Chapter 19

A Decision Support System for Planning and Operation of Maintenance and Customer Services in Electric Power Distribution Systems

Carlos Henrique Barriquello, Vinícius Jacques Garcia, Magdiel Schmitz, Daniel Pinheiro Bernardon and Júlio Schenato Fonini

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.69721

Abstract

This chapter aims to present the design and development of a decision support system (DSS) for the analysis, simulation, planning, and operation of maintenance and customer services in electric power distribution system (EPDS). The main objective of the DSS is to improve the decision-making processes through visualization tools and simulation of real cases in the EPDS, in order to allow better planning in the short, medium, and long term. Therefore, the DSS helps managers and decision-makers to reduce maintenance and operational costs, to improve system reliability, and to analyze new scenarios and conditions for system expansion planning. First, we introduce the key challenges faced by the decision-makers in the planning and operation of maintenance and customer services in EPDS. Next, we discuss the benefits and the requirements for the DSS design and development, including use cases modeling and the software architecture. Afterwards, we present the capabilities of the DSS and discuss important decisions made during the implementation phases. We conclude the chapter with a discussion about the obtained results, pointing out the possible enhancements of the DSS, future extensions, and new use cases that may be addressed.

Keywords: decision support system, system reliability, visualization tools, maintenance, customer services, electric power distribution

1. Introduction

A reliable and efficient power grid plays a central role in the infrastructure of a country, with a huge impact to its economic and social progress. An uninterrupted, secure, and efficient

power flow since generators to consumers is not only desirable but an important requirement for a reliable power grid, where electricity must be produced and delivered in the right amount at the right time.

Although electrical energy can be stored on energy storage systems (EESs), like batteries, to be available at the consumer side in case of a network fault, this is yet not an economically viable solution for bulky quantities [1]. Albeit rapid progress has been observed in the development of EES in the last years, mainly driven by the requirements imposed by the introduction of renewable energy resources into the electric power system, several drawbacks are still present. Most EES technologies are not yet mature, have very limited lifetime, low power/energy density and/or low storage capacity, and always have a round-trip efficiency penalty due to the required charge/discharge cycles [2].

Even in case of higher EES availability at the consumer side due to the current trend of technology evolution and price reduction, there is also the increasing penetration of renewable generation sources near to the consumer, known as distributed generation (DG). As such, DG sources are being adopted at a growing rate by the consumers; they also bring more challenges to the system operators and to the distribution infrastructure, imposing a higher demand for network reliability, mainly at distribution level [3].

This situation is even more challenging when considering the aging infrastructure of distribution systems, uncertainties and regulatory changes, the entry of new players into the energy market (e.g. prosumer), the constant pressure for cost reduction, and the more stringent requirements of reliability. On the contrary, the modernization of the distribution system with the deployment of information and communication technologies (ICTs) and the integration with operational technologies (OTs) is enabling a smarter grid and helping the distribution system operators (DSOs) to face those challenges in a better condition. Nevertheless, the deployment of those smart grid technologies, such as advanced metering infrastructure (AMI) and supervisory control and data acquisition systems (SCADA), also brings new decision variables to the DSOs and, consequently, the need of decision support systems (DSSs) to help them in their decision-making processes [4].

In the literature, several works have been published with proposals of DSS targeted to support the decision-making of the different players in the electric power systems, such as electric utilities, producers, consumers, and market regulators. In Refs. [5] and [6], DSSs were proposed for assisting the competitors in their trading strategies for the electricity market. Also, for the utility companies, there are proposals of DSS for the planning and expansion of the distribution system network based on geographic information system (GIS) [7, 8], for power grid operation and planning considering meteorological conditions [9], for analyzing disturbances of electrical power distribution transformers to diagnosing failures [10], and for assisting utilities to comply with the limits of sulfur dioxide (SO_2) emissions [11]. Additionally, some DSSs have been proposed for the producers and consumers, including support for the calculation of the solar energy potential of surfaces in urban landscapes [12], for managing energy-efficient buildings [13], and also for assisting consumers to participate in demand response programs [14]. There

are still proposals of DSS for assisting regulators in deciding about the installation of DG facilities [15] and about the deregulation of the electricity market [1].

Different from prior works, the main contribution of the DSS proposed here is in supporting the distribution system operators in the planning and the execution of customer and maintenance services, which are introduced in the next section. In the sequel, the design and the development of the DSS are reported in the third section. The evaluation of the DSS is left for the fourth section, followed by our concluding remarks at the end of the chapter.

2. Customer and maintenance services in the electric power distribution system

The electric utilities are responsible for maintenance services required to maintain the distribution network operating satisfactorily. Accordingly, the utility shall allocate its material and human resources (e.g. maintenance crews) in order to attend its customers and to fix the network as soon as possible in case of emergency (e.g. equipment failures and blackouts). However, due to the resources constraints, there is a need to prioritize emergency orders over normal maintenance orders, while taking the best decision possible regarding the available maintenance teams and the required response time for each order in the service queue.

The decision problem that must be solved by system operators working at the utility company is called the emergency dispatching and routing problem (EDRP) and can be stated as follows. Given pending emergency work orders and normal work orders in a service queue, and a set of available maintenance teams, to allocate the emergency orders to the crews in order to minimize the waiting time, which is defined as the sum of the travel time and the execution times of normal orders under execution or scheduled to be executed before the pending emergency orders.

It is important to note that the ERDP is dynamic by nature, given that new emergency orders may arrive in an unpredictable manner, changing the inputs of the problem and, thus, requiring a real time and continuous decision-making process. Therefore, a new dispatching solution must be found during the execution of the previously defined routes, possibly requiring changes to the routes assigned to the teams and the insertion of new emergency orders into their service queue. On the other hand, the decision process must take into account several conflicting criteria, such as the minimization of the required time to complete the emergency orders and the reduction of the cost incurred by the changes in the assigned routes. Note that the routing cost includes the delays of the previously assigned services and the possible decrease of the work efficiency, which is the ratio of effective service time to the total spent time (i.e. service times plus travel times). The relationship among the decision variables and decision outputs is represented in **Figure 1**, while the overall decision process is illustrated in **Figure 2**.

