Quality Assurance Model in IS Design and Development

Michael I. SMIRNOV

GMD FOKUS (Berlin)
(On leave The Saint-Petersburg State Electrotechnical University, Research Institute for Modelling and Intellectualization of Complex Systems - IMICS)

Abstract

The proposed approach to Individual User Requirements (IUR) estimation aims at IS quality assurance and employees the quality assurance model. The model provides some structure for the quality of service (QoS) area regarding the IS Design and Development (D&D) process. Consequently the User viewpoint on QoS could be matched with that of the Designer. Quality assurance assumes a sequence of steps. First, the IS quality area is to be partitioned into quality issues relative to the User viewpoint presentation; then quality issues should be estimated by the User representatives, and, finally, obtained issue estimations should be allocated in the knowledge distribution map within the IS. Actually, this mapping deals with the alignment of two models - for the QoS domain and the D&D domain respectively. A model is presented by a sample QoS area structure and quality issues.

1. Introduction

Following known methodologies of the IS Quality Assurance we have to admit that estimation of real user requirements should be considered as one of the most serious problems [4, 11, 13]. The IS "user" is a set of users with their own requirements, expectations, skills, location both in space and in organizational structure, etc. This set of users potentially produces a set of unknown Individual User Requirements for the IS services [s]he would like to obtain. The IS Designer can gain lots of useful considerations out of these requirements if the latter could be available at the very beginning of the design process when they are mostly needed. Some driving forces are usually involved instead of real users, such as Top Management, IS Management, IS Staff, IS development staff and possible external consultancy, not excluding another relevant and available sides. Resulting IS quality thus achieved reflects quality estimations of those involved in the process of the IS Design and Development. "If information - and not the technological infrastructure - is managed at all, this is done ("by default") by persons competent for managing the technology", [5]. It seems practically useful to make some assessments of the IUR during the D&D process in order to facilitate actually involved sides to take care of the "end-user" satisfaction. Thus "IS Quality" achievement could be transformed to the real quality of the IS Service for real users.

The paper is organized as follows. In section 2 major contradictions between IS User and IS Designer are pointed out in parallel with the Quality Function Deployment (QFD) presentation. The QFD methodology aims at aligning of both the User and the Designer viewpoints and will be considered in this paper as a background methodology for the quality assurance model. Section 2.1 presents the User viewpoint introducing in a formal way possible structure for the User requirements. The Designer viewpoint is considered in section 2.2 together with the QFD basic tool. Section 3 presents major considerations for the quality requirements structuring, and section 4 gives some practical results and their discussion. The paper is concluded by possible further research areas. Appendix 1 presents a sample quality questionnaire used in this research.

2. Background and Viewpoints Presentation

Strategic IS planning starts with the strategic planning of system development projects. Decisions on detailed requirements management and analysis are made at this stage with the help of meetings between various groups of designers and decision-makers. These group meetings serve to resolve conflicts reflecting different viewpoints: conflicts always exist due to conflicting requirements for systems and due to different visions of future systems. Providing a common structure within which the group may work to develop a common vision will help to enhance the quality of group process. Such a structure could be of high importance enabling alignment of the IS quality issues for both users and designers communities.

One of the recent papers on quality engineering [15] deals with the Quality Function Deployment method existed since 1967. According to [15] “QFD is an integrated set of tools for recording user requirements, technical specifications that satisfy these requirements, and trade-offs that might be necessary among these technical specifications, and several other items that might help in the system development process”. The method known worldwide since 1988 has been investigated by more than 200 United States corporations. While the structure provided by the QFD can be of significant benefit, the tool can be very difficult to use [15]. It utilizes the basic matrix approach (the QFD specific matrix was named the Quality House due to it’s shape, while set of matrixes is known as Quality Street and even the City of Quality). The basic matrix aims at matching the User requirements with the so called Design Parameters. The QFD history, references and hypertext extension could be found in [15].

Let us summarise the QFD features in a following list together with some comments.
1. The method tries to assign responsibility for producing a quality item to all parts of a corporation.
Comment: this feature requires design and maintenance of a corporation-wide knowledge map.
2. While the final goal of the QFD is to economically produce goods and services which satisfy customers’ requirements it addresses mostly the design and development proc-

ess.
Comment: applying the QFD methodology properly to the IS D&D means addressing the final goal of a corporation, not specific IS goals: the user voice is of prime importance here.
3. QFD makes the design improvement possible providing charts that document the design process.
Comment: the structure which the method brings is to reflect different user groups within the organization; this could be done by additional structuring of user requirements.
4. QFD is easily understood as a formal structure, all the QFD methodologies have the matrix shown in Figure 1 in common as one of their charts or subcharts.
Comment: the user requirements grouping in a set of charts is not obvious and may cause main problem for the QFD usage; this paper tries to attack this problem providing for preliminary user-based evaluation of requirements adding their classification and ranking.
5. QFD provides an overview of the entire project without overwhelming the general manager with excessive detail.
Comment: this feature should be supported by common understanding of what requirements should be considered as essential and what as overwhelming detail; here again we need additional structuring of user requirements.

