

Evaluation of a Dependability Mechanism for Cyber Physical Systems

Gilbert Regan¹, Fergal Mc Caffery¹, Jan Reich², Eric Armengaud³, Cem Kaypmaz³, Joe Zhensheng Guo⁴, Simone Longo⁵, and Eoin O Carroll⁶.

¹ Lero @DKIT, Dundalk, Ireland

Gilbert.regan and fergal.mccaffery@dkit.ie

² Fraunhofer IESE, Kaiserslautern, Germany

jan.reich@iese.fraunhofer.de

³ AVL List GmbH, Austria, Turkey

eric.armengaud and cem.kaypmaz@avl.com

⁴ Siemens, Munich, Germany

joe.guo@siemens.com

⁵ General Motors, Turin, Italy

simone.longo@gm.com

⁶ Portable Medical Technology, Kerry, Ireland

eoin@portablemedicaltechnology.com

Abstract. Cyber-Physical-Systems (CPS) are systems of collaborating computational entities. Concepts such as autonomous cars, smart electric grid, implanted medical devices and smart manufacturing are some practical examples of CPS. However, the open and cooperative nature of CPS poses a significant new challenge in assuring dependability. The DEIS project addresses this important and unsolved challenge by developing technologies that facilitate the efficient synthesis of components and systems based on their dependability information. The key innovation that is the aim of DEIS is the corresponding concept of a **Digital Dependability Identity** (DDI). A DDI contains all the information that uniquely describes the dependability characteristics of a CPS or CPS component. DDIs are synthesised at development time and are the basis for the (semi-)automated integration of components into systems during development, as well as for the fully automated dynamic integration of systems into systems of systems in the field.

In this paper we present an overview of the DDI. Additionally, we provide metrics for evaluating the DDI's impact on CPS dependability, and the results of an evaluation of the DDI's impact on dependability in four CPS industrial use cases. These results demonstrate the positive impact of the DDI on the dependability of CPS.

Keywords: dependability, cyber physical system, evaluation, cyber security.

1 Introduction

Cyber-Physical Systems (CPS) harbor the potential for vast economic and societal impact in domains such as mobility, home automation and delivery of health. At the same time, if such systems fail they may harm people and lead to

temporary collapse of important infrastructures with catastrophic results for industry and society. There are two core challenges while assessing the dependability of a CPS. First, the inherent complexity of modern CPS and the resulting complex market organisation requiring the tight cooperation between different teams, expertise, and institutions, while managing confidentiality issues. The second challenge is related to the increase of connectivity, e.g., through machine to machine cooperation enabled by Internet of Things, which introduces a new dynamic in system operation. As a result, Cyber-Physical Systems of Systems (CPSoS) come together as temporary configurations of CPS, and which dissolve and give place to other configurations. This leads to a potentially infinite number of variants, with cooperation between systems potentially not analysed during design time.

The DEIS project¹ addresses these important and unsolved challenges by developing technologies that form a science of dependable system integration. In the core of these technologies lies the concept of a Digital Dependability Identity (DDI²) of a component or system. The DDI targets (1) improving the efficiency of generating consistent dependability argumentation over the supply chain during design time, and (2) laying the foundation for runtime certification of ad-hoc networks of embedded-systems. During the DEIS project, four industrial use cases are provided to evaluate the performances of the DDI. The target is to evaluate the impact of the proposed methodology for process improvement during product development, and to support the emergence of new functions, e.g., supported by higher degree of collaboration. The core challenge for the evaluation of the project relies on the two levels of innovation: first the dependability engineering approach shall be enhanced, second this shall be deployed to improve the industrial product with new solutions.

Contribution of this paper is to present a systematic approach for the evaluation of dependability methodologies for CPS, and to apply this method for the evaluation of the DDIs in the four industrial use cases of the DEIS project. The paper is organized as follow: Section 2 presents related works on quality assessment. An overview of the DDI is presented in Section 3, and the research methodology as well as the use cases are introduced in Section 4. In Section 5, the tailoring of the standards used for evaluation are presented, and in Section 6 the evaluation results are provided. Finally, Section 7 concludes this work.