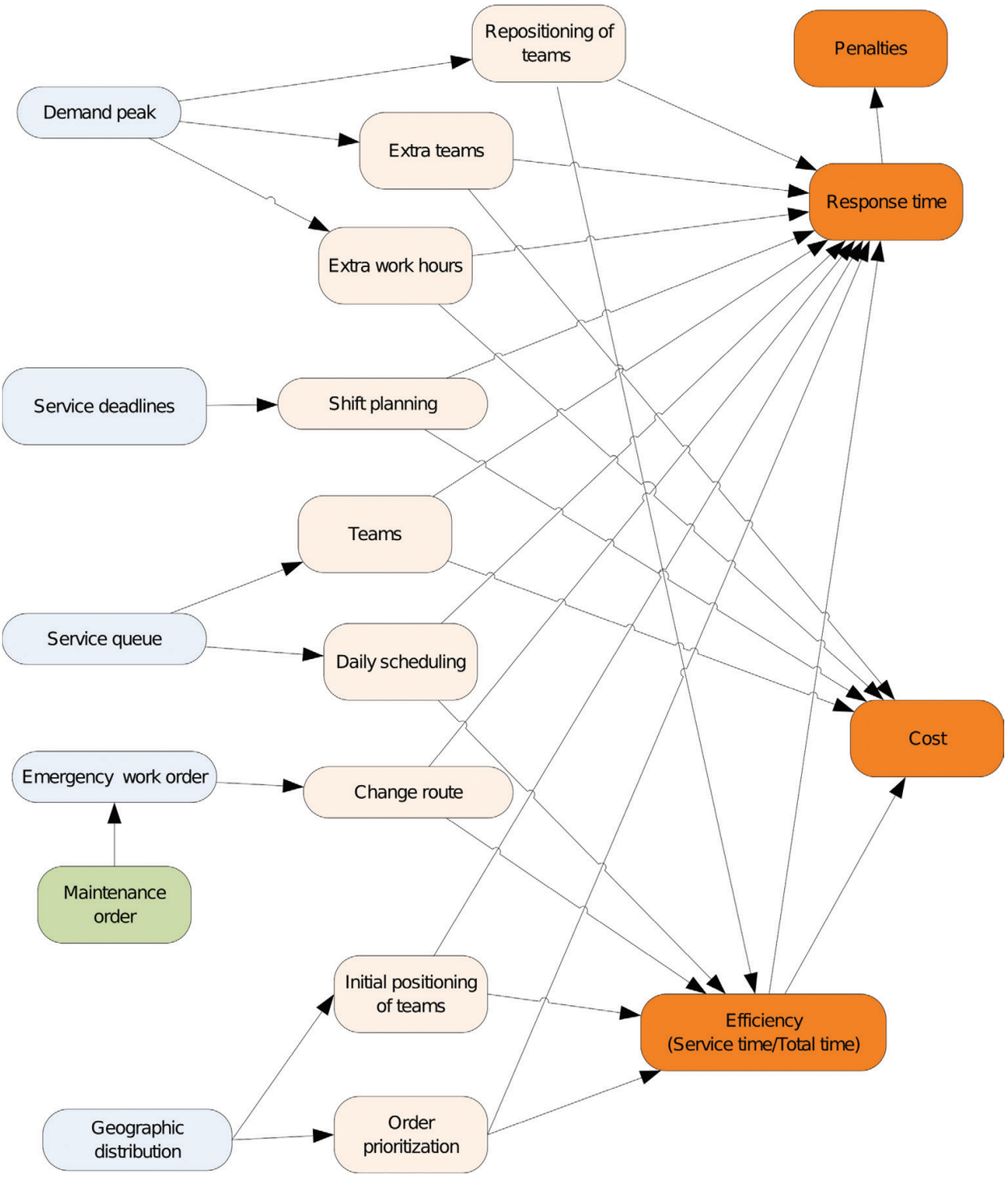

Figure 1. Diagram of the relationship among decision variables.

As shown in **Figure 1**, the main goals of the decision-maker are the increasing of the work efficiency and the reducing of the operational cost, the response time, and, consequently, the penalties incurred. Yet, the response time influences the calculation of the electric power distribution system (EPDS) reliability indices, such that the reduction of the response time has

