19th Int. Workshop on IFSs, Burgas, 4–6 June 2015
Notes on Intuitionistic Fuzzy Sets
ISSN 1310–4926
Vol. 21, 2015, No. 2, 87–98

An intuitionistic fuzzy approach for IT service-level-management

Roland Schuetze

University of Fribourg, Switzerland
e-mail: roland.schuetze@unifr.ch

Abstract: Managing the quality of virtualized, distributed and multi-tiered services is a hot topic in today's service research. IT-centric service levels, written in IT technical terms need to be bridged to business-oriented service achievements. Due to the financial impact of Service Level Agreements (SLAs) there is great research interest in integrated management tools that automatically monitor the performance of multi-tier applications, autonomously warn for arising problems and predict in case of incidents on possible frontend impacts like end-user experience or other business implications. These problems are known as root cause analysis and business impact analysis, respectively.

In addition the impact of service levels defined for these technical services on customers' business processes, is difficult to estimate. Thus, it is a major objective to identify SLA's that directly affect the performance of customers' business departments.

The proposed concept is providing a bridge between business impacts to distributed systems and technical components by defining dependency couplings in a practical and feasible manner in order to satisfy aspects of the distributed and fuzzy nature of SLA dependencies.

Keywords: Service level, SLA, Business impact, Services quality, Intuitionistic fuzzy sets

AMS Classification: 03E72, 03E75.

1 Couplings and dependencies

1.1 Virtualized composite services

IT landscapes are inherently integrated and the fulfilment of any higher-level objective requires proper enforcements on multiple resources at several levels. It would be easier to monitor and manage Quality of Service (QoS) metrics related to individual services and the resources they use (such as storage, network, processing power, etc.). However, the virtualized service delivery model requires the composition of services to deliver the over-all service to the client.

SLA's are largely confined to the front stage, both in terms of the objectively measured variables as well as the perceived measures such as quality. The interactions between the individual services from different sources makes it hard to control the performance of the overall service or provide quality measures for it in dependency of the quality and performance of the underlying services [1]. 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 [2]. An initial concept called IFCFIA (Intuitionistic Fuzzy Component Failure Impact Analysis) was first published at ICIFS conference, Nov. 2013, Sofia Bulgaria [3], the following version includes several conceptual extensions and evaluation of implementation scenarios.

1.2 Loosely versus tightly coupling

Dependence Coupling is a measure that we propose to capture how dependent the component or service is on other services or resources for its delivery. Loose coupling describes an approach where integration interfaces are developed with minimum assumptions between the sending/receiving parties, thus reducing the risk that failure in one module will affect others. Loose coupling isolates the components of an application so that each component interacts asynchronously and treats others as a "black box". E.g. for a web application architecture, the application server can be isolated from the web server and from the database. Tight coupling on the other hand indicates that successful delivery of other services or availability of resources is a prerequisite for the completion of a service. 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 [4, 5]. More advanced are object-oriented coupling measures [6] and further several metrics are proposed to evaluate the coupling level real-time by runtime monitoring, introduced as dynamic coupling metrics [7].

1.3 Inductive measurement of coupling relationships

When an inductive approach is chosen to investigate for impacts between servers or services, historical data is collected from the actual server network and the performance behavior of related components is analyzed. While deductive methods are practical to use, when it comes to yes or no questions (e.g. "does server y crash if server *x* crashes?"), it is hard to get a mathematical expression on how much dependent servers are on each other. Therefore the inductive approach is chosen which allows to analyze the system behavior in a more exact way applying active or passive approaches. Within an active scenario robots are generating workload by simulating end-users and events. Passive supervision collects performance data by the monitoring instrumentation, which can be compared through the systems to find related effects and analyze on impact dependencies. Also experts can judge by their experience on collected data-series and define if performance attributes can be seen as coupled to some degree. This approach leads to more precise values than a pure estimation of a system architect. As opposite a deductive method would be applicable, where dependencies are not calculated based on data the system produces, but rather the system itself, for example plans system architects make or comparisons to other systems, which have a similar layout.