¹ Dependability Engineering Innovation for automotive CPS. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 732242, see www.deis-project.eu

² Schneider, D., Trapp, M., Papadopoulos, Y., Armengaud, E., Zeller, M., & Höfig, K. (2015). WAP: Digital dependability identities. *26th International Symposium on Software Reliability Engineering (ISSRE'15)*, (pp. 324-329)

2 Related Work

A software quality model can be defined as ‘a model that describes, assesses and/or predicts quality’ [1], or as ‘a set of factors, criteria and metrics (characteristics) and the relationship between them. These relations provide the basis for specifying quality requirements and evaluating quality’ [2].

Several models of software quality factors and their categorisation have been suggested over the years. The first software quality models were published in the mid 1970’s by Boehm et al. [3] and Mc Call et al. [4]. Mc Call identified three main perspectives for characterising the quality attributes of a software product i.e. a product’s ability to change, adaptability to new environments, and basic operational characteristics. From these three perspectives Mc Call identified eleven characteristics. The major contribution of the McCall method was to consider relationships between quality characteristics and metrics. This model was used as a base for the creation of others quality models [5]. The main drawback of the Mc Call model is the accuracy in the measurement of quality, as it is based on responses of Yes or No. Furthermore, the model does not consider the functionality so that the user’s vision is diminished [6]. Boehm’s model [3] constitutes an improvement on Mc Call’s model because it is based on a wider range of characteristics and because it adds factors at different levels.

The FURPS quality model [7], which was proposed by Robert Grady from Hewlett Packard in 1992, takes into account the following five characteristics: Functionality, Usability, Reliability, Performance, and Supportability. A main drawback of this model is that it does not consider some important characteristics such as portability, which may be an important criterion for application development [8]. In 1995 Robert Dromey proposed a product based quality model [9] based on the idea that a more dynamic way of modelling process was needed. This was due to the fact that quality evaluation differs between products and the model needed to be wide enough to apply to different systems.

In order to standardise quality assessment, the International Organisation for Standardisation developed ISO 9126 [10] in 1991 and revised it in 2001 [11]. This standard is an extension of previous models as defined above, and is divided into four parts which address the following subjects: quality model; external metrics; internal metrics; and quality in use metrics. The quality model is divided into the following six characteristics: Functionality; Reliability; Usability; Efficiency; Maintainability; and Portability. The internal metrics are static metrics that do not rely on software execution, whereas the external metrics rely on running software. Quality in use metrics can be measured only when the final product is used in a real environment with real conditions.

The ISO 9126 model was updated in 2005 and evolved to become part of the ISO 25000:2005 [12] series, and which has further been revised with ISO 25000:2011 [13]. Studies conducted by [14-16] indicate that the ISO/IEC 25010 [17] model is the most comprehensive quality model available because it covers the most quality characteristics and sub-characteristics. It achieves this by adding new characteristics such as security and compatibility.

3 Overview of DDI

Assurance cases represent the backbone of modern dependability assurance processes, because they record the dependability requirements to be fulfilled by a system (of system) in an intended operational environment together with the evidences that support the requirement's validity in the finally implemented system. All produced dependability-engineering artifacts using such evidence are motivated by an uncertainty about whether a dependability claim about the system is actually fulfilled.

Since there is an interrelation between the system, its dependability claims, and the supporting evidence artifacts that exist in the real world, we claim this should also be the case for the system's model-based safety reflection, i.e. its DDI (see Fig. 1). DDIs represent an integrated set of dependability data models that may be (semi-)automatically analysed, generated or manipulated during the execution of safety engineering processes.