http://dx.doi.org/10.5772/intechopen.69721

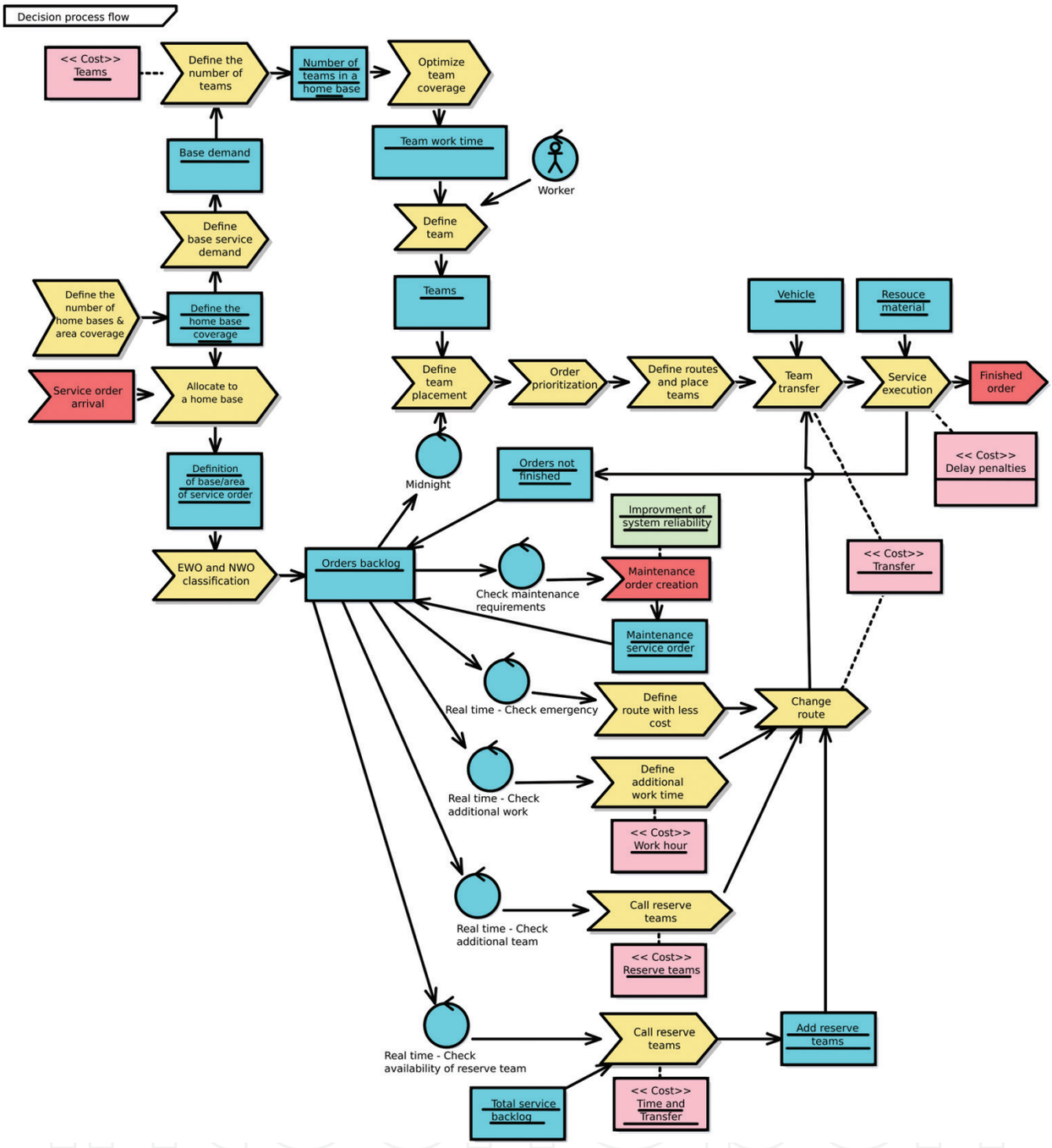

Figure 2. Decision process flowchart.

a positive impact in the improvement of the indices. Therefore, in fact, the decision-making process affects ultimately the EPDS reliability.

The assessment of the EPDS reliability is done by a set of reliability indices, whose definitions and forms of calculation can be found in Ref. [16]. In general, the reliability indices are measurements of the system performance and represent the frequency, duration, and magnitude of the interruptions of the electric energy supply. To this end, their calculations may take into

account the number of customers, the connected load, the duration of the interruption, the amount of power interrupted, and the frequency of interruptions.

In summary, the following reliability indices can be used: System Average Interruption Frequency Index (SAIFI), System Average Interruption Duration Index (SAIDI), Customer Average Interruption Duration Index (CAIDI), Customer Total Average Interruption Duration Index (CTAIDI), Customer Average Interruption Frequency Index (CAIFI), Average Service Availability Index (ASAI), Customers Experiencing Long Interruption Durations (CELID), Average System Interruption Frequency Index (ASIFI), Average System Interruption Duration Index (ASIDI), Momentary Average Interruption Frequency Index (MAIFI), Momentary Average Interruption Event Frequency Index (MAIFIE), Customers Experiencing Multiple Sustained Interruption, and Momentary Interruption Events (CEMSMIn). However, the most commonly used indices are SAIFI, SAIDI, CAIDI, and ASAI, which provide information about the average EPDS performance [16].

SAIFI and SAIDI give information, respectively, about the average frequency and duration of sustained interruptions experienced by the customers over a predefined period of time, usually 1 year, and a predefined area. Mathematically, SAIFI is the ratio between the total number of customers affected by the interruption and the total number of customers served in that area, while SAIFI is the ratio between the sum of the number of customers affected by each interruption multiplied by the restoration time of each interruption and the total number of customers served in that area. SAIFI and SAIDI are, respectively, given in Eqs. (1) and (2), where Ni is the number of customers interrupted for each sustained interruption event, Nt is the total number of customers served for the area, and ti is the restoration time for each interruption

$$SAIFI = \frac{\sum Ni}{Nt} \tag{1}$$

$$SAII = \frac{\sum ti \times Ni}{Nt} \tag{2}$$

CAIDI, by its turn, gives information about the average time needed to restore service. Mathematically, it is the ratio between the sum of the durations of customer interruptions and the total number of customers served. Equivalently, it is the ratio between SAIDI and SAIFI, as given in Eq. (3)

$$CAIDI = \frac{SAIDI}{SAIFI} \tag{3}$$

Finally, ASAI represents a fraction of time of the defined reporting period, usually 8760 h (equivalent to 1 year), which the average customer has received power. It is given in Eq. (4), where Nhy is the number of hours in 1 year and is equal to 8760 h in a non-leap year and 8784 h in a leap year

$$ASAI = \frac{Nt \times Nhy - \sum ti \times Ni}{Nt \times Nhy} = 1 - \frac{SAIDI}{Nhy} \tag{4}$$

http://dx.doi.org/10.5772/intechopen.69721

3. Decision support system design and development

The proposed DSS, named Pladin (a contraction of "dynamic planner"), was designed using a three-tier architecture based on a client/server model, as shown in **Figure 3**. The client/server model was chosen to allow the DSS to support its deployment in a cloud infrastructure. According to Ref. [17], cloud-based systems are attractive for utility companies due to the business model (i.e. memory/computation elasticity, scalability, economies of scale, pay-as-you-go model, etc.), beyond offering more efficiency, productivity, and anywhere access. Moreover, the deployment of a DSS in cloud is also interesting due to the possibly large variability of the required computing resources (e.g. memory and computing power) and the complete integration with web-based GIS portals when required [12].

In order to implement the proposed architecture, modern web technologies have been selected and employed according to the requirements of each layer of the DSS, as shown in **Figure 4**. On the server side, the decision models, the database access, and the middleware were implemented in Java language, where the middleware follows a Spring MVC [18] design pattern and the database access is managed by the Hibernate ORM tool [19]. On the client side, the user interface was implemented in Javascript language, following the React JS [20] and the Redux JS [21] frameworks. Additionally, the client/server data exchange is implemented as a web service Application Programming Interface (API) based on the Representational State Transfer (REST) architecture [22] using JavaScript Object Notation (JSON) data format [23].

The next step in the development process of the DSS was the design of the user interface, where special attention has been given to provide to the user visualization tools, such as maps and graphs, which are valuable for assisting her/him in the decision-making process given the spatial and temporal complexity of the problem at hand. Thus, the user interface has been prototyped and developed, according to the modeled use cases, which are shown in **Figure 5**.