Operating the basic QFD matrix demands at least 4 steps. The first step is to determine and organize a list of end-user requirements, where “to organize” means to group these requirements into appropriate categories. The second step is to prioritize these requirements. The third step is to develop a list of design parameters or specifications. The last step is to determine which parameters influence which requirements. In this paper two first steps reflecting major user involvement will be addressed. Some structure will be introduced at the first step and a mechanism for the second step will be presented.

Powerful software and lots of tricky engineering are involved usually in the process of “hiding” real User requirements within the domain of Design considerations. It is a fairly known procedure: when looking at unstructured and highly stochastic business area from the well structured and highly formalized design area every Designer will tend to explain the complexity of the real life with tools available at the moment and known well from the previous design. However the problem is such that Design tools are good enough only for the Designer. Let us continue employ “the User” and “the Designer” in this paper assuming that actually both of them exist virtually representing the two communities.

Two different viewpoints are to be aligned within the QFD methodology: the first one will be the User viewpoint and the second one - the Designer viewpoint. From the very beginning as the viewpoint concept is presented the author have to make clear that this paper is also an example of introducing of a highly formalized framework of Open Distributed Processing [10] to the highly unstructured and informal area of human collaboration within the process of IS D&D. In the given context the User viewpoint should be business oriented, aiming at precise definitions of objectives, functional requirements and policy statements. The Designer viewpoint aims at a proper design construction. According to the ODP standards the User viewpoint should be valid only at the so called Enterprise level, while all other levels are to be comprised with the Designer's viewpoint components.

2.1 The User Viewpoint

The User viewpoint could be introduced via objectives, functional requirements and policy statements; for the purposes of this work it will be valuable to derive some structure for the user requirements from the viewpoint presentation. Needed structure will have three domains - System, Designer and User. Each domain will be filled with a number of requirements; some of them could be seen as indifferent to a particular business area, organization or company, while the others are deeply influenced by these. Therefore let us consider each of the domains consisting of two subdomains, namely, Special and General to reflect this difference.

Objectives: this is a number of *business purposes* in favour of which the current IS design is undertaken; these purposes can appear in a form of needed business activities supported or/and provided by future IS. It is important to note that User, when defying the purposes relies on existing understanding of current business, not changed yet by future IS implementation. In this sense the User formulates very *soft requirements* for implementation of these purposes (as a rule he understands this clearly and is not very insistent on his *business* purposes). The Designer, on contrary is focused on a specific IS purposes and namely on that which are the objective of *his* design. At this level it is very important to make sure both communities that they have essentially different understanding of system objectives and need collaboration for their alignment. In these paper we address this process to a System domain.

Functional requirements: the User speaks his own language and lacks ability to find needed technical explanations and explicit criteria for functional requirements acquisition. We may try to get User's vision of specific IS features in a form of requirements, making him sure that these belong to a Designer domain.

Policy statements: the User is eager to speak about policies which he is aware of - accounting, staff management, product quality checking, etc., however he has to be able to speak about policies supported or/and provided by IS, such as information access, information update and validity checking; information authoring and ownership, etc. In this paper we address this process to a User domain.

According to [10] this viewpoint should be considered as the Enterprise Viewpoint, however the Designer also is eager and able to formulate his own Enterprise Viewpoint. To avoid some possible mistakes and misunderstandings let us consider that the User and the Designer are able to produce a common vision for the Enterprise Viewpoint, however it should be named the User Viewpoint. Even an "ideal" common vision implemented in the IS and able to influence the human knowledge [14] should initially be based on the User viewpoint. When Designer creates his own understanding of business which is to be supported/provided by future IS he produces lots of top-down specifications such as Data Flow Diagrams and Entity Relationship Diagrams. Let us imagine that these documents (even in electronic form in the case of any CASE tool usage) exist in two parallel versions. One version belongs to the User and is produced in Business language and reflects the User's understanding of what the business is about and how it operates. However the User always explains business individually, i.e. the User Viewpoint could be complete enough if it is presented as a set of Individual User Viewpoints (IUV). A set of IUV is a set of overlapping frames each having its own vision of reality; this vision is based on personal knowledge about business and organization. Another version belongs to the Designer and is produced in a form of specifications aiming at code generation. The main difference between these two approaches is emphasis on partially integrated (overlapped, in case of IUV) presentation and fully integrated (in the Designer' case).

