriented architecture can be defined on the basis of business service resources and their error types.

II. Types of operational risks

Operational risks in SOA based information systems are closely tied with errors in resources used by these systems, for instance with resources of business services. The following types of system resources could be identified (see. Fig. 1)

A. Human resources

Human resources are employees participating in the process or service. Fig. 1 shows the resources of a typical business service. From the human resources point of view each business process contains Line of business (LoB) representatives, experts responsible for key service operations,

Irina Pyrlina is with Higher School of Economics, 105187, Moscow, Russia (e-mail: Irina.pyrlina@gmail.com).

[1] TO estimate IS costs methods like TCO can be used [7].

[2] Common definition of the term «operational risk» can be found in [8, стр.185].

process and data owners, authors of requests, as well as administrators of service applications;

B. Software resources

Software resources are systems which have a possibility to provide/support technical services. This is not only business applications but also system software. The following types of software resources exist:

- Access module – diverse types of user interfaces for system activities («Thin» client on the basis of Web-technologies and «Rich» client on the basis of Windows-applications);
- Software (or applications) – an mathematical mechanism of the system;
- Middleware used to organize data exchange among applications and modules;
- System software – applications which support and control server operations;

C. Technical resources

Technical resources – hardware to support and store system resources.

According to the resources classes the following operational risk types can be identified:

1. Personnel risks stand for risks of losses from unauthorized, inaccurate or inappropriate work of personnel;
2. Software risks can be defined as risks of losses from software components failure. This type of damage is normally a result of software component disruption;
3. Technical risks represent risks from hardware disruption (e.g. any equipment where software is installed and operates).

One of the categories is software risk which is derived from software errors losses. Software error [5] stands for not achieved results of system activities or any deviation from forecasted output of the system.

Software errors contain:

- Any defects identified during system execution and data input;
- Incorrect system output;
- Incorrect activities of personnel which result in incorrect output.

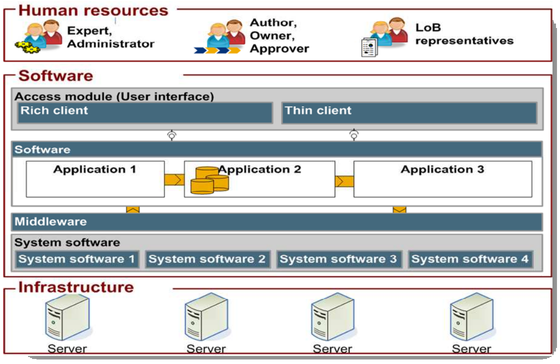


Fig. 1 Business service resources

Software risks are the largest category of operational risk. Therefore let us define software classification which expands proposed operational risks classification for information systems with SOA. As software risks depend on the software errors we can define or adopt existing software error classifications to serve as a basis for SOA error classes.

III. A Review of Software Error Classification

The quality of composite applications[3] consuming services of other systems depends on the number of errors not only in the composite application itself but also on the quality of services provided by other systems. Therefore to classify software risks for information systems with SOA it is sufficient to define the classification of software errors for service-providers. This classification as a result can be applied to both service-providers and consumers.

A great number of software error classification methods can be found in [1]-[4],[6]. The most popular among them classify errors per

A. Priority and level of impact

Classification per priority and level of impact (classification applied by SAP) uses the following types:

- Very high priority;
- High priority;
- Medium priority;
- Low priority;

B. Application type

Classification per application type uses classes where error weres made, (or as mentioned in [1],[2] per «place of error») with the following types

- user interface errors;
- Application errors;
- Middleware errors (or «errors of data processing and interpretation» as in [1]);
- System errors;
- Hardware errors [2];

C. Error Reasons

Classification of error reasons (as mentioned in the article [3] where the author proposes to use Beizer's classification) contains the following classes

- Functional errors;
- System errors;
- Process errors;
- Data errors;
- Code errors;
- Documentation errors;
- other errors, where the reason can't be identified;

D. Place of software lifecycle

Also there is an approach which proposes to classify errors per place of software lifecycle (according to Government Standard 34.601-90 the software lifecycle stages can be the following – Gathering requirements, blueprint design, technical project design, document system operations, system Go-live, system support).