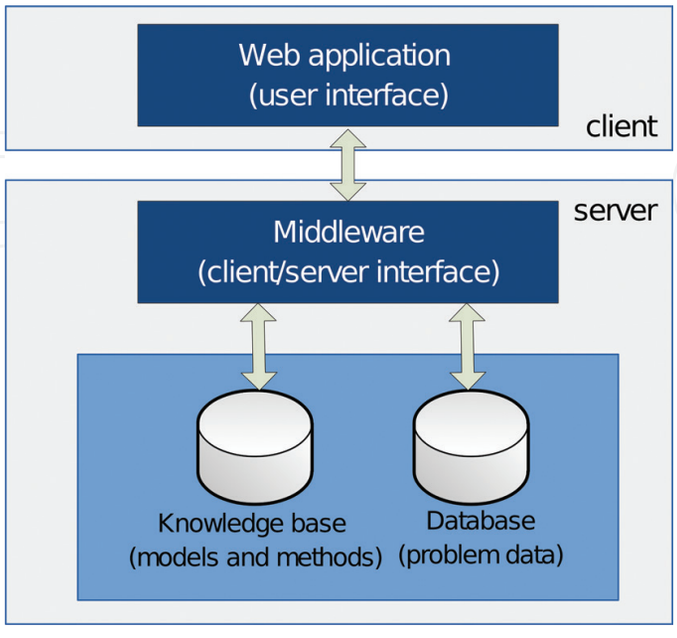

Figure 3. DSS three-layer architecture based on a client/server model.

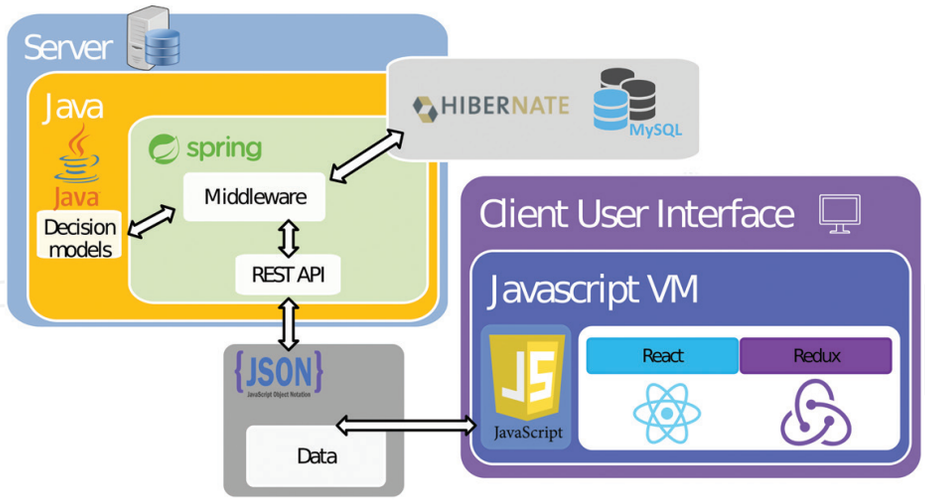

Figure 4. Web technologies used for client and server developments.

Due to the foreseen complexity of the user interface and the evolving requirements, it was decided to develop it as single-page application (SPA). Single-page applications are web applications written in Javascript that almost fully execute on the web browser at the client side, thus similar to desktop applications (i.e. applications that run natively). Some of the advantages of SPAs are flexibility, easy maintenance and deployment, rich visual experience, and a clear separation of concerns.

When developing an SPA, a choice of some architecture can ease the development and the maintenance of the application. There are basically four main architectures used in SPAs: model-view-controller (MVC), model-view-presenter (MVP), model-view-view-model (MVVM), and Flux. Flux is the most recent one and brings several advantages over the other architectures, mainly compared to the well-known MVC.

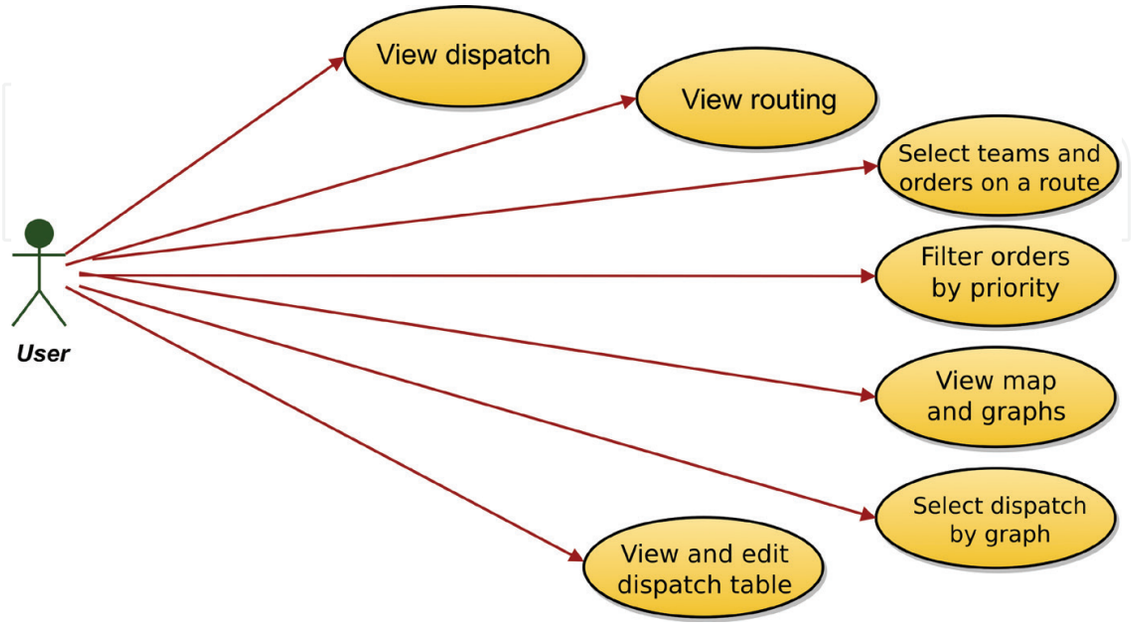

Figure 5. DSS use cases modeling.

http://dx.doi.org/10.5772/intechopen.69721

In the MVC architecture, as shown in **Figure 6**, the model is responsible for maintaining and updating the application data, the view displays the data to the user (in some visual form) and receives his/her inputs, and the controller is in charge of taking the user inputs to update the model and to notify the view of the updates of the mode. The main principle of the MVC is the separation between the three components, which allows the implementation of different views for the same model and improves testability of the components. However, the MVC architecture is not easily scalable due to the need of adding more instances of the components (models, views, and controllers) as the application grows and the increasing difficulty of reasoning about the logic of the application due to the bidirectional data flows among the components.