Actually both presentations are valid and their alignment would be possible if the Designer could sequentially take each IUV for translating it into his own language. Almost perfect alignment could be achieved if the User himself would be involved during this process and all inevitable contradictions would be solved with the involvement of corresponding management level.

2.2 The Designer Viewpoint

The Designer viewpoint could be introduced at various levels (ODP viewpoints) - Informational, Computational and Engineering - due to existing experience of system engineering.

<u>Informational level</u> (viewpoint): it is taken for granted that corporate information resources which will be managed by the IS should be presented as information elements. Each information element (object) has it's own information structure; these structures in turn are defined in general by requirements for information handling. At this level services should be determined, however they would be defined as "functions, provided by object at computational interface" [10]. No User presence with close connections to the business area could be found at this level. However software classes and objects reflect the real information resources of the organization, and their methods are derivatives of typical templates for business activities defined by the User.

<u>Computational</u> and <u>Engineering</u> levels let us no chance to find suddenly a widely open window into the business area: all "Windows" are open inside the IS.

Let us take again the QFD basic matrix as an example illustrating the User and the Designer approaches, which should be aligned (Fig.1). With no doubt the QFD methodology has a lot of sense for this process but along with proper structuring, i.e with some preliminary arrangements of both the requirements and the parameters..

Design parameters (DP#)

User Requirements (UR#)

	DP*1*	DP*2*	DP*3*	DP*4*
UR*1*	✔			✔
UR*2*	✔			
UR*3*	✔	✔		✔
UR*4*				
UR*5*				
UR*6*		✔		
UR*7*	✔			

Fig. 1 A Quality Sample Deployment (QSD) matrix (after [15]) shows relationship of qualitative user requirements (desired customer functionalities) with design parameters (engineering characteristics). The symbol ✔ indicates that the given EC*i* addresses the respective UR*j*;

When observing such a matrix several questions could be produced from each of the viewpoints. First, extra parameters could appear in the matrix: the design parameter DP*3* matches in Fig.1 no user requirements, and, secondly, some user requirements could have no matching design parameters (UR*4* and UR*5* in Fig.1). Let us imagine that initial User requirements are presented to him for editing and evaluation in a form of a questionnaire. If the QSD and the questionnaire are produced by the IS Designer (aiming at quality design) we can expect not to have the first case. We also might expect not to have the second case if the User is provided by a large enough (“*complete”*) questionnaire developed *a priory* by the Designer. Due to the *softness* of the User requirements (c.f. 2.1, Objectives) he will never argue against proposed issues - he knows that his business will be changed somehow by future IS but lacks knowledge on possible change management. At maximum the User will omit some of the issues or combine the others. As a result the Designer viewpoint will dominate from the very beginning of the D&D process and “the discovery that the forms must be modified for the organization can be a major problem” [15]. That is why the proper starting point should be the User requirements structuring and evaluation by the User himself. This paper proposes a common questionnaire for this problem with some structure sheltered in it. On the other hand the User can *create and structure his requirements* independently of the Designer’s viewpoint thus providing the necessity for joint discussion of the content of the User requirements.

Providing a User with preliminary structure of his requirements could be seen as a sort of outsorcing. However nobody in contemporary IS/IT world can afford to avoid outsorcing. The outsorcing becomes a vital part of information technologies which highly utilize networking and object distributed platforms. To give an example consider the software house providing software packages and platforms to it's customers and keeping track of *all* basic objects of customer applications *during the IS lifecycle*. Another example could be seen in networking: some network operators consider possibility of outsorcing for the network management.

To confirm the list of possible individual user requirements, to integrate them and to achieve mutual understanding of the final list for both User and Designer communities several joint discussions are needed. Such discussions (workshops, round tables, brainstormings, etc.) inevitably will turn to the IS strategy definition. The IS strategy definition is very relevant to the IS quality of service. Once again we can compare typical viewpoints of the User and the Designer, or what they should be. The User viewpoint on the IS strategy has to be focused on the Change Management Programme. The strategy (together with appropriate descriptive tools [1]) could be considered as a methodology for conceptual design of a system needed in future. This conceptual design starts with the proper understanding of the current system. The Designer's viewpoint on the IS strategy is *initially focused on future* that's why the Designer "is always faultless"; he has no need to look for the business improvement; he works in a virtual space.

