High level design of architecture for software reliability management of Power Supply Company Jugoistok

Aleksandar Dimov[*], Nikola Davidović[**], Leonid Stoimenov[**]

[*] Faculty of Mathematics and Informatics, University of Sofia, Bulgaria
aldi@fmi.uni-sofia.bg
[**] Faculty of Electronic Engineering, University of Niš, Niš, Serbia
nikola.davidovic@elfak.ni.ac.rs, leonid.stoimenov@elfak.ni.ac.rs

Abstract—**Power supply companies require IT support and infrastructure at heterogeneous levels that vary from supervisory control to low level management of field devices. On the other side quality of software for power supply systems is important characteristic to consider, but it is difficult to be measured and reason about in formal way. This is mostly due to the fact, that for such systems testing is not always applicable into real-world environment. A big power supply company in Serbia is Jugoistok. In order to overcome the aforesaid problem there, the paper presents architecture of a platform that will collect data to be used for calculation of software quality of the information systems in Jugoistok. To overcome the problem with testing, data will be gathered in various ways – by simulation, user feedback and expert opinion.**

I. INTRODUCTION

Power supply companies are large companies responsible for managing energy usage for wide areas. That implies a large number of households that need to be served, a significant grid area, as well as highly utilized information systems (ISs) for internal business procedures, such as Customer Information Systems (CIS), Document Management System (DocMS) and Geo-Information System (GIS). Additionally, at a single power supply company level, enterprise IS are typically interconnected with various field devices, controls and metering devices within a utility-wide network. In order to enable effective monitoring and management of power supply networks according to the parameters collected (very often in real time), electric power supply companies utilize various specialized information systems, such as Supervisory Control and Data Acquisition System (SCADA), Distribution Management System (DMS) and Automatic Meter Reading (AMR), Technical Information System (TIS) [1].

An inherent property of information systems for power supply companies is about their high requirements for system quality. This includes not only hardware for such systems, but also the software and all information infrastructures (third party software, operating systems, etc.). Quality characteristics may have different definitions depending on the domain. For example, in terms of Service Oriented Architecture, they are referred as Quality of Service, which covers a wide range of techniques that match the needs of service requestors with those of the service provider's based on the network resources available [2]. In other domains quality requirements are called non-functional requirements and should be distinguished from functional requirements. The latter define what the system should do and the former put some additional conditions (in form of constraints or specifications) on how the system should perform or deliver its functionality. There are a lot of examples for quality characteristics, but the most popular are performance, reliability, usability, etc.

Monitoring and management of software quality is important for all kinds of information systems. However, at current time software quality lacks enough structured and formal support for measurement, monitoring and management [3]. The natural way for collecting data in order to determine quality of software systems is to rely on data, gathered during system testing. Nevertheless, in case of information systems for electrical supply, many of the components there are impossible to be tested in real environment, which requires other methods to be applied for such high demanding systems.

In this paper we propose a work in progress architecture for monitoring and control of software systems for power supply and electricity distribution company Jugoistok Niš (from now on Jugoistok) for one of the very important quality characteristics for such systems – namely reliability. Reliability is considered to be part of the broader notion, named dependability, which in terms of software is defined as *the ability of a computing system to deliver services that can justifiably be trusted* [2]. Besides reliability, dependability is also characterized by some other attributes, such as availability, integrity, safety, confidentiality and maintainability. The architecture we propose is service oriented, which makes it highly reusable and also applicable for other quality characteristics, not only reliability, given that one provides services, which implement the appropriate functionality.

The rest of the paper is organized as follows: Section 2 gives more information about reliability and its measurement; Section 3 makes a brief presentation of Jugoistok as a power supply company and its information system; Section 4 presents the architecture of quality management platform and finally section 5 concludes the paper and states some directions for further research.

II. SOFTWARE RELIABILITY

Generally, software reliability represents the belief we have for a system that it will not crash over a specified

period of time, given that it is operating properly at the beginning of this interval [2]. It is a statistical value and may be represented by one of the following measures:

- Probability of failure
- Failure rate
- Mean time to failure

In software engineering there exist a number of models that are generally divided in two big groups that assess software reliability. These groups are named black-box and white-box reliability models. The group of white-box models consists of several kinds of models that are used to estimate the reliability of software systems, based on the knowledge of their internal structure and processes going on inside them. On the other hand, the group of black-box models encompasses much larger number of methods that treat the software as a monolithic whole, i.e. as a black-box.

