An Intuitionistic Fuzzy Service Model: Use Case for Swiss Health Platform

Daria Balkenende, Roland Schüetze, and Andreas Meier

1 Service Level Management

The IT Infrastructure Library (ITIL) is a collection of books that contains requirements for specific practices of IT service management. According to ITIL service is 'a means of delivering value to customers by facilitating outcomes customers want to achieve without the ownership of specific costs and risks' [1].

The IT service is determined in a Service Level Agreement (SLA) between IT service provider and service customer. For example, SLA documents service level targets, and specifies the responsibilities of both the provider and customer. Service Level Management (SLM) is a management process that includes SLA negotiations and service design according to agreed service level targets. The goal of SLM is to find a balance between the customers' needs / expectations and the costs of associated service.

Regular service's quality assessments are essential, for this purpose ITIL provides an extensive collection of Key Performance Indicators (KPIs). KPI is a metric that is used 'to help manage a process, service or activity' and 'to ensure that efficiency, effectiveness, and cost effectiveness are all managed' [1]. The target level of each defined KPI and procedures in case of underperformance are specified in the SLA.

According to [2], service 'dependencies represent consumer/provider relationships between various cooperating components in a distributed system'. The operability of consumer component requires a service performed by provider component.

D. Balkenende • R. Schüetze (✉) • A. Meier
University of Fribourg, Fribourg, Switzerland
e-mail: daria.kurochkina@unifr.ch; roland.schuetze@unifr.ch; andreas.meier@unifr.ch

The software quality metric *coupling* was suggested by Larry Constantine at [3] as a degree of interdependencies between software modules. *Dependency coupling* can be applied to the SLM 'to capture how dependent the component or service on other services or resources for its delivery. Loose coupling describes an approach where integration interfaces are developed with minimum assumptions between sending/receiving parties, thus reducing the risk that failure on module will affect others' [4]. The goal within a Service Oriented Architecture (SOA) is to build components that do not have tight dependencies on each other, so that if one component were to die (fail), sleep (not respond) or remain busy (slow to respond) for some reason, the other components in the system are built to still continue to work. E.g. in the case of web application, the application server can be isolated from the web server and from the database.

The direction of the coupling is correlated to the direction of dependency between elements. The level of dependency between coupled Configuration Items (CIs) in service model can be expressed by two types of logical relationship:

– element A *is tightly coupled* to element B when the delivery of the services of A requires availability and successful delivery of services of element B. The relationship indicates the risks resulting from interdependencies.
– A *is loosely coupled* to B when A is able to deliver services independently from the state and output of B. It represents the resilience and mitigation capabilities of A.

Directly connected CIs are *directly coupled*. *Indirect coupling* is a relationship between two CIs which are connected via other CIs. The amount of elements between indirectly coupled CIs has an influence on the dependency coupling.

2   Literature Review: Service Quality Impacts by Couplings

The fulfilment of any higher-level objective requires proper enforcements on multiple resources at several levels. The challenge with such enterprise SLAs is translating metrics for business applications into measurable parameters for technical services that can be defined and reported against an SLA and monitored under Service Level Management (SLM). Service compositions, translation and mappings lie therefore in the core of SLA management, in that it correlates metrics and parameters within and across layers [5]. For example, in order to guarantee certain bounds on the response times for ERP-type, it involves the ERP software, the application and database servers, the network configuration, and more [6]. When knowing the relation and dependency of this backend service to the end-user service (or composite service), service administrators can then pro-actively track and verify these dependencies by periodically polling the measures of individual services and gathering the overall quality status of the end-user service. This will allow administrators responsible for the functioning of a service to monitor its quality based on the measurements typically already done for the infrastructure

components. When the dependency is between a service and some resource it uses, coupling will essentially be a function of how often the resource is used. For instance, the dependence of a service on the network layer might be measured by how often it is making a socket call, or how much data it is transferring. For web-services we can examine environmental coupling which is caused by calling and being called.