3. Quality requirements structuring

A good example of possible scenario for the quality requirements structuring could be found in [3] which employs a typical SODA programme for cognitive mapping. It is important that proposed technique combines formal and informal steps and finally tends to integrate a set of single cognitive maps into a group map. It was shown [6] that the approach is valuable for the User requirements definition. In the case of IS quality assurance it would be extremely useful to have such integrated vision of the User requirements.

The research idea which foundates this paper is very simple. Despite a number of standards and recommendations expressing existing experience, every particular IS is unique and requires unique identification of the key IS quality assurance issues. This could be done by very well known questionnaire approach [7, 12] based on Delphi methodology. The questionnaire approach was enlarged with the value adding processing technique by R.D. Galliers [2]. A typical questionnaire consists of a list of the "key issues", each provided with sufficient rationale. To answer the questionnaire means to estimate the key issue importance and problematic using the same scale. The issue criticality is then calculated as a product of average (for all non zero answers) importance and problematic. The criticality is used to range the issues.

It was shown by the author [9] that the issue criticality estimation could be compared in a sense of accuracy with the very thorough and time consuming examination. This prototype work aimed at evaluation of 34 IS proposals by 18 highly qualified experts. Each expert had to examine thoroughly 2 proposals (up to 100 pages each) with the use of individual questionnaire (more than 150 elementary estimations contributing the final integral individual range). In parallel with this work every expert was asked to estimate the importance and problematic of every of 34 proposals according to another group questionnaire. The group questionnaire consisted of 34 shortened titles of proposed projects and their rationalies. Each rationale was derived from the project objective. All proposals were ranked twice - according to individual and group ranks. In both rankings the most important “Top Ten” group was represented by the same projects.

The current research follows this approach for the key IS quality issues estimation. The research tasks could be described as follows:

1. To produce a list of possible key IS quality issues, thus *structuring the problem area* intentionally opposing the User’s and the Designer’s viewpoints;
2. To provide expert estimations for the list and to process the obtained data;
3. To assign priorities (ranges) to all issues treating them as issue “weights”;
4. To recall the *structure of the questionnaire,* reflecting some knowledge distribution map within the organization (initially this map is to be constructed but then to be hidden in a mixture of issues) and provide for integral estimation.

In this work the initial issue list was produced based on national standards for software products along with a number of bibliography sources of both Russian and foreign origin. Intentionally both Designer’s and User’s issues were combined in the questionnaire. When the estimation is made and corresponding data processed we can apply some structure to these data. As a result we can obtain average estimation of the User preferences for the given structure parts. Another possible way of expert data utilization - to divide raw data in accordance with a structure of the expert group. Imagine we have obtained questionnaire results from several groups of experts and each group represents sufficiently particular User level within the organization. In this case it will be useful to process data separately for each group and to apply the questionnaire structure to each group.

It is important that almost any structure could be derived from the estimated list of issues. This feature could be useful for a sensitivity analysis in IS strategy formulation when deciding on grouping of experts in order to present “single cognitive maps” [3]. Another useful utilization of obtained issue “weights” could be found within the formal QFD approach [15]. In the last case we can prioritize the User particular requirements in accordance with the User understanding of this requirement criticality.

4. Achieved results

It should be mentioned that the approach presented here is far from the real life approbation. Some reasons preventing broad approbation could be mentioned:

- lack of widespread understanding of the need for a quality assurance in the IS D&D;
- lack of IS information available for common use;
- shortcomings of national legislation and national standardization efforts in the IS area.

Results presented here were obtained within a framework of the research conducted currently in the city of St.Petersburg by the state research Institute for modelling and intellectualization of complex systems (founded 1992) within his state Centre for testing and certification of software products and information systems (founded 1994). More than 50 experts of Russian Universities were involved within the Russian state R&D programme “Informatics” aiming at design and development of IS and software components in education. The number of experts answering the questionnaire was 27. Obtained results should be treated as a testbed for future research in this field and could be applied to improve national standards in the area.

The purpose of the IS quality questionnaire is, first of all, to outline possible sphere of quality and, secondly, to make some provocative statements about quality thus forcing possible reader to prioritize them according to his current estimations and possibly add some new issues. All experts were provided with a short instruction explaining purposes of the research.