White box models are also called Architecture-Based Reliability Models (ABRMs). Usually architecture-based software reliability estimation takes the following main steps [6]:

1) Identification of computational modules (components) within software architecture;

2) Description of the actual architectural model – this includes how components are interconnected and interact with each other;

3) Definition of components failure behaviour – at this step the reliability parameters of components and their measures are identified;

4) Combination of the failure behaviour with the architectural model.

Application of white box models has a lot of advantages, among them are: ability to reuse information about reliability parameters of both the system and the components that constitute it; ability to find these modules that influence systems reliability the most, i.e; possibility to isolate and remove reliability "bottlenecks" within the system and etc. For these reasons we focus our research work on white box models.

On the other hand, black-box models take as an input some preliminary data and make statistical processing over it [4]. These models are sometimes also called reliability growth models. Reliability growth assume extensive testing of the software system and observation of failures and the time that have passed between two subsequent failures. Such data may have different representations according to the model and may be obtained by different means. When a failure is detected, the fault that caused it is removed and the process continues with the assumption that correction of the fault did not introduce new errors into the code. However, this is quite unfeasible assumption as real-world practice shows that bug fixing always introduce additional problems with the entire system and that is why regression testing is being run. As already stated in the introduction, testing all parts of the electrical supply information systems, only by testing is not always possible. Consequently, it is necessary to use other methods, not only testing for collection of input data for reliability models. Some of the other popular methods for collection of such data are: software testing, simulation, users feedback and experts opinion [5].

Simulation takes into account that it does not depend only on the structure of the software but also on the runtime information such as frequency of component reuse, execution time spent interactions between the components, etc. Users' feedback is a technique to get information about software reliability parameters of a system, by gathering data, after it has been shipped to the market and during its real usage. Data about system failures is gathered by bug reports submitted by users to a bug report subsystem and bug reports may be classified according to specific levels of severity. Experts opinion takes into account that for simple enough portions of code, reliability may be verified via code review or formal verification of source code [7].

In next section we briefly present Jugoistok as an electric supply company, and in section 4 we show the architecture, needed to monitor reliability of its information system.

III. INFORMATION SYSTEMS IN JUGOISTOK

The Jugoistok Power Supply Company in Nis, Serbia, is responsible for power management of southeast Serbia. In 2012, within the project Study on development feasibility of interoperable data exchange platform for the Jugoistok information systems, we have performed analyses of the current state of the company's information systems through existing applications, their mode of usage and internal and external communications [9]. Based on the performed analysis and its results, the importance of developing an integration solution for the company was confirmed. The main problem is that these ISs have generally not been integrated, resulting in a complex and inefficient working environment for the users. Every IS vendor in Jugoistok developed each system only to comply with requirements imposed by particular department or set of users. In order to satisfy demands from various users, each vendor tried to cover broader set of functionalities for processing large scales of different data.

A large number of business processes within electric power supply companies in Serbia, such as electric power distribution network planning, repairs, maintenance and reconfiguration, is based on the proper network model. Currently, the mentioned network models can be found in various information systems within electric power supply company. But in each information system that uses such network model, it is implemented it in a different way. Such different network modeling approaches make data exchange and manipulation between different systems within a company hard or even impossible. This produces a need for a specialized platform that should enable data exchange on electric distribution network. Usage of such platform will lower network model gap and would greatly improve the efficiency of everyday business processes related to the electric power distribution network.

Information systems in Jugoistok are mostly developed in different technologies using different platforms. Before the initial information integration has been introduced into the information system in Jugoistok, they were sharing data using point-to-point connections. Analysis has shown that in Jugoistok 20 distinct information systems are used. Majority of them (11 applications - 55%) have been developed by the IT Department of the Jugoistok while others have been developed by external partners. The same analysis has also shown that 85% of them (17

applications) use Oracle technologies, mainly Oracle database and related technologies like Oracle Forms and Oracle Reports. Besides Oracle technologies, the following technologies are also used C/C++/C#, .NET Framework, MySQL database, Perl, HTML, JavaScript, PHP, WordPress.