1.4 Pilot within a flexible hosting data-centre

To pilot this inductive approach and test if an inductive way of determining dependencies makes sense, real data extracts of an application hosting environment for mid-size customers were depicted here. The network serving as pilot is shown in the graph below (Figure 1). It is part of a larger network in a productive hosting environment. Two application servers host a SAP software, which can be accessed through one user interface, and share incoming requests equally. One database server is linked to both application servers, hosting a DB2 software. In addition one backup database server mirrors the first database server and handles incoming requests in case the first database server is not available. Each of the different sorts of data are based on a period of two months, even though the interval in which the data is collected differs. Both, the CPU usage data and the memory usage, are measured on an hourly basis. This is assembled as average of the preceding hour. CPU and memory data were collected for both application and both database servers. Robots in this context are programs running on separate servers, accessing the user interface of the SAP software performing the same steps a normal user would perform on the system. In addition Helpdesk tickets have been evaluated. The tickets are arriving in the service center which is responsible for that server network. Ticket numbers are collected as per-day amount and created due to a variety of reasons, ranging from user tickets to automatically created tickets because of system outages.

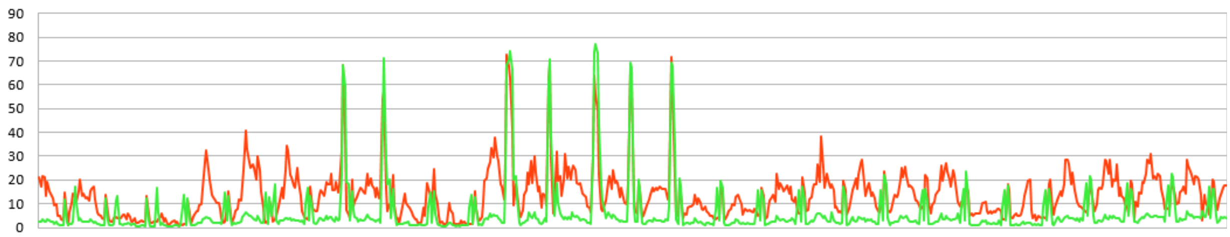

Figure 1. Comparison Database Server 1 - App Server 2

1.5 Assessment

Comparing CPU usage data, is a simple task, since no normalization has to be done and the interval in which the data was collected matches as well. When comparing one of the application servers with the database server quite a different view appears. Looking at figure 1 one may notice that in a kind of periodic behavior, the values seem to converge for a given time and then diverge again. Making assumptions about the dependencies of the two underlying servers is quite harder in that case, especially for statistical methods. For system architects or experts looking at the sample however it might be an easy task, since they may know why these differences in the convergence behavior appear – maybe due to a scheduled maintenance or automated backups of the database. But also it is clearly evident in the inductive data graph that the two underlying servers are dependent on a specific degree. As another example (Figure 2), the CPU usage of Application Server 1 and the Transaction Response Times of an arbitrary robot, which works as described above, is evaluated:

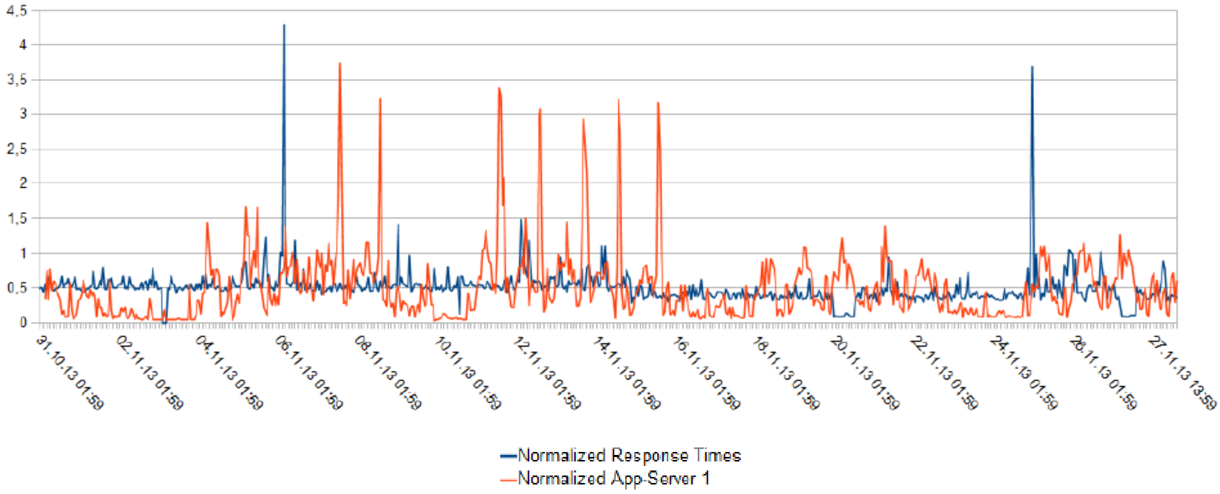

Figure 2. Comparison App Server 1 – SAP Transaction Response Time

This can be seen as an analysis on what influence the application server itself, namely the utilized CPU capacity has on the user experience, here measured as the response time of the interface. Since the CPU usage data range between 0 and 100% and the Total Response Times are a value given in milliseconds, the data series have to be normalized first, to be compared correctly. There are areas where the graphs seem to be synchronous, but since most of time they do not obviously match, an expert would have to judge that the two underlying systems are dependent on each other. Therefore the CPU of Application Server 1 and the User Interface can be assigned a low tight-coupling index. An expert may now argue that the CPU usage does not seem to be a bottleneck for the user interface and that there is an additional different limiting factor, like loading data from the disk, or even the network connection.

1.6 Creating fuzzy coupling rules

A comparison between different types of data should be supported by additional mathematical methods to get a meaningful assessment about coupling relationships like the following:

1. Normalization is a basic method to compare data which has a different value ranges.
2. Slope: Since the influence of an event in a network differs from server to server, and can usually be seen by a peak in the graph, taking only the slope of the graph resulting from the data series into consideration is a good approach.
3. Slope sign: An even more simplified approach is to not even take the actual slope of the graph into consideration, but only the sign of the slope. This approach follows the theory, that an event in a network which affects different components, affects them in the same way, but giving a different impact. If the sign of the slope of two data series graphs is equal for most of the time, they can be considered to be coupled.

Rules for interdependencies can be mined best from a large volume of historical performance data. Using the normalized and mathematical refined data-series, the performance of a service can be related by rules to the performance parameters of other services which it depends on their degree of coupling, as well as the resources it needs. Fuzzy sets provide the ability to classify elements into a continuous set using the concept of degree of membership. The characteristic function or membership function not only gives 0 or 1 for membership, but can also give values between 0 and 1. For instance, instead of expecting an exact numeric measure of dependence between two services, we could use a description like 'dependence is high'. The relation of a dependence measure to a linguistic term such as high or low will be captured in the membership function. (Figure 3). Fuzzy if-then rules can consider different interpretations of fuzzy implications. Once we have determined the fuzzy rules to define the performance measures, we can create linguistic rules for the service that will help to predict the impact to the front-stage service quality e.g.: *If {"Component Service" is tightly coupled to "Business Service" and ("Component Service Performance" is LOW or "Component Service Reliability" is LOW)} then "Business Service" performance is LOW.*