The lack of the scalability of the MVC motivated the introduction of the Flux architecture, which adheres to the philosophy of unidirectional data flow. In Flux, as shown in **Figure 7**, there are four components: the action creators, the dispatcher, the stores, and the views. The dispatcher, the stores, and the views are components with independent inputs and outputs, while the action creators are components that create, as the name suggests, special objects known as actions, which carry the data of the application (payload) and a unique property that identifies the action type.

The dispatcher is a central hub that manages all the data flow in the application by maintaining a registry of callbacks into the stores. These callbacks are used as a mechanism for distributing the actions to the stores, thus inverting the control of the application. Actions are generated in the application usually in response to the user interaction with the views or may come from the server. Then, each generated action goes to the stores via the dispatcher and is processed in the stores. The stores contain the application state (data) and logic, and are somewhat similar to the models in MVC. However, the stores can manage the state of many objects and not a single one as models in MVC.

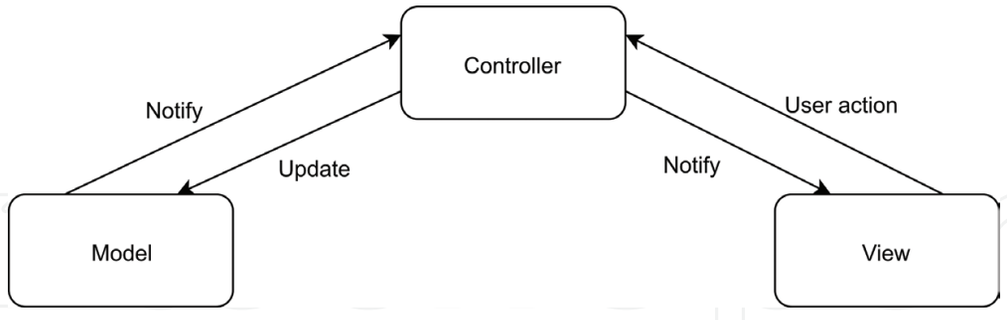

Figure 6. MVC architecture.

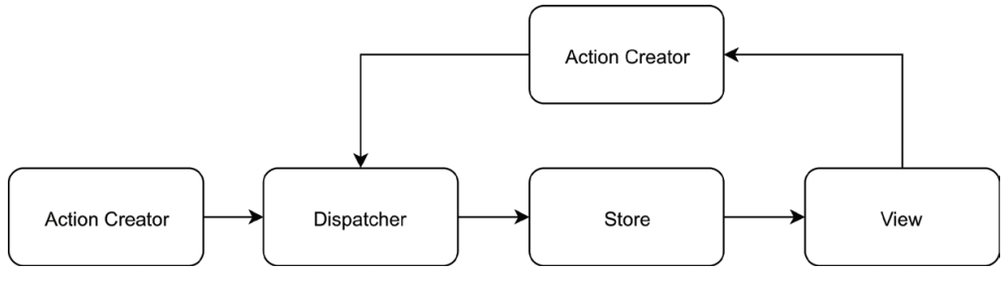

Figure 7. Flux architecture.

After receiving an action, a store interprets it according to the action's type using a switch statement. Then, the store updates its state and broadcasts an event declaring that its state has changed. In turn, special kinds of view, which are known as controller views, listen for events broadcast by the stores that they depend on. These controller views sit on top of a hierarchy of views and are able to obtain the data from the store and pass it down to their descendant views. Usually, the entire state of store is passed down the chain of views, so that views can take the part of the state they need and update themselves accordingly. Finally, views can create new actions based on the user interaction, closing the flow loop.

Basically, the Flux architecture avoids the complexities of the MVC architecture, mainly due to the restriction of unidirectional data flow in the application, which make easy for reasoning about the application workings. However, it is possible to simplify the Flux architecture even further in order for using a single store in the application. This concept is behind of Redux, which is an implementation of the Flux architecture, with added constraints in order to make it a predictable state container for the application, as shown in **Figure 8**.

Redux follows three fundamental principles: single source of truth, read-only state, and changes made with pure functions [21]. The first principle means that state of the whole application is stored in a single store as an object tree, which allows for easier debugging of the application. The second principle implies that it is only possible to change the state emitting actions, which are processed one by one in a strict ordering, avoiding race conditions and simplifying logging and debugging. And the third principle means that a new state of the application is created by a reducer, which is a function that takes as inputs the previous state and the action, and returns the next state without mutating the previous state. Therefore, because the state transitioning in Redux is done by functions (i.e. reducers) instead of event emitters as in the original Flux architecture, the Redux architecture does not have the concept of a dispatcher and allows easier composing of reducers. In fact, due to the several qualities of Redux compared to other alternatives architectures, such as simplicity, scalability, and predictability, it has been chosen as the basis architecture of the proposed DSS.

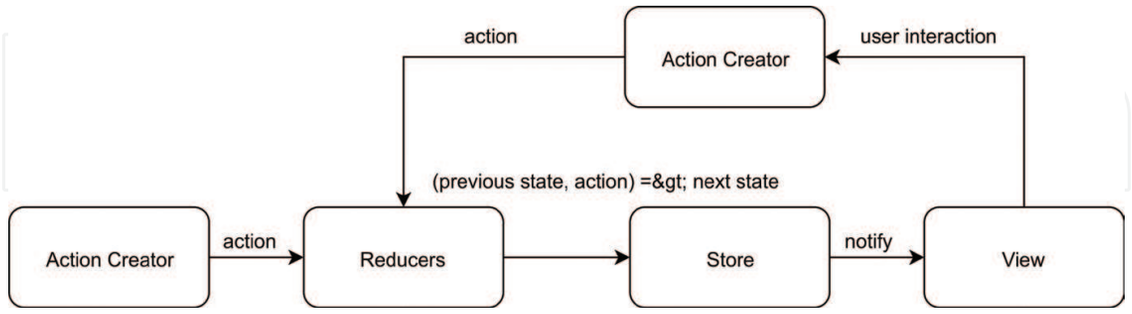

Figure 8. Redux architecture.

4. Evaluation of the developed decision support system

In this section, implemented use cases are presented and discussed. The first one is the "View dispatch" case, where the user can have an overview of a dispatch solution for a given

instance of the dispatching problem. The instance is based on the user selection of the position of an operating base and the target date. Then, the user can select from a list one of the available dispatching solutions for the chosen day in order to have a graphical overview of its characteristics, such as the number of crews and orders, the classification and the priority assignment of the orders and the distribution of the execution and travel times by orders and by crews. A sample screen of this use case is shown in **Figure 9**.