Traditional components are more tightly and statically integrated and measurements are related mostly to procedural programming languages [7, 8]. More advanced are object-oriented coupling measures [9] and further several metrics are proposed to evaluate the coupling level real-time by runtime monitoring, introduced as dynamic coupling metrics [10].

Managing the quality of virtualized, distributed and multi-tiered services is a hot topic in today's service research. Traditional approaches are measured bi-modal (means either operate correctly or fail) and concentrate on local technical IT performance measurements rather than with business-oriented service achievement. There are some more advanced approaches [11], including proposed models of QoS ontologies [12] or concepts that are based on Fuzzy Performance Relation Rules [6] or assessing on business and monetary impact information [13] on service levels to define efficient service level objectives.

The novelty of our approach lies in an integrated step-wise methodology, supported automated information assimilation, support of gradual failures or service degradations (e.g. predicting a partial SLA achievement) and bi-polar fuzzy impact assessments. Combining academic research with practice oriented business scenarios by expanding IT reliability engineering with fuzzy mathematical models provides high value to the service business, especially as the framework is general enough to be applied to any type of IT service.

3 Fuzzy-Based Intuitionistic Dependency Model

The fuzzy logic and fuzzy set theory was introduced by L. Zadeh in 1965 [14]. The fuzzy set is 'a class of objects with a continuum of grades of membership'. The membership function of an element in a set is valued in the real unit interval [0,1]. Fuzzy logic can help to specify granular terms, for instance, element A is more tightly coupled to element B than to element C with the membership of 0.9 and 0.8 respectively.

The theory of intuitionistic fuzzy set (IFS) further extends both concepts and was proposed by K. Atanassov in [15]. IFSs generalize fuzzy sets by allowing two functions be assessed to the element: membership μ and non-membership ν, both belong to the real interval [0,1].

The formal definition proposed by K. Atanassov: Let E be a fixed universe and A is a subset of E. The set $A^* = \{\langle x, \mu A(x), \nu A(x) \rangle | x \in E\}$ where $0 \leq \mu A(x) + \nu A(x) \leq 1$ is called Intuitionistic Fuzzy Set (IFS). Every element has a degree of membership (validity) $\mu A(x) : E \to [0, 1]$ and a degree of

non-membership $\nu A(x) : E \rightarrow [0, 1]$. Intuitionistic Fuzzy Sets (IFS) have only loosely related membership and non-membership values unlike classical [Zadeh] fuzzy sets. An IFS is a generalization of the classical fuzzy set which defines another degree of freedom into the set description, the independent judgment of validity and non-validity. For each IFS A in E, $\pi A(x) = 1\mu A(x)\nu A(x)$ is called the intuitionistic index of x in A which represents the third aspect, the degree of uncertainty or limited knowledge.

Let now a be the intuitionistic fuzzy logical statement of coupling with membership and non-membership $\langle \mu_a, \nu_a \rangle$. The coupling degree of truth is $\langle \mu_a \rangle$ and degree of falsity $\langle \nu_a \rangle$ with possible values between zero and one omitting that the sum of both degrees of truth is equal or less than one.

3.1   Intuitionistic Fuzzy Direct Coupling Index

Each node in the service model represents certain CI of the system. Each edge is a dependency between two nodes. As it was mentioned above the impact of one CI on another is defined as a dependency coupling. R. Schuetze in [4] proposed the algorithm of estimation the degree of the direct coupling between two nodes. The membership degree of the tightly coupling is estimated using inter-modular coupling metrics. The validity of loosely—by applying intrinsic component resilience capabilities. It can require the assistance of the specialist of the considered system to judge on the validity of both couplings and to specify the statements certainty. Both degrees can be also obtained via statistical analysis of the performance of each component or based on other parameters like mean time of failure.