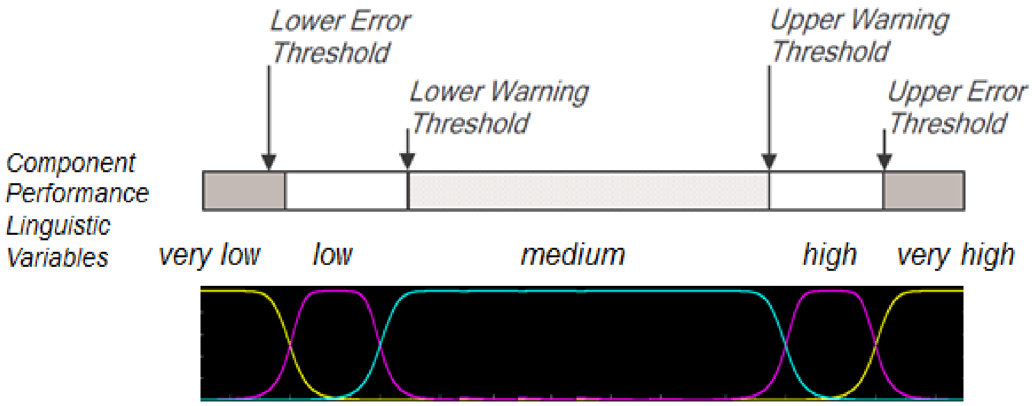

Figure 3. Map thresholds into linguistic variables

1.7 Bi-polar coupling aspects

A key principle of realistic impact simulations is the idea of considering both, positive and negative instances of the dependency relation and simultaneous consideration by pulling these strengths together. For a complex IT system the risk are the dependencies through interactions, the controversy mitigation ability are the built-in system resilience capabilities. The simultaneous and free play of contrary forces, dependence and resilience together will define the overall system behaviour and the expected impact to the business. Considering and judging positive and negative aspects isolated will not lead to reliable assessments. Further the intelligence in any complex system analysis will be the modelling of the indirect dependencies and interactions. There are several scenarios how an incident may interfere indirectly with other components which is mainly resulting out of the combination of the contrary forces. IT systems try to implement strategies that the resilience capabilities of each component should pro-actively limit the inference and impact of the incident to related components or the business services. In praxis impacts are complex which constitutes uncertainty. They involve a multitude of effects that cannot be easily assessed and may involve complex causalities, non-linear relationships as well as interactions between effects [8]. This may render it difficult to determine exactly what may happen.

2 Applying the model of intuitionistic fuzzy sets

2.1 Coupling statements as intuitionistic fuzzy sets

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) [9]. Every element has a degree of membership (validity) $\mu_A(x) : E \rightarrow [0, 1]$ and a degree of non-membership $\nu_A(x) : E \rightarrow [0, 1]$. Intuitionistic Fuzzy Sets have only loosely related membership and non-membership values unlike classical (Zadeh) [10] 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. This two-sided view, including the possibility to represent formally also a third aspect of imperfect knowledge could be used to describe many real-world problems in a more adequate way — by independent rating of both, positive and negative aspects — for each variable in the model. 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, indeterminacy, limited knowledge etc. In this approach let now a be the intuitionistic fuzzy logical statement of tightly coupling and b of loosely coupling with membership and non-membership $\langle \mu_a, \nu_a\rangle$ and $\langle \mu_b, \nu_b\rangle$. The tightly coupling degree of truth is $\langle \mu_a\rangle$ and degree of falsity $\langle \nu_a\rangle$. The same is done for assessing the loosely coupling index $\langle \mu_b, \nu_b\rangle$, respectively.

2.2 Defining the intuitionistic fuzzy direct coupling between two components

The validities (membership degrees) for tightly and loosely couplings are independently estimated by separate approaches, for 'tightly' using the described inter-modular coupling metrics and for 'loosely' applying assessed intrinsic component resilience capabilities. In praxis dependencies are naturally expressed by positive forms (membership) only, which is the way human assessments work. Thus, the proposed method does only require the experts to judge on the validity of the coupling and to specify a level of assumed certainty. The vagueness is expressed in linguistic terms and mapped into a crisp number with regard to the applied complement function (omitting for Sugeno $\lambda \geq 0$ — for Yager $w \leq 1$). The non-validity can then be automatically set by fuzzy complements (Figure 4 and 5).