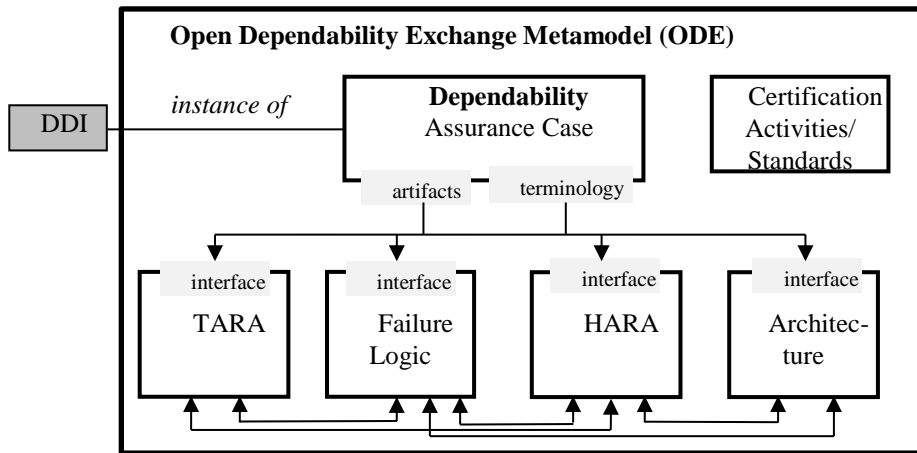


Fig. 1. The Open Dependability Exchange Metamodel (ODE)

A DDI contains information that uniquely describes all dependability characteristics of a system required for certifying the system's dependability. DDIs are formed as modular assurance cases and their composability allows for the (semi-)automatically synthesizing of system DDIs from the DDIs of the subcomponents. The DDI of a system contains a) claims about the dependability guarantees given by a system to other systems and derived system dependability requirements and b) supporting evidence for those claims in the form of various models and analyses. For security assurance, it contains e.g. threat and risk analyses (TARA) and attack trees, while for safety assurance, hazard and risk analyses (HARA), architecture modeling and failure propagation modeling such as fault trees, FMEA or Markov chains are supported.

Due to the integration and standardisation of these models in the Open Dependability Exchange Metamodel (ODE), a self-contained system dependability package can support many dependability-engineering activities of the system lifecycle.

4 Research Methodology

Section 4.1 provides an overview of the methodology employed to select the quality metrics used to evaluate the impact of the DDI, while section 4.2 provides a brief description of the four use cases used for this same evaluation.

4.1 Methodology

The methodology used to conduct this research comprised the following main stages: Select metrics; Select use cases; Evaluate DDI impact in use cases; and Report findings.

As stated in section 2, the ISO 25010 quality model is the most comprehensive quality model available and so this model, and specifically metrics from the following standards within the ISO 25000 series were selected to assess the impact of the DDI: ISO 25022 [18]; ISO 25023 [19]; and ISO 25024 [20]. Details of the metrics and their selection are provided in section 5.

Four industrial partners on the DEIS project each put forward a Use case for assessing the impact of the DDI. Two use cases are embedded in the automotive domain while one each are embedded in the railway and healthcare domains. A short description of these use cases is provided in section 4.2. For each use case a team of people from within each use case's organisation conducted the evaluation. Each use case was evaluated both before and after application of the DDI. The make-up of the teams was decided upon by the organisation themselves, so for example the Siemens team included 1 model-based safety and reliability engineer, 1 model-based safety and reliability consultant, 1 safety engineer, and 1 reliability engineer. Through expert judgement and consensus, and with the use of the measurement formulae within the standards, each team determined values for the selected characteristics. An example of how one organisation arrived at a value for the 'functional suitability' quality characteristic is provided in Table 2 in section 5. The results for each organisation's quality characteristic assessment are provided in section 6.

4.2 Use cases

Portable Medical Technology (PMT): ONCOassist, PMT's mobile decision support application for oncology professionals will evaluate DDI in the integration between ONCOassist and hospital systems electronic health records. The use case will show the security and data validity as it transfers patient data between systems. Oncology professionals will pass patients data to ONCOassist to preform calculations that would not be possible on the hospital system. The integration will save time and increase accuracy significantly.

General Motors (GM): The Dependable Physiological Monitor System (DPMS) use case describes the sensing environment inside the vehicle that monitors the health condition of drivers and passengers. The DPMS aids prevention of accidents in cases where the occupant's health condition deteriorates. The DPMS will apply the DDI at both development time and at runtime. During the development phase a reduction of time-to-market is expected as a consequence of the usage of DDI methodology. At runtime, DDI is evaluated against the overall dependability of this system, dealing with security and privacy aspects in V2V and V2C communication.