For two components where y dependent on x the direct coupling can then be defined as V and calculated using formula below where μD is the degree of truth and νD the degree of falsity of the direct coupling impact between x and y.

$$V(dc(x, y)) = \begin{cases} \langle \mu D(x, y), \nu D(x, y) \rangle & \text{if } \langle x, y \rangle \in D \\ \langle 0, 1 \rangle & \text{if } \langle x, y \rangle \notin D \end{cases} \tag{1}$$

This degree is called Intuitionistic Fuzzy Direct Coupling Index (IFDCI) between component x and y.

3.2   Intuitionistic Fuzzy Indirect Coupling Index

The concept of indirect coupling calculation was invented by Kolev / Ivanov within the Fault Tree Analysis in 2009 [16]. The proposed Forward Impact Calculation (FIC) can help to determine the indirect impact of component y from component

x as:

$$V(idc(x, y)) = \begin{cases} v_{i,y \in D} idc(x, i) \wedge dc(i, y) & \text{if } x \neq y \\ \langle 1, 0 \rangle & \text{if } x = y \end{cases} \quad (2)$$

This methodology implements bottom-up approach of Forward Impact Calculation (FIC) and takes into account direct and indirect impacts. It can help, for example, to analyse how BS can be affected in case of a certain node's failure by following forward dependency direction.

The reverse task of finding elements on which a business process depends on can be solved by the root case analysis. This is a top-down approach and it refers to the Reverse Impact Calculation (RIC). The indirect impact is calculated starting from the dependant and traversing through its couplings in the reverse direction:

$$V(idc(x, y)) = \begin{cases} v_{x,i \in D} dc(x, i) \wedge idc(i, y) & \text{if } x \neq y \\ \langle 1, 0 \rangle & \text{if } x = y \end{cases} \quad (3)$$

Depending on the carrying information the intuitionistic fuzzy dependencies between components may be presented by functional or probabilistic semantic. According to this classical and probabilistic interpretations of the logical operations conjunction and disjunction are involved in calculation of indirect coupling. The impact between service model components is 'expressed by means of intuitionistic fuzzy values carrying probabilistic information' [16].

Example 1 Example: Considering an impact scenario between component $C2$ and service $B0$, the coupling relation is calculated as $indcpl_{best}(C2, B0) = (0.36, 0.51)$. Using probabilistic semantics it means that in case the component $C2$ fails, the expected probability that business service $B0$ breaches the SLA is 36% and 51% that the performance of $B0$ stays within the tolerated thresholds. An uncertainty of 13% is estimated which means this coupling relation is seen as quiet mature. As an example for a functional semantical interpretation using best an ordinary measurable coupling relationship this statement would mean that the service $B0$ is expected to be functional degraded or partly available (e.g. response time goes down by 36%) in case the component $C2$ performance fails. This allows a notion of having the business service still usable with some kind of degradation. Three types of impact calculations were introduced in [16]:

The worst case impact analysis involves the usage of classical conjunction and probabilistic disjunction to maximise the impact.

$$V(p \wedge q) = \langle min(\mu(p), \mu(q)), max(v(p), v(q)) \rangle \quad (4)$$

$$V(a \vee b) = \langle \mu(a) + \mu(b) - \mu(a)\mu(b), v(a)v(b) \rangle \quad (5)$$

The best case impact analysis uses probabilistic conjunction and classical disjunction:

$$V(p \wedge q) = \langle \mu(p)\mu(q), \nu(p) + \nu(q) - \nu(p)\nu(q) \rangle \tag{6}$$

$$V(a \vee b) = \langle max(\mu(a), \mu(b)), min(\nu(a), \nu(b)) \rangle \tag{7}$$

In a moderate analysis the logical operations can be either classical or probabilistic. If the dependency coupling has a probabilistic nature like e.g. calculations for expected overall system availabilities, then the moderate case can be calculated like:

$$V(p \wedge q) = \langle \mu(p)\mu(q), \nu(p) + \nu(q) - \nu(p)\nu(q) \rangle \tag{8}$$

$$V(a \vee b) = \langle \mu(a) + \mu(b) - \mu(a)\mu(b), \nu(a)\nu(b) \rangle \tag{9}$$