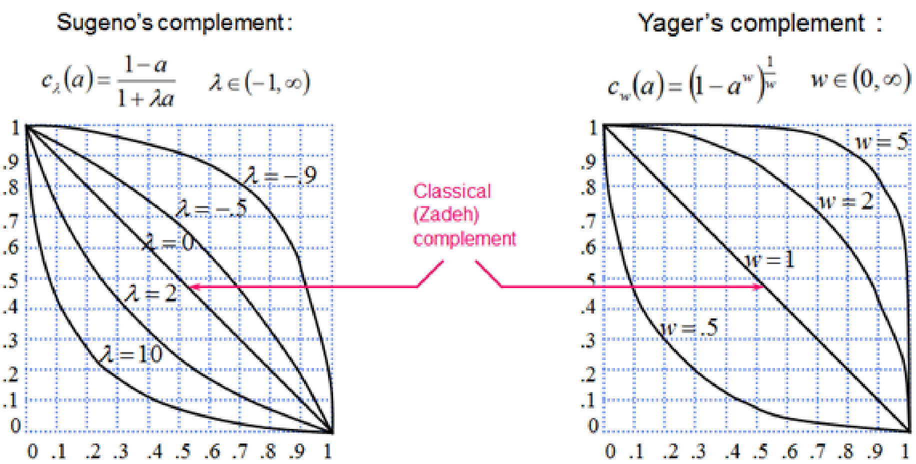

Figure 4. Sugeno and Yager fuzzy complements

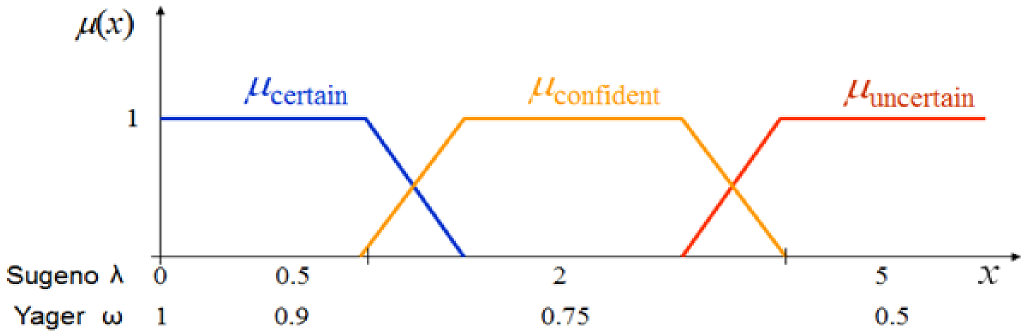

Figure 5: Certainty mappings for Sugeno and Yager

To define now the direct Coupling C association between two components the intuitionistic fuzzy logical statements of tightly coupling and loosely coupling are pulled together. Several operations over IFS are possible. As tightly and loosely couplings have contrary effects a meaningful operation for building the combined IFS C is for instance $A@\neg B$ by adding membership 'tightly' with non-membership 'loosely' and vice versa divided by 2.

$A@\neg B$: Merge of the direct tightly and loosely couplings into $\mu_{combined}$:

$$\mu_{combined}(x) = \frac{\mu_A(x) + \nu_B(x)}{2} \quad \text{and} \quad \nu_{combined}(x) = \frac{\nu_A(x) + \mu_B(x)}{2} \tag{1}$$

$$V(\text{dirdcpl}(x,y)) = \begin{cases} \langle \mu_D(x,y), \nu_D(x,y) \rangle, & \text{if } \langle x,y \rangle \in D \\ \langle 1,0 \rangle, & \text{if } \langle x,y \rangle \notin D \end{cases} \tag{2}$$

This requires that tightly and loosely membership are weighted and normalized equally. The combined degrees are further referred as μ_D and ν_D for direct coupling index and is called the intuitionistic fuzzy probabilistic direct impact between two components. The direct coupling can now be formal defined, where V is the described evaluating function of the intuitionistic fuzzy coupling statement.

2.3 Calculation of indirect coupling impacts

In order to satisfy aspects of the distributed nature of SLAs in a multi-tier environment, after assessing the direct couplings the indirect impacts can automatically be calculated. The indirect coupling from component x to service y can be defined with formula 3 where i is the component directly coupled to y on the path from x to y. The possibility of both, a classical, probabilistic interpretation of the logical operations conjunction ($\wedge$) and disjunction ($\vee$) is a key concept in the proposed indirect impact calculations. This is derived from the Fault Tree Analysis concept of B. Kolev and I. Ivanov in 2009 [11].

$$V(\text{indcpl}(x,y)) = \begin{cases} \bigvee_{i,y \in D} \text{indcpl}(x,i) \wedge \text{dircpl}(i,y), & \text{if } x \neq y \\ \langle 1,0 \rangle, & \text{if } x = y \end{cases} \tag{3}$$

The partial impact between the component and business performance is now expressed by means of intuitionistic fuzzy values carrying probabilistic information. The following IFS operations are proposed for classical, moderate, worst and best case impact analyses [11] (Table 1). Depending on which operations are applied, classical or probabilistic, the results will be greater or smaller.