[3] We define composite application as a system which uses data and functions provided by the different platforms and systems as services. It combines these services to support user oriented processes and views.

However this approach of classification is not applicable in our case. Such classification is more generic and does not reflect SOA specifics.

IV. Classification of SOA System Error and Operational Risks

Almost all the classification methods reviewed for this research provide no statistical data to verify their efficiency. Only [3],[4] conducted a classification which included a demonstration of statistical research results for the manufacturing and IT industries. The goals of their research was to verify the proposed code classification and use the results for further custom system development. Classifications proposed in [3],[4] are either superfluous or oriented error classification to different stages of system design, implementation and testing. Additionally in [4] the drawbacks of existing classification methods are listed.

Our research is oriented to the analysis of error statistics for large systems and a classification proposed for systems like ERP (i.e., large standard solutions) in the enterprises system landscape. Classification is done per error type, priority, error frequency and complexity (which can be assessed according to the duration of error fixing).

We propose the following software error classification which can be applied to all types of SOA systems. The classification refers to three error classes and several sub-classes as follows

A. Main class

- Input-output errors (I/O or user interface errors) stands for errors in constant values or variables, and errors of input or output data;
- Functional errors in system code, or processing logic where transformation of input data is done;
- Middleware errors contain data exchange problems between different applications, distortion of the data during the transfer or errors in message exchange.
- Data errors represent errors of data change in any system component or data storage (this type of error is inherited from classification [6] where it is used for web-applications and is applicable for SOA systems as it has web-applications component);
- System errors software including server configuration errors, user access errors, productivity problems, operation system failure and hardware failure (including time-out problems[4], memory errors[5], network errors, database errors), and errors during installation and support of the system;
- Other errors (or ’not an error’) can be fixed without customization change or code modification. It contains all errors in system documentation and absence of system description.

B. Priorities

Error priorities are also maintained in the classification as:

- Very high (1) – errors have high business impact or effect on productive system operations, critical for core processes or function execution that result in failure of critical system;
- High priority (2) errors have considerable effect on the systems and business processes and result in visible productivity decrease;
- Medium priority (3) errors have an effect on system functionality and correspondingly can influence business operations, however they are not critical for the business;
- Low priority (4) errors are not show-stoppers for the business. E.G. user interface errors that do not prevent system use, however, they are inconvenient for users and should be fixed.

C. Duration

Duration of error fixing is a class to weigh the complexity of error which is estimated according to the time required to fix it. The duration is counted from the creation date of the error message to the date when the customer confirms the problem is fixed, i.e. the date when it’s status is changed to «confirmed».

The proposed classification method was used in 2 enterprises to gather and classify error messages from the systems with service-oriented architecture. Table 1 illustrated the results for one of the enterprises. It operates in oil and gas industry and has a complex IT landscape. Due to the large number of messages Table I contains just extract data of the 820 messages gathered and classified.

For each error message the priority data is gathered as well as the time of message creation and date of fixing. This gives the data for further analysis of error complexity and importance. Statistics for 2006-2010 were analyzed for both oil&gas and metal&mining enterprises. The resultant number of errors is shown in Table II.The groups of gathered errors are shown in Table III and on Fig.2. However, Table III contains an extract of the statistics with examples for each error type. Figure 2 illustrates total number of errors per error type. The results show that during the 5-year period the largest number of errors was recorded for functional and system errors. Statistics proves that newly implemented systems for large enterprises have some drawbacks and functional errors which can only be identified and eliminated after system execution starts, i.e. dutring the go-live phase.

It can also show that wrong assumptions for hardware, or software functionality were made during the system design phase. We should emphasis that statistics is gathered for first five years after systems go-live date. Fig. 2 and Fig.3 show statistics for both enterprises with error type distribution and reflection of errors priorities. Fig.2 shows the number of errors and color signifies error priority. The largest number of errors have medium and high priority. Errors with very high or low priority are rarely identified. Fig.3 shows average duration of error fixing for each type, which, is quite small. The longest duration was recorded for “not an error” type and is26 days. Input-output error fixing process lasted 22 days on average and systems errors took 21 days to beeliminated. The minimal duration of error fixing is recorded for “middleware” error type and is 10 days on average.

[4] We define time-out as an unexpected break of system execution as a result of excessive load on the system.