If the component dependencies have a min/max nature (e.g. the weakest component in a chain determines the maximal throughput) then the classical intuitionistic fuzzy operations are more applicable for the indirect impact calculation:

$$V(p \wedge q) = \langle min(\mu(p), \mu(q)), max(\nu(p), \nu(q)) \rangle \tag{10}$$

$$V(a \vee b) = \langle max(\mu(a), \mu(b))min(\nu(a), \nu(b)) \rangle \tag{11}$$

4 Applying the Concept to Swiss Health Platform

The following research is a case study which applies previously published research findings to business processes at client environment of the Centris AG [17].

4.1 Centris AG Service Model

Centris AG service model was created using data from ZIS System [18] configuration management database, ZIS settings, and verbal information collected from authorised employees. Collected data include the relevant CIs and interdependencies between them. The process of relevant elements extraction from ZIS System had the top-bottom direction—from top level representation of the service to the lowest level of the infrastructure. Each extracted configuration item got the type according to the level of infrastructure to which it belongs.

For the formalization and visualization collected data were entered and visualized in Gephi [19, 20] as a directed property graph. It is attributed, labelled, multi-relational graph which contains connected entities and which can hold attributes in the form of key-value pair [21]. The approach of application of property graph for SLA data management was proposed by Stamou [22] due to the fact

that all information can be encapsulated in the graph and that dependences can be represented by directed edges. The Centris AG service model consists of several layers which are correlated with the types of CIs (Fig. 1).

Additional business impact layer includes SLAs which are affected by system status, some of them are 'Cost-of-Failure', 'SLA_Availability' and 'SLA_Performance'.

The different layers are connected via the association relationship of dependence coupling which represents for each layer a different type of relationship with a different measurement of the coupling degree (Fig. 1). This generalization into one fuzzy coupling index allows an indirect coupling calculation over several logical tiers and different types of relationships. The determined index can be used to weight and connect impacts using monetary measurements to each component within the property graph (Fig. 2).
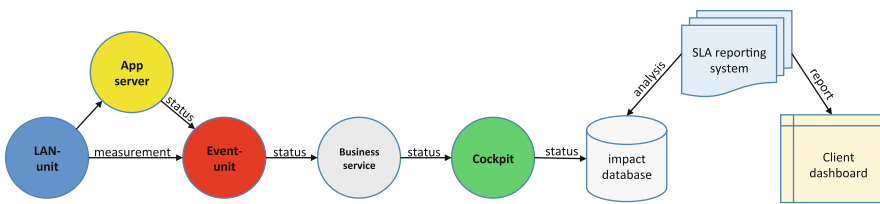

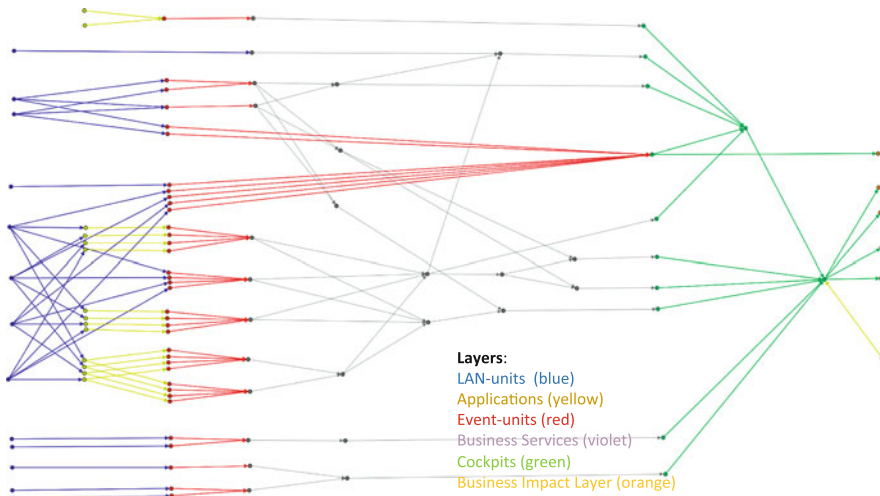**Fig. 1** Interaction of different layers of Centris service model