Siemens (SAG): The Siemens railway use case is to evaluate DDI on a European Train Control System (ETCS) and its development. Currently this use case focuses on the on-board unit instead of on the track-side unit. Concretely this use case will be used for proving and illustrating strengths and drawbacks of DDI in several engineering phases, such as architecture, qualitative and quantitative dependability analysis, and Goal Structural Notation (GSN) based dependability assurance case development.

AVL: Heavy-duty trucks create a platoon to reduce time gaps between the trucks, to increase energy efficiency, improve safety, and to reduce truck driver loads. In the SAE L4 platoon function, platoon level decisions are taken by the platoon leader and broadcasted to the follower vehicles and executed by each member without any need of a human driver or operator in a constrained operation boundary. Two-way information flow between vehicles and different communication topologies between members creates a wide range of dependability.

5 Metric Selection

The quality characteristics, and the metrics for these characteristics, were selected from the following standards: ISO 25022; ISO 25023; and ISO 25024. ISO 25022 defines characteristics and measures for evaluating quality in use characteristics i.e. quality from the end user's perspective, ISO 25023 defines characteristics and measures for quantitatively evaluating system and software product quality characteristics, while ISO 25024 defines data quality characteristics and measures for quantitatively evaluating the quality of the data within the system.

These standards contained more quality characteristics than was relevant for measuring the impact of the DDI, therefore the first task was to decide which characteristics were relevant. A focus group, containing members from the DEIS project partners, was formed to conduct this task. The members of this focus group are listed as authors of this paper. The process of selecting the relevant metrics included five 1.5 to 2 hour on-line meetings. At these meetings, each metric within the three standards was discussed in detail with two main considerations in mind: 1) its relevance to assessing the impact of the DDI; and 2) practicality for each partner to make the measurement. Decisions on whether to include a metric, or not, were based on a general consensus which was largely unanimous in each case. The number of characteristics and measures within each standard, along with the number of characteristics and measures selected from each of the standards, is displayed in Table 1.

Table 1. Number of characteristics and measures within standards versus number of characteristics and measures selected for measuring DDI impact

Number of characteristics and measures within standards			Selected for measuring impact of DDI	
Standard	Characteristics	Measures	Characteristics	Measures
ISO 25022	5	36	4	10
ISO 25023	8	86	7	29
ISO 25024	15	63	10	19
	28	185	21	58

In total, the team selected 21 quality characteristics (from 28 within the standards). From these 21 characteristics, 42 sub-characteristics were selected as shown in section 6 below. A total of 58 separate measures (from 185 within the standards) were employed in order to determine values for the quality characteristics. An example of how a value for the 'Functional suitability' quality characteristic could be arrived at is shown in Table 2.

Table 2. Example determination of 'Functional suitability' characteristic

Characteristic	Measure	Measurement function	Result for each measure	Result for characteristic
Functional suitability measures	Functional coverage	$X = 1 - A/B$ A = Number of functions missing B = Number of functions specified	A=2 B=60 X=0.967	$(0.967+0.983)/2 = 0.975$
	Functional correctness	$X = 1 - A/B$ A = Number of functions that are incorrect B = Number of functions considered	A=1 B=60 X=0.983	

6 Evaluation Results

The results from evaluating the DDI in the four industrial use cases are now presented in the following three subsections.

6.1 ISO 25022 Quality in Use Results

Table 3 presents the results from evaluating the four 'Quality in Use' characteristics which have been selected from ISO 25022. The selected sub-characteristics (three

in total) are listed in column 2. The ‘Effectiveness’ and ‘Efficiency’ characteristics have no sub-characteristics. The results for each use case is presented in 2 columns with the first column presenting the result without the DDI applied, and the second column (in italics) presenting the result with the DDI applied. Each individual result can vary from 0 to 1, with 1 being equivalent in percentage terms to 100. The last column in the table presents the average improvement, in percentage terms, across the four use cases, so for example ‘Effectiveness’ increased by an average of 14.2% when DDI was applied.