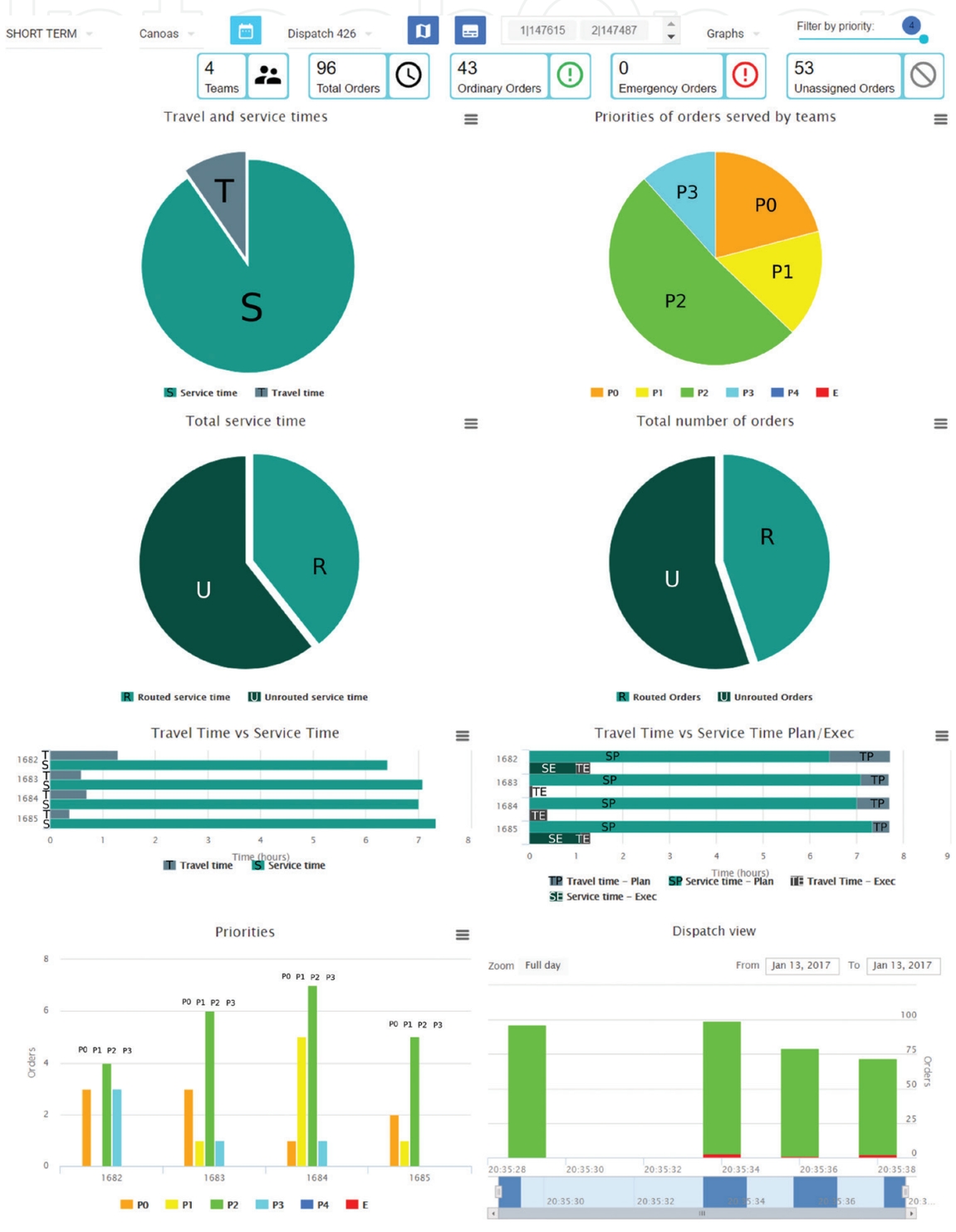

Figure 9. Sample screen of "View dispatch" use case of the Pladin DSS.

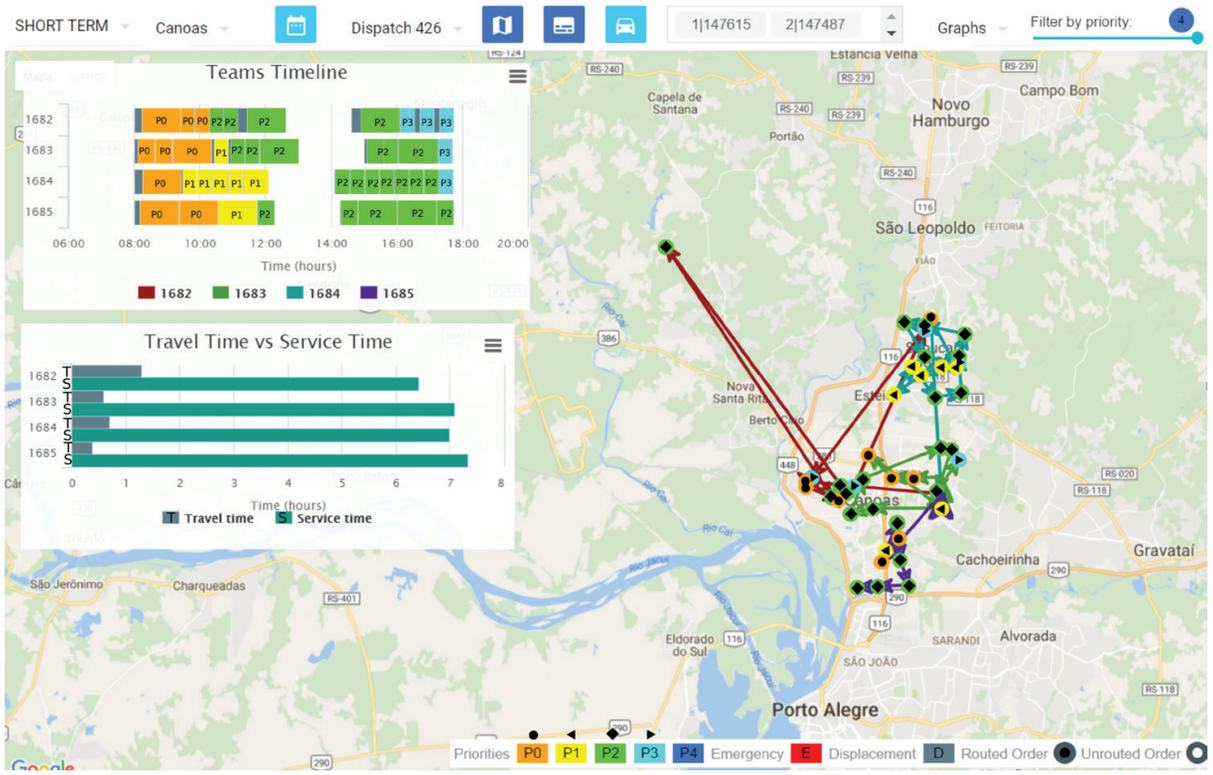

Figure 10. Sample screen of the "View routing" and "Show map and graphs" use cases.