Fig. 2 SLA dependency graph with distributed layers

4.2   Determining the IFDCI for the Service Model

The dependency between two CIs is represented by the values of truth and falsity of an impact. The true value is correlated to the Ok Status Threshold (OST) of the top element. OST is a parameter added to each CI to represent the minimum conditions under which the status of the element is considered Ok. In most cases the status of CI is dependent on the statuses of connected predecessors. For example, the application server is considered ON when his OST $= 100\%$ and it can operate only when the lying below LAN-unit is up and running. Therefore, the application server is dependent on the LAN-unit with the level of (1, 0).

Recovery Time Objective (RTO) represents the time period needed to recover failed business process before the business is affected. The metric was used to define a true value of resilience. The correlation between RTO and resilience was estimated based on the technical specifications of the system. For example, resilience $= 1$ when failed element comes back to the normal productivity fast and have no impact on its business target. In opposite, resilience $= 0$ when the recovery takes more than 1 day.

The false value of resilience was calculated as (1—true value of resilience—π). The intuitionistic index is 0 when the knowledge about resilience is confirmed. In other case the expert judgement about knowledge perfection was used.

For each direct coupling the IFDCI was calculated based on assigned dependency and resilience values. The calculation of the indirect impact between nodes is based on the determined IFDCIs and presented above formulas.

4.3   Business Impact Assessment

The financial impact of each lower-level component on BSs was calculated multiplying the indirect coupling index with the monetary value of the impacted business. These analyses can be applied for the assessment of the business impact of incidents as well as for justifying the infrastructure changes. Calculated costs can be used in an IT infrastructure investment business case as an objective 'cost versus benefit' assessment.

The risk of the business impact calculation is presented by the level of uncertainty of assessments of dependency and resilience values. It is also presented in the calculated indirect impacts and accumulates all intuitionistic indexes of included dependences. The risk is caused by the fact that the nature of the 'rest value' is unknown and can be derived from resilience, dependency, or from both. As higher the intuitionistic value as lower the accuracy of the calculation and higher the risk. The assessments of the business impact and risk by means of intuitionistic logic can be accomplished as a new layer of the model. It can add the monetary value and specify the uncertainty of results which can be used for the infrastructure planning, business service impact calculation, SLA definition, and etc.

Due to the complexity of the Centris AG service model the calculations of indirect impact were automatized by the prototype. The first version of the prototype was developed on Python and Neo4j by Dennis Wohnsland with the goal to create a 'system that calculates indirect couplings between servers which are not directly attached to each other on the network diagram' [23]. Therefore, four cases of FIC were implemented: best, worst, moderate, and classical fuzzy. The determination of how one node is dependent on another starts from the considered node and follows the forward dependency direction to find all couplings between two nodes.

In the new version the RIC was implemented. The indirect impact is calculated starting from the dependent and traversing through its couplings in the reverse direction. It can be used for the root case analysis and helps to discover tight couplings which can affect the availability and productivity of the system. Another functionality which was added to the prototype is the calculation of the financial impact which the failure of certain CI can have on the business. It is based on the amount of money which will be lost per user per hour if the system will not work. It is possible to add monetary value to each SLA if there is a need for such information. The developed service model representation of the client environment consolidates the need of a new perspective in data presentation of the Centris CIs and related monitoring systems. It visualized for Centris AG configuration items from physical elements till business SLAs, and included all interdependencies and their rules. For the system specialists this model visualization provides an overview of the system environment, supported monitoring and relationships to business impacts and SLAs.

4.4 Case Study for a Clustered Load Balancer

The total evaluation of impacts for Centris AG client model revealed that the load balancer is the most critical component of the system and would cause the most financial impact. The reason of this was found in the created property graph. First, the load balancer is not clustered and nothing can substitute it. Second, in the real environment the whole system cannot operate without this component.