Table 3. ‘Quality in Use’ characteristic values across four use cases

Characteristic (4)	Sub characteristic (3)	GM	<i>GM</i>	AVL	<i>AVL</i>	PMT	<i>PMT</i>	SAG	<i>SAG</i>	AVG % imp.
Effectiveness	n/a	0.39	<i>0.56</i>	0.5	<i>0.63</i>	0.5	<i>0.58</i>	0.64	<i>0.83</i>	14.2
Efficiency	n/a	0.44	<i>0.62</i>	0.25	<i>0.44</i>	0.53	<i>0.82</i>	0.05	<i>0.95</i>	39.0
Freedom from Risk	Economic risk mitigation	0.61	<i>0.73</i>	0.65	<i>0.85</i>	0.33	<i>0.68</i>	0.73	<i>0.87</i>	20.3
Context Coverage	Context completeness, and Flexibility measures	0.2	<i>0.2</i>	0.42	<i>0.67</i>	0.5	<i>1.0</i>	0.35	<i>0.69</i>	27.3

The last column above indicates that the application of the DDI resulted in significant improvement in each of the ‘Quality in Use’ metrics. The ‘Efficiency’ metric, at 39% improvement, is particularly influenced by the SAG results who state that ‘*We are expecting a significant increase of the number of the objectives achieved for the same period of time by introducing DDI. Furthermore, we are expecting a significant decrease in the cost for carrying out the task for the same amount of objects in ETCS use case*’.

Another interesting observation from Table 3 is the GM result for ‘Context Coverage’. Context coverage assesses the degree to which a product or system can be used with effectiveness, efficiency, satisfaction, and freedom from risk in both specified contexts of use and in contexts beyond those initially explicitly identified. GM results indicate no improvement as ‘*no other scenario has been evaluated for DPMS usage*’.

6.2 ISO 25023 System and Software Quality Results

Table 4 presents the results from evaluating the seven ‘System and Software Quality’ characteristics which have been selected from ISO 25023, and is structured the same as Table 3. The seven characteristics contain twenty sub-characteristics.

While all characteristics indicate an average improvement across the four use cases, ‘Performance efficiency’ is the lowest at 4.5%. This is due to most of the use cases reporting a very small improvement in this characteristic, with Siemens reporting practically no improvement due to such factors as: ‘*Resource utilization: for mean processor utilisation and bandwidth utilisation, we could not observe any improve-*

ment by use of DDI. Both mean processor utilisation and bandwidth utilisation remain low for railway safety-critical system'

Table 4. System and Software characteristics across four use cases

Characteristic	Sub characteristics	GM	GM	AVL	AVL	PMT	PMT	SAG	SAG	AVG % imp
Functional suitability	Functional completeness, and correctness	0.65	0.7	0.55	0.65	0.67	0.83	0.99	0.99	7.8
Performance efficiency	Time behavior, and Resource utilisation	0.46	0.51	0.39	0.47	0.44	0.48	0.39	0.4	4.5
Compatability	Co-existence, and Interoperability	0.5	0.5	0.71	0.83	0.43	0.73	0.79	0.83	11.5
Reliability	Maturity, Availability, and Fault tolerance	0.71	0.77	0.66	0.78	0.66	0.70	0.75	0.63	7.0
Security	Confidentiality, Integrity, Authenticity, Accountability, and Non-repudiation	0.6	0.68	0	0.2	0.2	0.38	0	0.2	16.5
Maintainability	Reusability, Analysability, Modifiability, Testability	0.45	0.52	0.39	0.6	0.44	0.78	0.51	0.55	16.5
Portability	Adaptability, Replaceability	0.3	0.3	0	0	0.33	0.83	0	0	12.5

The 'Portability' characteristic, which assesses the degree of effectiveness and efficiency with which a system, product or component can be transferred from one hardware, software or other operational or usage environment to another, indicated no improvement across three use cases (GM, AVL, and SAG). This was mainly due to factors such as no portability is conducted in the use case: GM stated '*No improvement here, considering that no other scenario has been evaluated outside the GM architecture ecosystem*', while AVL stated '*No portability related implementation has been done*'. For SAG the reason was somewhat different in that they reported that '*the DDI does not offer additional data or handling in case of porting the ETCS system onto another environment*'.