Worst case impact analysis $V(p \wedge q) = \langle \min(\mu(p), \mu(q)),\ \max(\nu(p), \nu(q)) \rangle$ $V(a \vee b) = \langle \mu(a) + \mu(b) - \mu(a).\mu(b),\ \nu(a).\nu(b) \rangle$	**Best case impact analysis** $V(p \wedge q) = \langle \mu(p).\mu(q),\ \nu(p) + \nu(q) - \nu(p).\nu(q) \rangle$ $V(a \vee b) = \langle \max(\mu(a), \mu(b)),\ \min(\nu(a), \nu(b)) \rangle$
Moderate impact analysis $V(p \wedge q) = \langle \mu(p).\mu(q),\ \nu(p) + \nu(q) - \nu(p).\nu(q) \rangle$ $V(a \vee b) = \langle \mu(a) + \mu(b) - \mu(a).\mu(b),\ \nu(a).\nu(b) \rangle$	**Classical fuzzy impact analysis** $V(p \wedge q) = \langle \min(\mu(p), \mu(q)),\ \max(\nu(p), \nu(q)) \rangle$ $V(a \vee b) = \langle \max(\mu(a), \mu(b)),\ \min(\nu(a), \nu(b)) \rangle$

Table 1. Combined classical and probabilistic IFS operations [11]

The indirect intuitionistic fuzzy dependencies between components may have different kinds of semantics (functional and probabilistic) depending on the type of information they represent. Combinations of classical and probabilistic applications of the logical operations can as result be interpreted either as a probabilistic indirect dependency between component and the business performance (means the probability that a service breaches the SLA in case the component fails) or an ordinary indirect fuzzy dependency (means that the service is degraded in case the component performance fails). This allows a notion of having the service still usable with some kind of degradation (functional and/or probabilistic).

3 Impact analysis for gradual failure modes

In praxis, technical compliance is mostly measured bi-modal (either operate correctly or fail). This model can now be extended for granular failure impacts or service degradation effects and the consideration of several parallel incidents which causes the total impact. The direct coupling dependencies can be visualized within a directed graph representing the direct intuitionistic fuzzy impacts. The map consists of nodes and arcs between nodes. Each node represents a quality characteristic of the system. In the IT landscape model these characteristics could indicate the level of compliance to the SLA quality targets Each quality is characterized by a number *Ai* that represents its value and it results from the transformation of the SLA compliance level for which this node stands, in the interval [0, 1]. The tightly coupling model describes the causal relationships between two nodes. A decrease in the value of a quality parameter (QoS) or SLA compliance level would yield a corresponding decrease at the nodes connected to it via tightly coupling relationships, thus soft effects of partial functioning or degraded SLA compliance between IT components can be directly modeled by the same approach. This concept is briefly derived from the mathematical model of cognitive maps. In

1986, Bart Kosko [12] created the theory of Fuzzy Cognitive Maps (FCMs). A fuzzy cognitive map is a graph within which the relations between the elements (e.g. components, IT resources) can be used to compute the 'strength of impact' of these elements. FCMs are used in a much wider range of applications [13] which all have to deal with creating and using models of impacts in complex processes and systems. In the IT landscape scenario FCMs can be used to describe mutual dependencies between infrastructure and higher level IT components. The activation level of a quality parameter indicates then the level of SLA compliance (Figure 7).