The questionnaire (Appendix 1) is intentionally unstructured and has following features:

- any issue in the questionnaire could be considered independently from all the others due to their total amount of 26 issues;
- the questionnaire presents a mixture of “key_for_sure” issues with “provocative” issues thus motivating experts to use low and high estimations in parallel and enabling them to find probably the “core” issue in accordance with their experience;
- questionnaire has several overlapping issues; however they were formulated in slightly different manner to meet various levels of experience;
- questionnaire has several issues presenting the same quality estimation from different viewpoints.

As it was mentioned earlier the quality questionnaire could be structured in several possible ways; the only idea which was presented to experts when the questionnaire was created could be outlined as “a mixture of issues from different viewpoints”. The experts were invited to estimate every issue’ importance and problematic level with the use of 1, ... , 10 scale, where 1 stands for “no importance” and “no problems” and 10 - for “highest importance and “largest problem”, while 0 stands for the expert’ uncertainty and was used to signal not to count corresponding estimation. As usual, the expert data processing is a two steps procedure:

1. Calculating average importance and problematic values for all actual responses for each issue;

2. Calculating issue criticality as a product of it’s average importance and problematic; ranking of all issues in accordance with their criticality; the criticality value itself should serve as the issue “weight”.

The resulting ranking and weighting of all issues together with their ordering numbers within the initial questionnaire are presented in Table 1.

Table 1: Quality Issues Ranking

QI Rank	QI No	QI Title	Criticality
1	16	Debugging and Testing	74.26
2	24	Fault Tolerance	67.47
3	5	Skills	57.73
4	15	Modelling	57.12
5	18	Systems Approach	55.04
6	9	Decision Procedure	50.85
7	8	Verification	50.22
8	2	Standards	47.46
9	4	Quality System	46.00
10	14	Design Technology	45.62
11	22	Confidence	45.41
12	1	Specification	42.56
13	19	Structurability	41.50
14	6	Customizability	41.23
15	3	User Involvement	40.95
16	13	Software and Information portability	36.47
17	17	Ergonomics	34.76
18	23	Predictability	32.98
19	21	Motivation	32.64
20	25	Visibility of Resources	29.87
21	7	Design Platform	27.16
22	12	Unification	26.46
23	11	Modules	24.09
24	10	User Technology	23.90
25	20	Exhaustiveness	23.21
26	26	Certainty	19.99

It should be mentioned that most of the experts involved in this research had much more experience as IS designers than IS Users, however dealing with educational systems and being professors themselves they do understand the importance of quality assurance in a sense of meeting user requirements. Nevertheless the maximum criticality "weight" (74,26 out of 100 ever possible) belongs to a purely Designer's domain; normally the User ("end-user") has nothing to do with "Debugging and Testing". Purely User's issue ("Certainty") was almost neglected (only 19,99 as the issue weight). The average criticality according to Table 1 is 47,13 and only 8 issues have more than average criticality.

Let us reason about structuring of the IS quality assurance issues. One possible structure based on the User viewpoint presentation was introduced in section 2.1. However some other simple structures are also useful. The first approach could be a pure division of all issues into two domains corresponding respectively the User Viewpoint and the Designer Viewpoint. It should be mentioned that this taxonomy arises out of the issue's rationalies (Appendix 1). The resulting domains features are summarized in Table 2. It was a big surprise that even deciding in favour of the Designer domain every time when it could be argued (Customizability, Fault Tolerance, Modelling, etc.) we have achieved nevertheless not so much difference between average weights.

Table 2: Two Domains

The User Domain	The Designer Domain
Decision Procedure, Quality System, Confidence, User Involvement, Ergonomics, Predictability, Motivation, Visibility of Resources, User Technology, Certainty	Debugging and Testing, Fault Tolerance, Skills, Modelling, Systems Approach, Verification, Standards, Design Technology, Specification, Structurability, Customizability, Software and Information portability, Design Platform, Unification, Modules, Exhaustiveness
Average issue weight is: 35.74	**Average issue weight is: 44.85**

The next decision on structuring was based on splitting of both the User and the Designer domains and withdrawing all argued issues to a new also splitted domain which was called System domain. Each of three domains was splitted according to a simple rule: if the issue could be considered as business, design, and organization indifferent then it was moved to a "general" subdomain, otherwise - to a "special" subdomain. The resulting structure is presented in Table 3. Resulting subdomains could be ranked according to obtained data as follows: 1. System general domain; 2. System special domain; 3. Designer special domain; 4. User special domain; 5. User general domain; 6. Designer special domain.