[5] Memory errors are errors connected with incorrect main memory handling.

TABLE I
SOFTWARE ERROR OF OIL&GAS COMPANY 2006-2010 (EXTRACT)

№	Message text (short)	Class	Priority
1	Credential for the Adobe Interactive Forms scenario	I/O errors	4
2	Anonymous users don't see KM content	I/O errors	3
3	Error with system data editing in support portal	I/O errors	3
4	Not all components are inserted after BOM explosion	Functional errors	2
5	Create new cancellation document after cancellation	Functional errors	2
6	Report 'Price Comparison' fails	Functional errors	2
7	Different configuration screens in Integration Builder	Middleware errors	4
8	Middleware - Bdoc validation error	Middleware errors	2
9	integration Directory-Communication Channel not found	Middleware errors	3
10	Authority for master data in BI	Data errors	3
11	master data time interval error	Data errors	2
12	product catalog replication	Data errors	2
13	Clustering information in SDB	System errors	2
14	High CPU consumption by a server node	System errors	2
15	OutOfMemoryError in during export	System errors	3
16	SAP Solution Manager Preparation Service	Not errors	3
17	Incorrect create SD message from ECP system	Not errors	3
18	Test message	Not errors	4

TABLE II
STATISTICS COMPARISON OF TWO COMPANIES

Year	Oil & Gas company	Metal & mining company
2006	106	0
2007	153	21
2008	200	207
2009	166	327
2010	156	264

TABLE III
STATISTICS PER SYSTEM ERROR TYPE

Error type (2006-2010)	Error number						%
	2010	2009	2008	2007	2006	**Totally**	
1.I/O error	21	21	95	22	3	**168**	10
2.Functional error	365	269	166	68	62	**787**	48
3.Middleware error	79	26	21	12	2	**110**	7
4.Data error	29	37	19	9	10	**92**	6
5.System error	170	114	86	53	24	**410**	25
6. Not error	9	26	20	10	5	**71**	4

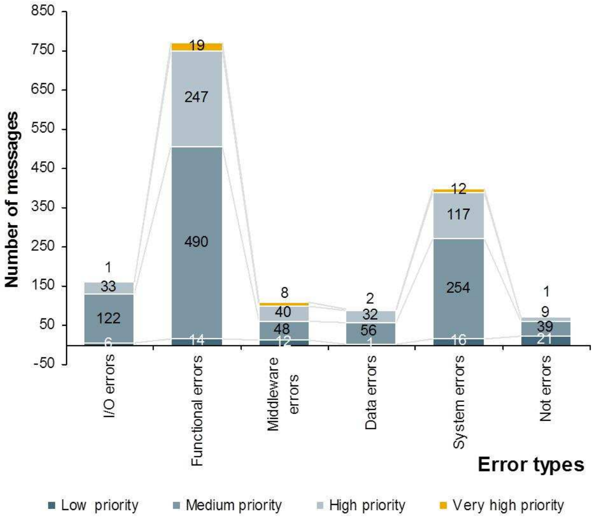


Fig. 2 Error types, priority and status (2006-2010) part 1

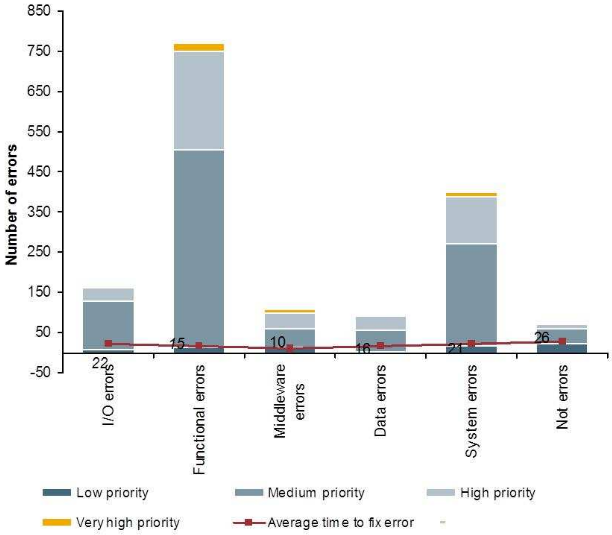
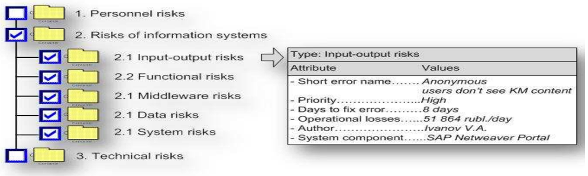