The Centris client infrastructure can be improved based on the found information. The clustered layout of the load balancer can increase resilience of this part of the system. Two clusters which work in parallel can manage higher amount of data. In case if one cluster is down the second one can operate independently and balance the whole load. But would it be reasonable from the business impact perspective? (Fig. 3)

This question was answered by comparing the results of business impact calculation for two cases. The first case represented the system with the single load balancer. In the second case the considered CI had two clusters. The comparison of both cases gave the view on the result of the clusterization. The benefit was estimated as more than a double decrease in a cost of failure in case of an incident and double increase in resilience from 0.352 till 0.703. The suggestion which can

Not-clustered load balancer:

Start	SLA_Sicherung	Cost_of_Failure	Money, CHF/hour	SLA_performance	Risk	SLA_availability
uxlpdps01_PingHeartbeat	0.000;0.000	0.925;0.075	18500	0.833;0.167	0	0.916;0.084
uxlpdps01	0.000;0.000	0.601;0.352	12025	0.541;0.417	0.046	0.595;0.359

Clustered load balancer:

Start	SLA_Sicherung	Cost_of_Failure	Money, CHF/hour	SLA_performance	Risk	SLA_availability
uxlpdps01_PingHeartbeat	0.000;0.000	0.425;0.575	8500	0.383;0.617	0	0.421;0.579
uxlpdps02_PingHeartbeat	0.000;0.000	0.425;0.575	8500	0.383;0.617	0	0.421;0.579
uxlpdps01	0.000;0.000	0.276;0.703	5525	0.249;0.732	0.021	0.273;0.705
uxlpdps02	0.000;0.000	0.276;0.703	5525	0.249;0.732	0.021	0.273;0.705

Fig. 3 Use case: clustered load balancer

be given within current project is that the system availability and fault tolerance can benefit from the clusterization of the load balancer (Fig. 1).

Practically, the real business impact is hard to measure due to the knowledge uncertainty, fuzziness of statements, and approximate cost values. Nevertheless, obtained costs provide a guidance on expected financial system impacts caused by components interdependencies and indirect cascading effects.

5   Conclusion

In conclusion, application of the dependency coupling and the intuitionistic fuzzy set theory are useful for the impact and root case analyses in Business Service Monitoring. It supports Service Management to proactively analyze environment monitoring data and suggest/execute appropriate reactions. The fuzzy mathematical models expands IT reliability engineering to bring new values and perspectives to the service business by combining academic research with business cases.

This study is based on previously published research results that are applied to the business processes and client environment of Centris AG. We developed a logical dependency model for Centris' eHealth platform environment based on real data from the company's monitoring environment information.

The dependency model is represented by a directed multi-relational property graph visualized by Gephi platform, stored, and analyzed by Neo4j database. The model is extended by the degree of the direct coupling that is estimated using the R. Schuetze algorithm [4]. The assessment of the degree considers two aspects of 'tightly coupled' and 'loosely coupled' relationships between the coupled components. Both levels of the coupling corresponds to the intuitionistic fuzzy degrees of truth and falsity of the dependency impact and resilience capabilities. The intuitionistic fuzzy indirect coupling index is estimated for all dependencies in the service model by using the approach that was proposed in [16].

The novelty and advantage of the IFDCI approach is the combination of bi-polar aspects of the coupling by bringing together positive and negative instances of the dependency relation: impact and resilience. Additionally, the knowledge uncertainty

is also included into the index which supports the process of assessment of the tight and loose coupling values.

The results of this study are implemented in a prototype that supports the steps of relationships definition and the assessment of the degree of the direct couplings. Additionally, it automates the calculations of the IFICIs for the model dependencies. Furthermore, this prototype implements the impact and root case analysis to support the prediction of the effects of incidents on the BSs. It relates a set of fuzzy-related components to a BS with corresponding performance parameters. The prototype could be used to support Centris Service Management to predict on impacts of monitored back-end component failures to BSs.