Analysis has also shown that majority of internal inter-information system communication is done through the shared database. This means that applications can directly access database tables, whose data are filled by other applications, or they access data through dedicated views or stored procedures. Rarely, communication is done by other, non-digital means of communication. These non-digital means of communication usually imply exchange of data written/printed on paper and then manual data insertion into the respective information system. Analysis has shown that on internal company level exist around 60 connections between applications out of which the whole 56 of them (93.33%) are connected directly through a shared database. Although it exists, percentage of manual integration and integration through dedicated communication services is negligible. All 60 identified connections between applications are point-to-point connections. This implies existence of a large number of different, specially adapted communication interfaces. Among other downsides that these point-to-point connections produce, maintaining so many dedicated communication interfaces clearly becomes a hard and error prone task.

All accounted characteristics of the information structure of the described system pose a requirement for implementation of the data integration and standard communication models.

Previously described problems produce a need for integration of various existing information systems and applications, as well as new applications, yet to be developed both inside and outside of the Jugoistok. In order to fulfill these requirements, the only solution is an implementation of information exchange infrastructure. This infrastructure needs to be adaptable and extendible so it can meet the future requirements for information integration. It needs to provide common model which can be used in different technologies and with different integration platforms.

Applications of the business information system of the Jugoistok can be organized in four logical groups, based on the functionality they carry out:

- Technical information systems
- Business information systems
- Systems that use Data Warehouse
- Web portal

Technical information systems are the systems that contain technical data on power distribution network topology, objects connected to the network and other installations. These systems are SCADA, AMR, DMS, TIS and GIS. SCADA is a system for metering, monitoring and control of transformer stations. It provides access to data in real time as well as access to archive data needed for analyses and reporting. In order to provide real time data, it generates large amounts of data about current network state. AMR is the application for remote meter reading and it provides tools for both reading and collecting data and tools for collected data validation.

DMS is application for supervising, analysis, calculation and designing of the electric power distribution network. TIS is a technical information system that is used to store data on the objects and equipment of the electric power distribution network. Stored data are related to feeders, transformer stations, cables etc. GIS is a Geographical Information System for maintenance, evidencing and analysis of the electric distribution network. In Jugoistok GIS is used for mapping the whole network from the low voltage network up to high voltage network. It overlays data related to consumers, transformer stations, cables and the rest of the network infrastructure over raster maps. The whole network is shown in either schematic or line drawing modes.

Business information systems are the systems used for internal business processes of the company. They are related to employees, documenting and non-core business assets management (vehicles, buildings, HR etc.). These systems don't have anything to do with the primary business of the company.

Systems that use Data Warehouse can't be found at the moment in Jugoistok but are planned to be introduced in the near future. It will be implemented in the form of the central Data Warehouse repository, which will receive data about network and consumers from other information systems of the company. Systems that use Data Warehouse are used for advanced analyses over existing users' data and data on company's product/services usage. Based on these analyses, Data Warehouse based systems generate new information which is invaluable for future company success. Systems that will consume data from mentioned central Data Warehouse repository will be used for advanced analyses over existing consumers, their load profiles, energy consumption, electricity loses etc.

Web portal is an online presentation of the company business but also an entry point of the company's information integration. Access to data through Web portal is controlled based on different user privileges. This way, different stakeholders in and around Jugoistok Company will be able to access relevant data that can support their day to day business including company employees in various departments, regulators, external partners and others.