This is a picture of current priorities in general, while each IS requires its own understanding of the key quality assurance ranking

Table 3: Three splitted domains

The User General Domain	**The User Special Domain**
Confidence, Predictability, Motivation, Visibility of Resources	User Involvement, Motivation, User Technology, Decision Procedure
Average issue weight is: 36.09	**Average issue weight is: 37.09**

The Designer General Domain	**The Designer Special Domain**
Customizability, Software and Information portability, Exhaustiveness	Debugging and Testing, Fault Tolerance, Skills, Modelling, Systems Approach, Verification, Standards, Design Technology, Specification, Structurability, Design Platform, Unification, Modules
Average issue weight is: 33.64	**Average issue weight is: 45.06**

The System General Domain	**The System Special Domain**
Specification, Standards, Unification, Fault Tolerance, Quality System, Confidence, User Involvement, Predictability, Motivation, Visibility of Resources, User Technology	Ergonomics, Systems Approach, Verification, Modelling, Certainty
Average issue weight is: 45.99	**Average issue weight is: 43.43**

The User Special (US) and the User General (UG) domains have the minimal distance between them measured in absolute difference of their average criticalities. On contrary the Designer General (DG) and Designer Special (DS) domains stand far from one another. To make this distance interpretation more impressive let us consider a three dimensional space with some objects (domains) placed in it. The distances between objects will be consistent to the absolute difference of corresponding average criticalities. Resulting picture (with some distances between domains) is presented in Figure 2 reflecting user preferences for the given structure for the IS quality issues.

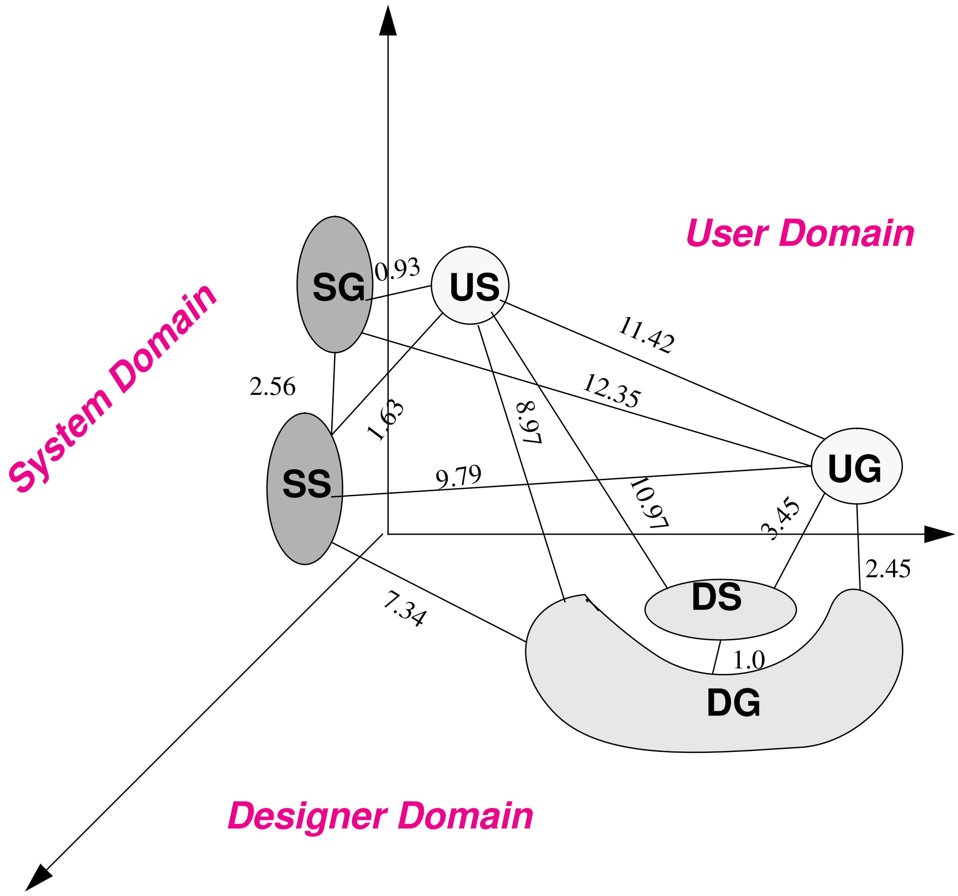

Fig. 2 Graphical presentation of the User preferences for the given structure for the IS quality issues.