Developed within the prototype forward and reverse impact calculations are applied to the real world use cases, such as cost-of failure and analyses for infrastructure improvements like clusterizations of components. For the total business impact calculation all known and calculated information are combined to predict the consequences of a disruption of business processes and/or functions. Developed configuration management framework could be generalized to support the application of any type of IT service.

References

1. IT Process Maps: IT Process Wiki: The ITIL Wiki | IT Process Maps. Retrieved from September 2016: http://wiki.en.it-processmaps.com/
2. Keller, A., Blumenthal, U., Kar, G.: Classification and computation of dependencies for distributed management. In: Fifth IEEE Symposium on Computers and Communications, 2000. Proceedings. ISCC 2000, pp.78–83 (2009)
3. Stevens, W., Myers, G., Constantine, L.: Structured design. IBM Syst. J. **13**(2), 115–139 (1974)
4. Schuetze, R.: Intuitionistic Fuzzy Service Failure Impact Analysis (IFSFIA) - A Gradual Method for Bi-Polar Impact Assessment. Springer, Berlin (2011)
5. Li, H.: Challenges in SLA Translation - SLA@SOI European Commission Seventh Framework Programme (2007–2013) SAP Research, Dec (2009)
6. Joshi, A., Yesha, Y.: Managing the Quality of Virtualized Services. In: Proceedings of the SRII Service Research Conference (2011)
7. Dhama, H.: Quantitative models. J. Syst. Softw. **29**, 65–74 (1995)
8. Fenton, N., Melton, A.: Deriving software measures. J. Syst. Softw. **12**(3), 177–187 (1990)
9. Alghamdi, J.S.: Measuring software coupling. In: Proceedings of the 6th WSEAS International Conference on Software Engineering, Parallel and Distributed Systems (2007)
10. Quynh, P.T.: Dynamic coupling metrics for service - oriented software. Int. J. Electr. Electron. Eng. **3**(5), 282 (2009)
11. Robak, S., Pieczynski, A.: Fuzzy modeling of QoS for e-business transactions realized by e-Services. J. Appl. Comput. Sci. **16**(1), 69–79 (2008)
12. Sora, I., Todinca, D., Avram, C.: Translating user preferences into fuzzy rules for the automatic selection of services. In: Proceedings of 5th International Symposium on Applied Computational Intelligence and Informatics (SACI'09), pp. 497–502 (2009)
13. S Kieninger, A., Berghoff, F., Fromm, H., Satzger, G.: Simulation-based quant-ification of business impacts caused by service incidents. In: Proceedings of the 4th International Conference on Exploring Service Science 2013, vol. 143, pp. 170–185. Springer, Berlin (2013)
14. Zadeh, L.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)

15. Atanassov, K.: Intuitionistic Fuzzy Sets. Springer, Berlin (1999)
16. Kolev, B., Ivanov, I.: Fault tree analysis in an intuitionistic fuzzy configuration management database. Notes Intuitionistic Fuzzy Sets **15**(2), 10–17 (2009)
17. Centris A.G.: Effizienter IT-Dienstleistungspartner für Kranken- und Unfallversicherer. Retrieved from September 2016: http://www.centrisag.ch/
18. LeuTek GmbH: Home - LeuTek - A Company of USU Grou. Retrieved from September 2016: https://www.leutek.de/en/
19. Gephi - The Open Graph Viz Platform. Retrieved from September 2016: https://gephi.github.io
20. Bastian, M., Heymann, S., Jacomy, M.: Gephi: an open source software for exploring and manipulating networks. ICWSM **8**, 361–362 (2009)
21. Rodriguez, M., Neubauer, P.: Constructions from dots and lines. Bull. Am. Soc. Inf. Sci. Technol. **36**(6), 35–41 (2010)
22. Stamou, K.: Systematic SLA data management. In : Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion, pp. 63–68 (2014)
23. Wohnsland, D.: Dependency modelling in server networks with fuzzy mathematical methods. Internship Report, Stuttgart (2014)