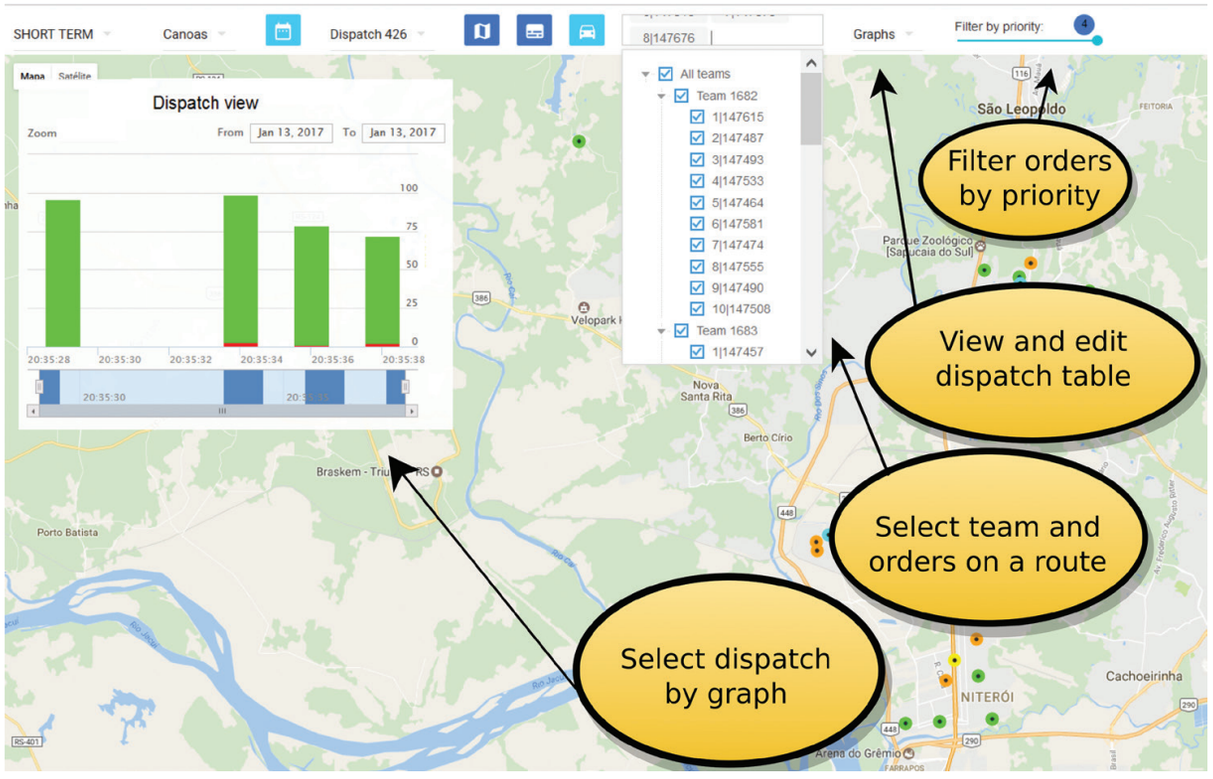

Figure 11. Sample screen highlighting the use cases "Filter orders by priority" and "Select teams and orders on a route," "Select a dispatch by graph," and "View and edit dispatch table".

After the user has selected a dispatching solution, he/she also can view the routes taken for each crew, according to the "View routing" use case. This use case allows the user to analyze the quality of the proposed dispatching solution with assistance of graphical tools, which are provided according to the use case "View map and graphs." A sample screen including both use cases is shown in **Figure 10**.

From the screen shown in **Figure 10**, the user also can access four other functionalities, which are implementations for the use cases "Filter orders by priority," "Select teams and orders on a route," "Select dispatch by graph," and "View and edit dispatch table." These use cases are highlighted in **Figure 11**.

Using the Pladin DSS, the user can better take his/her decisions for the planning and the execution of customer and maintenance services in the electric power distribution system. Therefore, he/she can appropriately allocate the available resources (e.g. teams) for improving the efficiency of the teams by reducing travel and response times. Thus, leading ultimately to reduce operational costs and improving the distribution system reliability.

5. Conclusions

In this chapter, we have introduced a web-based decision support system aimed to aid distribution system operators for planning and executing customer and maintenance services in the electric power distribution system. The DSS provides visualization tools for supporting the user decision to better allocating the available resources in order to solve the emergency dispatching and routing problem in real time.

In addition, the user interface of the DSS also allows for tweaking the input parameters, in order to change the solution for improving the work efficiency and reducing operational costs. Moreover, the user has access to facilities for analyzing what-if scenarios and studying the influence of the decision variables in the solution provided by the DSS.

Finally, due to the simplicity and the scalability of the chosen application architecture, it is expected that future extensions and enhancements will be added for improving it. Currently, the DSS is being used by the system operators working at a utility company located at the South Brazil, and thus the enhancements will be added after receiving the feedback of the current users. Possible enhancements and new use cases include the addition of more visualization tools and analytics services, artificial intelligence techniques, and the integration of more information sources.

Acknowledgements

The authors would like to thank the technical and financial support of RGE Sul Power Utility by project "Planejamento Dinâmico de Operações" (P&D/ANEEL), Coordination for the Improvement of High Level Personnel (CAPES), and the National Center of Scientific and Technological Development (CNPq).