5. Further research area.

The IS quality assurance issues are highly relevant to the IS strategy definition which influences organizational dynamics, namely defines the programme for the change management. The role of the IS in this process is very high and not precisely understood to the moment. Introducing different viewpoints could be the way to their harmonization through the alignment of both the User and the Designer viewpoints with the System viewpoint; according to obtained results one might expect to have always highest priorities to general values. Taking into account that contemporary business is more and more based on telecommunications [8] it will have a certain interest to build a system model purely based on concepts of open distributed processing. This will be of special interest for the IS quality assurance because this approach enables precise *business* definitions of future IS at a very high level of abstraction in concern with the IS D&D.

References

[1] Baets W., Smirnov M.: Strategic IS planning: relevance for Russia. In: Whitley E.A., ed. *Proceedings of the 1st European Conference on Information Systems*, (1993), 48 - 64.

[2] Galliers R.D.: *The key IS management issues*. IS/IT lecture courses provided in The Warwick Business School (UK) and The Saint-Petersbrug State Electrotechnical University (Russia),(1993, 1994)

[3] Eden C.: Cognitive Maps as a Visionary Tool: Strategy Embedded in Issue Management. - In: Dyson R.G., ed. *Strategic Planning: Models and Analytical Techniques*. J. Wiley and Sons, (1992), 107 - 118.

[4] Hochstrasser B.: Quality Engineering (QE): A new framework applied to justifying and prioritising IT investments. In: Whitley E.A., ed. *Proceedings of the 1st European Conference on Information Systems*, (1993), 213 - 230.

[5] Lenk K.: New Dimensions of Information Management - From Information technology Management to Knowledge Management. In: Bazewicz M. ed. *Proceedings of the ISAT School/IFIP TC 8/WG 8.5 Workshop*. Wroclaw (1993) 66 - 71.

[6] McKay J., Marshall P., McDermid D.: Cognitive Mapping for Information Requirements Determination in the Learning Organization. In: Baets W.R., ed. *Proceedings of the 2nd European Conference on Information Systems*. Nijenrode (1994) 759 - 770.

[7] Niederman F., Brancheau J.C., Wetherbe J.C.: Information Systems Management Issues in the 1990s. In: *Working Paper Series*, University of Minnesota, US, MISRC-WP-91-08, 37 p.

[8] Segars A.H.,Grover V.:Communications architecture: towards a more robust understanding of information flows and emergent patterns of communication in organizations. *European Journal of Information Systems*, v. 3, No 2,(1994) 84 - 100.

[9] Smirnov M.: Informatization in St.-Petersburg: status report. In: Baets W.R., ed. *Proceedings of the 2nd European Conference on Information Systems*. Nijenrode (1994) 165-174.

[10] Raymond K.A.: Reference model of Open Distributed Processing: a tutorial. In: Meer J. De, Mahr B., Storp S., eds. *Open Distributed Processing, II, Proceedings of the IFIP TC6/WG 6.1 International Conference on Open Distributed Processing*, North-Holland (1994) 3-14.

[11] Remenyi D., Money A.: Service Quality and Corresponding Analysis in Determining Problems with the Effective Use of Computer Services. In: Whitley E.A., ed. *Proceedings of the 1st European Conference on Information Systems*, (1993), 320-339.

[12] Watson R.T., Brancheau J.C.: Key Issues in information systems management. An international perspective. *Information and Management* 20 (1991) 213 - 223.
[13] Watson R.T., Pitt L.F., Cunningham C.J., Nel D.: User Satisfaction and service Quality of the IS Department: Closing the Gaps.In: Whitley E.A., ed. *Proceedings of the 1st European Conference on Information Systems*, (1993), 220-239.
[14] Wheeler F.P.: Executive IS as repositories of organizational understanding. In: Baets W.R., ed. *Proceedings of the 2nd European Conference on Information Systems*. Nijenrode (1994) 67-74.
[15] Wolfe M.: Development of the city of quality: A hypertext-based group decision support system for quality function deployment. *Decision Support Systems*, vol. 11, No 3 (1994).299 - 318.