For the 'Security' characteristic, which assess the degree to which a system protects data so that persons or other systems have the degree of data access appropriate to their types and levels of authorization, two use cases reported a score of zero for the security sub-characteristics listed in Table 4, however with application of the DDI their security score improved to 0.2. Both of the use cases reported an improvement due to implementation of authentication rules.

For 'Functional suitability' SAG reported no improvement due to the fact that they consider their system to be practically functionally complete and correct. They stated that their ETCS products *have 1% of missing intended usage of the system without*

DDI (99% of usage completeness).....This estimation is also true for the correctness of functions’.

While the majority of the selected system and software quality metrics are applicable to most of the use cases, according to the industry partners there are occasions where some metrics may not apply to some use cases. For example the ‘portability’ metric only showed improvement in the PMT use case.

6.3 ISO 25024 Data Quality Results

Table 5 presents the results from evaluating the ten ‘Data Quality’ characteristics which have been selected from ISO 25024, and is structured the same as Table 3. The ten characteristics contain nineteen sub-characteristics.

Table 5. Data quality characteristics across four use cases

Characteristic	Sub characteristics	GM	GM	AVL	AVL	PMT	PMT	SAG	SAG	AVG % imp
Data accuracy	Syntactic accuracy	0.6	0.68	0.5	0.63	0.5	0.63	0.88	1	11.5
Completeness	Record, and Attribute completeness	0.55	0.61	0.38	0.63	0.25	0.88	1	1	23.5
Consistency	Data format and Architecture consistency, and Risk of data inconsistency	0.38	0.48	0.33	0.58	0.33	0.53	0.61	0.66	15.0
Credibility	Values, Source, and Data model credibility	0.6	0.64	0.5	0.67	0.47	0.87	1	1	15.3
Compliance	Regulatory compliance of value, and Technology	0.54	0.65	0.44	0.75	0.3	0.8	1	1	23.0
Confidentiality	Encryption, and Non-vulnerability	0.66	0.66	0.38	0.63	0.55	0.71	0	0	10.3
Precision	Data values, and Format precision	0.48	0.51	0.38	0.63	0.6	0.8	1	1	12.0
Traceability	Traceability of data values, Data value Traceability	0.5	0.64	0.38	0.5	0.19	0.75	0.7	0.95	26.8
Understandability	Semantic understandability	0.55	0.68	0.25	0.63	0.6	0.8	0	0	17.8
Portability	Data portability ratio	0.48	0.63	0.38	0.63	0.4	0.8	0	0	20.0

Table 5 indicates that all the selected data quality characteristics show an average improvement across the 4 use cases, ranging from 10.3% to 26.8%. However for one use case, seven of the characteristics show no improvement. For the data complete-

ness, data credibility, data precision and data compliance, SAG state that their ETCS system has to be certified according to relevant safety standards and that these values are at 100% regardless of whether the DDI is applied or not. For data confidentiality, SAG state that this metric is not applicable but give no reason for this. With regards to data understandability SAG report no improvement, stating that ‘the *semantic understandability will not be changed by introducing DDI*’, and for data portability SAG state that ‘the *DDI does not influence the portability in the sense of operation environment adaptability and data reusability/import capability*’. The fact that three of the metrics have scored zero with and without DDI implementation clearly indicates that SAG do not think that these metrics can be improved in their use case.

For GM, the data confidentiality metric indicated no improvement. The reason for this according to GM is that applying the DDI guarantees the same level of data confidentiality as against not applying the DDI. However GM further state that the DDI can help in selecting at design time the best security solution to satisfy confidentiality requirements.

While the majority of the selected data quality metrics are applicable to most of the use cases, there were occasions where some metrics may not apply to some use cases.