Author details

Carlos Henrique Barriquello[1]*, Vinícius Jacques Garcia[1], Magdiel Schmitz[1], Daniel Pinheiro Bernardon[1] and Júlio Schenato Fonini[2]

*Address all correspondence to: barriquello@gmail.com

1 Federal University of Santa Maria, Santa Maria, Brazil

2 RGE Sul, Brazil

References

[1] Bergey PK, Ragsdale CT, Hoskote M. A decision support system for the electrical power districting problem. Decision Support Systems. 2003;**36**(1):1-17. DOI: http://dx.doi.org/10.1016/S1344-6223(02)00033-0

[2] Aneke M, Wang M. Energy storage technologies and real life applications – A state of the art review. Applied Energy. 1 October 2016;**179**:350-377. ISSN 0306-2619. Available from: http://dx.doi.org/10.1016/j.apenergy.2016.06.097

[3] Ipakchi A, Albuyeh F. Grid of the future. IEEE Power and Energy Magazine, March–April 2009;**7**(2):52-62. DOI: 10.1109/MPE.2008.931384

[4] Doğdu E et al. Ontology-centric data modelling and decision support in smart grid applications a distribution service operator perspective. In: Proceedings of the IEEE International Conference on Intelligent Energy and Power Systems (IEPS), 2-6 June 2014; Kiev. New York: IEEE; 2014. pp. 198-204. DOI: 10.1109/IEPS.2014.6874179

[5] Sancho J, Sánchez-Soriano J, Chazarra JA, Aparicio J. Design and implementation of a decision support system for competitive electricity markets. Decision Support Systems. March 2008;**44**(4):765-784. ISSN 0167-9236, http://dx.doi.org/10.1016/j.dss.2007.09.008

[6] Sueyoshi T, Tadiparthi GR. An agent-based decision support system for wholesale electricity market. Decision Support Systems. January 2008;**44**(2):425-446. ISSN 0167-9236, http://dx.doi.org/10.1016/j.dss.2007.05.007

[7] Yao FS, Zhang XQ, Zhang Y and Wang TH. Computer decision-making support system for power distribution network planning based on geographical information system. In: Proceedings of the International Conference on Electricity Distribution, 10-13 December 2008; Guangzhou. New York: IEEE; 2009. pp. 1-6. doi: 10.1109/CICED.2008.5211676

[8] Luo F et al. A practical GIS-based decision-making support system for urban distribution network expansion planning. In: Proceedings of the International Conference on Sustainable Power Generation and Supply; 6-7 April 2009; Nanjing. New York: IEEE; 2009. pp. 1-6. DOI: 10.1109/SUPERGEN.2009.5348306

[9] Li L, Yao-qiang X, Xing-zhi W, Kai W. Study on analysis and decision support system of power grid operation considering meteorological environment based on big data and GIS. In: Proceedings of the International Conference on Electricity Distribution (CICED); 10-13 Aug. 2016; Xi'an. New York: IEEE; 2016, pp. 1-6. DOI: 10.1109/CICED.2016.7576045

[10] Nina DLF, Neto JVdF, Ferreira EFM, Santos AMd. Hybrid support system for decision making based on MLP-ANN, IED and SCADA for disturbances analysis of electrical power distribution transformers. In: 2013 UKSim 15th International Conference on Computer Modelling and Simulation; Cambridge; 2013. pp. 12-20. DOI: 10.1109/UKSim.2013.147

[11] Ghandforoush P, Sen TK, Wander M. A decision support system for electric utilities: Compliance with clean air act. Decision Support Systems. October 1999;**26**(4):261-273. ISSN 0167-9236. Available from: http://dx.doi.org/10.1016/S0167-9236(99)00056-1

[12] Boulmier A, White J, Abdennadher N. Towards a Cloud Based Decision Support System for Solar Map Generation. In: Proceedings of the IEEE International Conference on Cloud Computing Technology and Science (CloudCom); 12-15 Dec. 2016; Luxembourg City,.New York: IEEE; 2017, pp. 230-236. DOI: 10.1109/CloudCom.2016.0047

[13] Perea E et al. Decision support system for distributed energy resources and efficient utilisation of energy in buildings. In: Proceedings of the 22nd International Conference and Exhibition on Electricity Distribution (CIRED 2013); 10-13 June 213; Stockholm. New York: IEEE; 2013, pp. 1-4. DOI: 10.1049/cp.2013.0557

[14] Sianaki OA, Hussain O, Dillon T, Tabesh AR. Intelligent Decision Support System for Including Consumers' Preferences in Residential Energy Consumption in Smart Grid. In: Proceedings of the Second International Conference on Computational Intelligence, Modelling and Simulation; 28-30 Sept. 2010; Bali. New York: IEEE; 2011, pp. 154-159. DOI: 10.1109/CIMSiM.2010.84

[15] Monteiro C et al. Spatial decision support system for site permitting of distributed generation facilities. In: Proceedings of the IEEE Porto Power Tech Proceedings (Cat. No.01EX502); 10-13 Sept. 2001; Porto. New York: IEEE; 2002. pp. 6 pp. vol.3. DOI: 10.1109/PTC.2001.964892

[16] IEEE Guide for Electric Power Distribution Reliability Indices. In: IEEE Std 1366-2012 (Revision of IEEE Std 1366-2003), May 31 2012. New York: IEEE; 2012. vol., no., pp.1-43. DOI: 10.1109/IEEESTD.2012.6209381

[17] Qin YB, Housell J, Rodero I. Cloud-Based Data Analytics Framework for Autonomic Smart Grid Management. In: Proceedings of the International Conference on Cloud and Autonomic Computing; 8-12 Sept. 2014; London. New York: IEEE; 2015, pp. 97-100. DOI: 10.1109/ICCAC.2014.39

[18] Spring MVC official website [Internet]. 2017. Available from: spring.io [Accessed: 2017-02-19]

[19] Hibernate ORM official website [Internet]. 2017. Available from: hibernate.org/orm [Accessed: 2017-02-19]

[20] React JS official website [Internet]. 2017. Available from: facebook.github.io/react [Accessed: 2017-02-19]

[21] Redux JS official website [Internet]. 2015. Available from: redux.js.org [Accessed: 2017-03-10]

[22] Fielding RT. Chapter 5: Representational State Transfer (REST). Architectural styles and the design of network-based software architectures [Ph.D.]. Irvine: University of California; 2000

[23] Bray T. RFC7159—The JavaScript Object Notation (JSON) Data Interchange Format [Internet]. March 2014. Available from: tools.ietf.org/html/rfc7159