Previously accounted systems have become necessary in everyday Jugoistok functioning. Nevertheless, based on the changes required by different users of different systems, each of them is constantly being further developed. This constant improvement of functionalities usually makes them get deeper into the company's various business processes. Although useful, this development also produces duplication of functionalities meaning, same functionality can be found in two different systems. For instance, DMS systems, in order to properly analyze electric power distribution network need access to network technical data and therefore store them such data locally. But the same set of data, although in different format, can be found in other technical systems like TIS. Similarly, if DMS would need functionality of pinpointing the location of the network failure in order to speed up field crews dispatching, it would need to be expanded with geographic component for visualizing network elements. This geo-component would also require
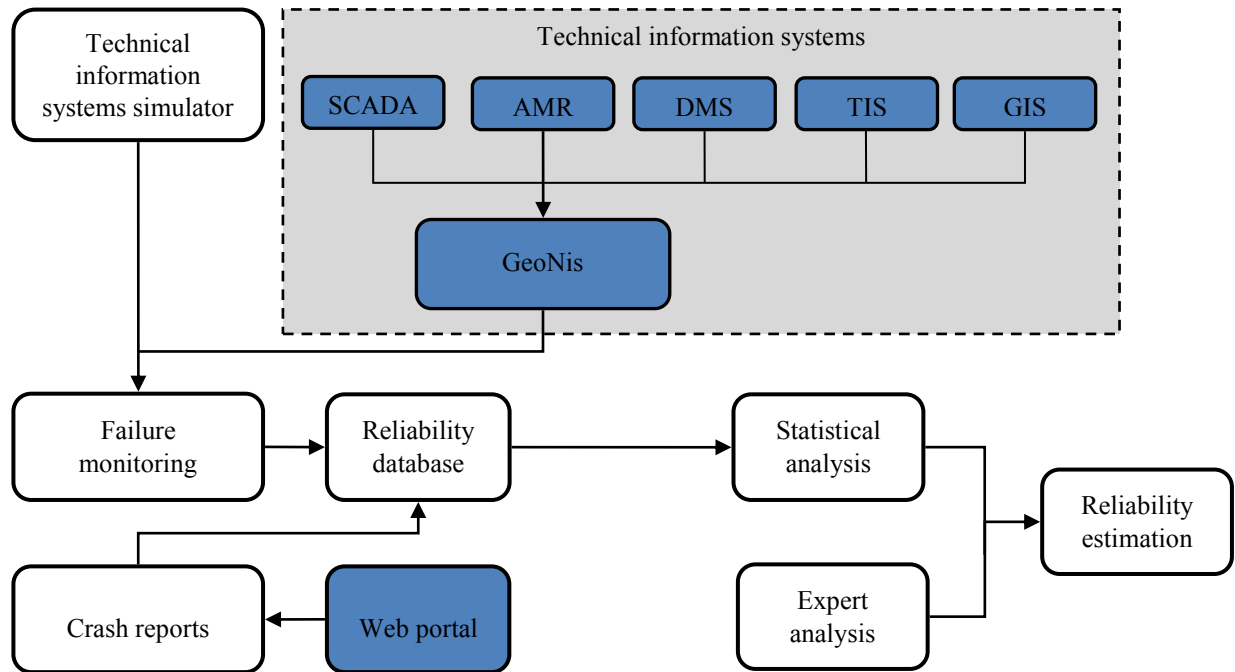
Figure 1.   Architecture for software reliability management Jugoistok (Legacy software components are marked with dark background)

geographic raster maps and vector layers of the network topology as well as additional geo-analyses features which could already be found in the GIS. On the other hand, if it is required to track state and quality of the service, network events should be paired with each consumer. Source of these data are DMS and technical systems. Such data from these systems could be used, for instance, in cases when consumer requires his power to be increased.

When developing a solution for information integration in Jugoistok, we have faced a concrete requests that among other have highly prioritized data management and integration of technical subsystems with field devices, controls and metering devices (sensors). The architecture of the integration system should also include a Web Portal component, as an important integration enabler, pointing to its role and position in the communication process. The proposed solution is based on GeoNis [10], our framework for semantic interoperability. GeoNis provides information integration solution for syntax and semantic heterogeneity using hybrid ontology approach.

Information from different applications integrated using GeoNis can be published through unique Web Portal [14]. Web Portal integrates information from different IS, and displays them in a consistent, user-friendly way. In this scenario, GeoNis framework acts as buffer between the portal and other systems.

IV.   ARCHITECTURE FOR QUALITY CONTROL AND MONITORING

The architecture of a system for monitoring of reliability of Jugoistok information system should have the following main components, as shown on Figure. 1:

- Technical information systems simulator – this module implements the basic workflows within the

technical information systems, but the respective functionality is implemented only as stubs. Hence, if there is a mistake in the workflow it is recorded into the database via the monitoring module.

- Failure monitoring – this module gathers data about system failures during all kinds of testing of the technical information systems, including black box, integration testing, etc.

- Crash reports – this module gathers respective data from user crash reports

- Reliability database – this is the central repository module in the system. It should hold the following information: (1) Failure number; (2) Failure type (user report, simulation failure, testing failure, etc); (3) Failure severity; (4) Time elapsed after the last failure; (5) Last failure number; (6) Last failure type; (7) Particular system/component within the technical information systems, where the failure occurred. This data is used as an input for the module for statistical analysis.

- Statistical analysis – this module implements one of the black-box models for analysis of software reliability [4]. This model should be applied for each module which is was tested, simulated or user feedback has arrived for it. However, currently no model is available, which takes into account different failure types, so all failures will be regarded as one single type of failure. In that case an existing tool (for example CASRE [8]) for software reliability analysis will be wrapped as a service and used within the proposed architecture.

- Expert analysis – this module is implements a user interface for experts to input their estimates about reliability of given parts of the system which are