Fig. 3 Error types, priority and status (2006-2010) part 2

Fig. 2 and Fig.3 show statistics for both enterprises with error type distribution and reflection of errors priorities. Fig.2 shows the number of errors and color signifies error priority. The largest number of errors have medium and high priority. Errors with very high or low priority are rarely identified. Fig.3 shows average duration of error fixing for each type, which, is quite small. The longest duration was recorded for “not an error” type and is26 days. Input-output error fixing process lasted 22 days on average and systems errors took 21 days to beeliminated. The minimal duration of error fixing is recorded for “middleware” error type and is 10 days on average.

We propose to extend the classification of operational risks and apply it to information systems with service-oriented architecture on the basis of the system error classification described earlier in this article. Service-oriented architecture inherits the restrictions and opportunities derived from the service-provider. The majority of SOA technical features, including service orchestration can be assigned to the “Middleware risks”, interface issues fall into the category “Input-output risks”, “Functionality risks” will categorized by logic errors of the SOA application (or composite application), other risk types are used without any changes. Therefore every identified risk type corresponds to SOA operational risk types with exactly the same name of risk but a slightly changed definition. As shown in Fig. 4 core attributes of each type of risk contain information about risk priority and duration of error fixing. Classification of SOA operational risk (or SOA systems errors) is illustrated on Fig. 4.



Type: Input-output risks

Attribute	Values
- Short error name…….	*Anonymous users don't see KM content*
- Priority………………..	*High*
- Days to fix error………	*8 days*
- Operational losses……	*51 864 rubl./day*
- Author…………………..	*Ivanov V.A.*
- System component……	*SAP Netweaver Portal*

Fig. 4 SOA Risks classification

Using types of operational risks for information systems with service-oriented architecture we can conduct the analysis of operational losses.

V. Operational Losses When SOA Risks Occur

To identify losses attributable to risks a survey was conducted with the goal of estimating the effect if the solution for a particular error was not found for several days. SAP experts could estimate at least the average time when the system with error is used by the customer per day and average number of users per system. This type of assessment was done for extraction of the error list shown above. Experts' estimations were used to calculate final losses which are reflected in Table 4 and estimated in Russian Rubles per day (about 30 rubles per 1 USD).

Losses were calculated using three options of person-day costs. Each option corresponds to three roles available for system usage (engineer, chief engineer and manager). Different error priorities were created by User Groupson and these where shown to influence the work of different personnel levels. The following dependence of the message priority per user title was found to exist on average

- Low level of priority of error corresponds to the cost of 1 engineer day work;
- Medium and high priorities of errors cost of chief engineer day can be used;
- Very high priority is assessed as cost of a manager day.

$$Losses = U \bullet Q_U \bullet W_t \qquad (1)$$

Where
U – a percentage of system function use per day;
Q_u – number of employees, using this function;
W_t – cost of person-day according to the role and error priority.

Additionally, Table IV shows the total number of days required to fix an error. This provides a view on the average sum of operational losses an enterprise may encounter with during SOA system implementation and use.

TABLE IV
OPERATIONAL LOSSES OIL&GAS COMPANY 2006-2010 (EXTRACT)

№	Short text	Error type	Priority	Fix error (days)	Losses (Rubl./Day)
1	Credential for the Adobe Interactive Forms scenario	I/O error	4	1	5 186
2	Anonymous users don't see KM content	I/O error	3	8	51 864
3	Error with system data editing in support portal	I/O error	3	8	51 864
4	Not all components are inserted after BOM explosion	Functional error	2	21	66 682
5	Create new cancellation document after cancellation	Functional error	2	11	66 682
6	Report 'Price Comparison' fails	Functional error	2	36	133 364
7	Different configuration screens in Integration Builder	Middleware error	4	4	15 559
8	Middleware - Bdoc validation error	Middleware error	2	1	66 682
9	integration Directory-Communication Channel not found	Middleware error	3	2	77 796
10	Authority for master data in BI	Data error	3	39	25 932
11	master data time interval error	Data error	2	1	66 682
12	product catalog replication	Data error	2	5	33 341
13	Clustering information in SDB	System error	2	14	51 864
14	High CPU consumption by a server node	System error	2	7	518 636
15	OutOfMemoryError in during export	System error	3	4	2 593