7 Conclusion

The selected metrics for measuring the impact of the DDI were chosen because of their relevance to assessing the impact of the DDI, and their practicality for each partner to make the measurement. The results of the evaluation indicate that applying the DDI has made significant improvements in the quality of each use case, from an end user and from a system and data perspective. The results show that in all cases applying the DDI has made an average improvement for all metrics, with the smallest improvement being 4.5% for Software and System Performance efficiency, and the largest improvement being 39% for the quality in use ‘Efficiency’ metric. These results demonstrate the positive impact of the DDI on the dependability of CPS.

However, the results of the evaluation also indicate that not all metrics may apply to all use cases, and that not all metrics showed an improvement in all use cases, for example the Systems and Software Quality metric ‘Portability’ only showed improvement in one of the four use cases. While all metrics were applied, the industry partners indicated that in some instances a relatively small number of the metrics did not apply to their use case.

Acknowledgement This paper is supported by the European Union’s Horizon 2020 research and innovation programme under grant agreement No 732242. It is also supported in part by Science Foundation Ireland grant 13/RC/2094.

References

1. Wagner, S., *Software product quality control*. 1 ed. 2013, Berlin: Springer-Verlag Berlin Heidelberg. 210.

2. Lincke, R. and W. Lowe, *Validation of a Standard and Metric Based Software Quality Model : Creating the Prerequisites for Experimentation*. 2007, The KK foundation: Sweden.
3. Boehm, B., J. Brown, and M. Lipow. *Quantitative Evaluation of Software Quality*. in *ICSE '76 Proceedings of the 2nd international conference on Software engineering*. 1976. San Francisco, California, USA: IEEE Computer Society Press.
4. McCall, J.A., P.K. Richards, and G.F. Walters, *Factors in software quality*. 1977, Griffiths Air Force Base, N.Y. : Rome Air Development Center Air Force Systems
5. Dubey, S., S. Ghosh, and A. Rana, *Comparison of Software Quality Models: An Analytical Approach*. International Journal of Emerging Technology and Advanced Engineering, 2012. **2**(2): p. 111-119.
6. Miguel, J., D. Mauricio, and G. Rodriguez, *A Review of Software Quality Models for the Evaluation of Software Products*. International Journal of Software Engineering & Applications (IJSEA), 2014. **5**(6).
7. Grady, R., *Practical software metrics for project management and process improvement*. 1992: Prentice-Hall.
8. Singh, I., *Different Software Quality Models*. International Journal on Recent and Innovation Trends in Computing and Communication, 2013. **1**(5): p. 438-442.
9. Dromey, R., *A Model for Software Product Quality*. IEEE Transactions on Software Engineering, 1995. **21**(2): p. 146-162.
10. ISO, *ISO 9126 Software Engineering - Product Quality* 1991.
11. ISO, *ISO 2196-1 Software engineering - Product Quality - Part 1: Quality Model*. 2001.
12. ISO, *Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE*. 2005.
13. ISO, *Sytem and software engineering - System and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models*. 2011.
14. Samarthyam, G., et al., *MIDAS: A Design Quality Assessment Method for Industrial Software, ICSE 2013 : Software Engineering in Practice*: San Francisco. p. 911-920.
15. Al-Badareen, A., *Software Quality Evaluation: User's View*. International Journal of Applied Mathematics and Informatics, 2011. **5**(3): p. 200-207.
16. Dubey, S., G. Soumi, and R. Ajay, *Comparision of Software Quality Models: An Analytical Approach*. International Journal of Emerging Technology and Advanced Engineering, 2012. **2**(2): p. 111-119.
17. ISO, *ISO/IEC 25010: 2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*. 2011, ISO: Geneva, Switzerland.
18. ISO, *ISO/IEC 25022:2016 Systems and software engineering -- Systems and software quality requirements and evaluation (SQuaRE)- Measurement of quality in use*. 2016.
19. ISO, *ISO/IEC 25023:2016 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Measurement of system and software product quality*. 2016.
20. ISO, *ISO/IEC 25024:2015 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- Measurement of data quality*. 2015.