Appendix 1:

Quality Issues

(The sample questionnaire for the IS Quality Key Issues Estimation)

1. Specification

Rationale: *The existence of precise and detailed specifications for the IS under development is a key issue both for the quality of the IS and for the quality of all development, use and maintenance processes.*

2. Standards

Rationale: *Standardizing all phases of the IS lifecycle will lead to a constructive quality which couldn't be less than standard value; this standard quality is provided by standard design decisions and IS elements unification such as user interfaces.*

3. User Involvement

Rationale: *Only total user involvement in the design process could help in IS development which will obtain a real quality estimated by a real expert. On contrary "turn key" projects - maybe fairly well designed - lack the real knowledge of a real business which guarantees the user satisfaction.*

4. Quality System

Rationale: *No one single issue taken separately could be treated as a key to the IS quality. The realistic way of high quality IS is a total quality management system supporting all phases of the lifecycle making the quality controllable.*

5. Skills

Rationale: *Only high qualification and experience of the IS designers can help in high quality IS development*

6. Customizability

Rationale: *The IS quality is a dynamic concept; it depends on the user and adopted viewpoint; hence the main support should be made for attempts to achieve as much Customizability in the IS as possible; the end-user should customize the pilot version himself (a reliable way to involve him in the process)*

7. Design Platform

Rationale: *Modern programming languages and software development platforms support the high quality of the designed IS via high level of automation and software models reusability.*

8. Verification

Rationale: *Decision on quality is closely connected with the verification of the ready to use IS employing the user adopted specifications of the IS functionality.*

9. Decision Procedure

Rationale: *Any IS is designed and developed to solve some set of tasks with lots of decision making on a business processes; the right way to estimate IS quality is to prove that all decision procedures embedded in the IS are correct and optimized.*

10. User Technology

Rationale: *When the system is installed it is of key importance to arrange for proper use of the IS; in this sense user education plays the main role.*

11. Modules

Rationale: *When the IS is divided into well defined and specified modules which fit well the business purposes then the quality of the IS could be fairly achieved and measured.*

12. Unification

Rationale: *The user viewpoint on IS quality is highly related to the unification of the IS elements such as user interfaces; decision support functions and technology connections to the business processes.*

13. Software and Information portability

Rationale: *The IS quality highly relates on business knowledge embedded in software and information pool; if software and information portability are achieved the IS evolution is enabled.*

14. Design Technology

Rationale: *The IS quality is a multifaceted issue and it is achieved with a systematic application of proper design technologies during the whole development life-cycle with the use of modern CASE tools, enabling further IS development and maintenance*

15. Modelling

Rationale: *To gain the IS quality (at the D&D phase) it is necessary to imagine a model of a final system; this model has to reflect both structure and functions as well as environment connections (signals, requirements, etc.) and, more important, to use this model for planning of quality activities and for evaluating of current and future quality levels.*

16. Debugging and Testing

Rationale: *Nice software and modern hardware integrated in a system (even well specified with thorough business orientation) can not achieve real quality without close user - developer interaction for the user-reasonable configuration of the IS parts and for a very accurate tuning of some even unknown earlier parameters. This configuration and tuning could be achieved only as a result of the on-site debugging and testing.*

17. Ergonomics

Rationale: *One may hardly believe what amount of money loss was saved by ergonomists when involved in a process of design of any man-machine system; it could be reasonable to apply for the IS quality achievement to ergonomic methods of system analysis and synthesis.*

18. System Approach

Rationale: *When applying a system approach to the IS design and development from the very early stages - definition of targets, tasks, activities, constraints and users - we are very near to a high quality IS development.*

19. Structurability

Rationale: *Structurability of the IS is a very important quality measure, facilitating step by step quality insurance of the IS components.*

20. Exhaustiveness

Rationale: *The IS quality as a measure for user requirements fulfilment could be measured by the needed functional exhaustiveness which supports earlier unknown (unspecified) user requirements and helps in improvement of user confidence in his own skills and IS resources.*

21. Motivation

Rationale: *The high quality IS should motivate the user to do his job in a creative manner, emphasizing elimination of absolutely all routine activities, however underlining the key role of the user in a data flow control and events management.*

22. Confidence

Rationale: *The user confidence in IS, as one of the key quality issues, should be provided by all means assuring results correctness, cost effective operation, predictability and full operation control with thorough analysis of all possible decisions.*

23. Predictability

Rationale*: Most of the users will claim predictability of their IS as one of the key quality issues; will refer to it as an integrated estimation of the quality of cooperation between the IS user interface and operational components.*

24. Fault Tolerance

Rationale: *Any quality feature is of less importance without fault tolerance assurance installed in a solid and clear manner in all strategic parts of the IS, making impossible any loss of advantages gained with the use of the IS.*

25. Visibility of Resources

Rationale: *One of the main IS quality features is it's ability to enable full vision of existing resources for a decision making person; if the IS takes part in some business unit valid for a decision making then all relevant resources of the unit should be made visible.*

26. Certainty

Rationale: *The IS should provide a certain way for any possible scenario choice.*