VI. Conclusion

The article proposes a method of operational risk classification for information systems with service-oriented architecture. It was defined that service recourses and software classifications be used as a backbone to classify operational risks. This expands operational risks classification and simplifies the search, gathering and statistic analysis for operational risk. Error data from two enterprises helped to identify elements which distinguish SOA systems within other software systems. Web-interface, middleware and the components of service-providers are the distinctive characteristics and correspondingly the classes of new method where the critical errors to run the SOA system can appear.

This leads to the risk categories. Additionally the estimation of the operational losses was done for the group of errors to justify the losses calculation method according to the operational risk classification. Defined types of operational risks will be used further to simulate operational risks in SOA and statistics will help to verify the simulation model.

Acknowledgment

The author acknowledges Prof. Dr. F.Aleskerov for the statement of the problem.

References

[1] S. Kaner, G. Folk, E.Nguen, „Software testing. Fundamental concepts of business application management", *Diasoft*, 2001.

[2] G. Mayers, *Software reliability*, Mir, 1980.

[3] H. Krawczyk, B. Wiszniewski, H. Mork, "Classification of software defects in parallel programs," in *HPCTI Progress Report* no 1, Apr. 1995.

[4] T.J. Ostrand, E.J. Weyuker, "Collecting and Categorizing Software Error Data in an Industrial Environment," in *The Journal of Systems and Software*, vol. 4, 1984, p. 289-300.

[5] IEEE Standard Glossary of Software Engineering Terminology, *IEEE Std 729-1983*, *Inst. Electrical and Electronics Eng.*, New York, 1983.

[6] Y. Guo, S. Sampath, "Web Application Fault Classification—An Exploratory Study," *ESEM '08 Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement ACM New York*, NY, USA, 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1414060

[7] "SAP TCO Framework: A Framework to Reduce Total Cost of Ownership," *SAP White Paper*, SAP AG, 2005.

[8] F.T.Aleskerov, I.K. Andrievich, G.I. Penikas, V.M.Solodov, *Analysis of mathematical models of Basel II*, Moscow, Phismatlit, 2010, 288 p. (in Russian)

[9] H. Haas, A. Brown, "Web Services Glossary", *W3C Working Group Note,* Feb. 2004, [Online]. Available: http://www.w3.org/TR/ws-gloss/

I.Pyrlina was born in Moscow on 30th November 1984.. She has anBusiness Informatics Specialist's Honors Degree in applied informatics in economics awarded by the State University Higher School of Economy, Moscow, Russia, 2007. Her specialization is the design and development of enterprise information systems. In 2005 she participated in an international Russian-German project called "Process Driven Business Value Assessment of Information Systems implementation effect (automotive industry)". The project focused on SAP Business consultancy operations and provided a business value assessment methodology and tool of IT-investment tool for use in the preliminary stages of IT-projects (SAP Solutions). In 2006 she joined SAP and initially worked as a Business Consultant gaining experience in process driven business value assessments of Information Systems implementationt (namely from SAP solutions), value engineering projects, SOA Feasibility Studies and Roadmaps, Proof of Concept projects (Master Data Governance Discovery Workshops). She is experienced in SAP EDM and SOA areas. In 2006 she published the article "Efficiency estimation of information system on preliminary stage of implementation project" (Moscow, Russia: Automation and modern technologies, vol.10, 2006, in Russian). She took part in international conference in artificial intelligence systems (AIS'06) and intellectual CAD (CAD'06). From 2008 she moved to Business Technology – EA&EDM department and worked as Senior Technology Consultant. She established an Enterprise Architecture practice and led Enterprise Architecture design projects. In 2010 she published an article "Enterprise architecture and SOA" (Moscow, Russia: *Open Systems Journal*, vol.2, 2010). Since 2011 she has worked as an Enterprise Architect for the SAP company itself. Currently she studies practical implementation methods for SOA realization, multi-criteria decision making process for SOA and risk analysis from SOA projects.