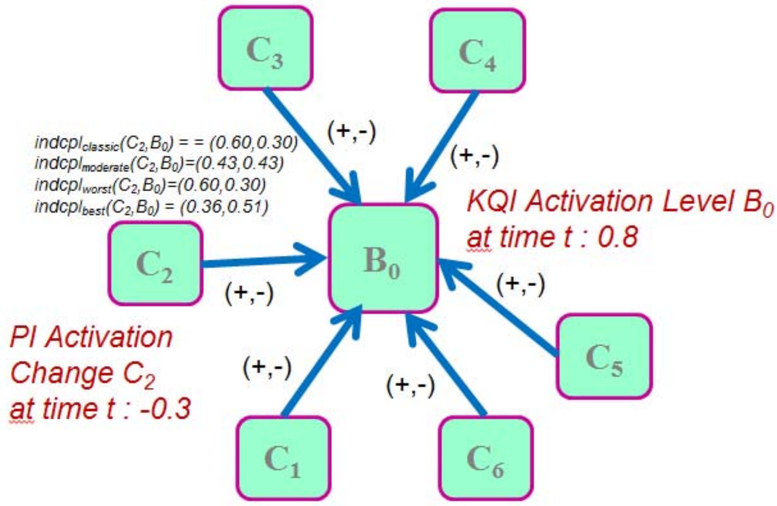

Figure 7. Couplings related to KQI activation levels

The model of the classical FCM is now leveraged to compute the value of each quality parameter that influenced by the values of the coupled quality indicator with the appropriate weights and by its previous value. So the value A_i for each quality indicator QIi can be calculated by the rule where A_i is the activation level of quality parameter QIi at time $t + 1$, A_j is the activation level of quality parameter QIj at time t, A_i old is the activation level of quality parameter QIi at time t, and W_{ji} is the weight of the dependence coupling between QIj and QIi , and f is a threshold function.

$$A_i = f\left(\sum_{j=1, j \neq i}^{n} A_j W_{ji} \right) + A_i^{\text{old}} \tag{4}$$

The weights of dependencies between the QIi and QIj could be positive ($W_{ji} > 0$) which means that an increase in the value of QIi leads to the increase of the value of QIj, and a decrease in the value of QIi leads to the decrease of the value of QIj. In case of negative causality ($W_{ji} < 0$), which means an increase in the value of QIi leads to a decrease of the value of QIj and vice versa. By adding also the activation levels of the QIs, each QI is characterized by a number A_i that represents its value and it results from the transformation of the SLA compliance level of this QI in the interval [0, 1].

As example: Using the Forward Coupling Calculation (FCC) method (applicable for Impact Analysis) of $indcpl(C_2,B_0)$ depicted in the example graph shows the indirect coupling of the Business Application B_0 on the Component C_2 with calculation of the KQI Activation Level for B_0 at time $t+1$ as follows using an activation level of $KQI_T\, B_0 = 0.8$ at point in time t.

- $\text{indcpl}_{\text{classic}}(C_2, B_0) = (0.60,0.30) \rightarrow QI_{T+1}\, B_{0\ \text{classic}} = (0.8 - 0.3 * 0.6) = 0.62$
- $\text{indcpl}_{\text{moderate}}(C_2, B_0) = (0.43,0.43) \rightarrow QI_{T+1}\, B_{0\ \text{moderate}} = (0.8 - 0.3 * 0.43) = 0.671$
- $\text{indcpl}_{\text{worst}}(C_2, B_0) = (0.60,0.30) \rightarrow QI_{T+1}\, B_{0\ \text{worst}} = (0.8 - 0.3 * 0.6) = 0.6$
- $\text{indcpl}_{\text{best}}(C_2, B_0) = (0.36,0.51) \rightarrow QI_{T+1}\, B_{0\ \text{best}} = (0.8 - 0.3 * 0.36) = 0.692$

In case the performance indicator C_2 decreases of 0.3, an impact between a decrease 0.108 and 0.18 to the quality indicator QI B_0 is estimated. This simple approach can be helpful when it is required to consider how several smaller improvements in total impact a business service performance as all impacts will be pulled together to the total effect on the business.

4 Indirect impact calculation and visualization using Python and Neo4j

As an opposite to the widely known SQL databases, graph databases like Neo4j [14] do not store their information in tables, but rather use graphs consisting of edged and vertices i.e. nodes and relation-ships to store information. While this approach is not appropriate for all kinds of data, it is more convenient and easier to use, when it comes to graph data that does consist of data objects and relationships e.g. as the application dependency graph in Figure 8.

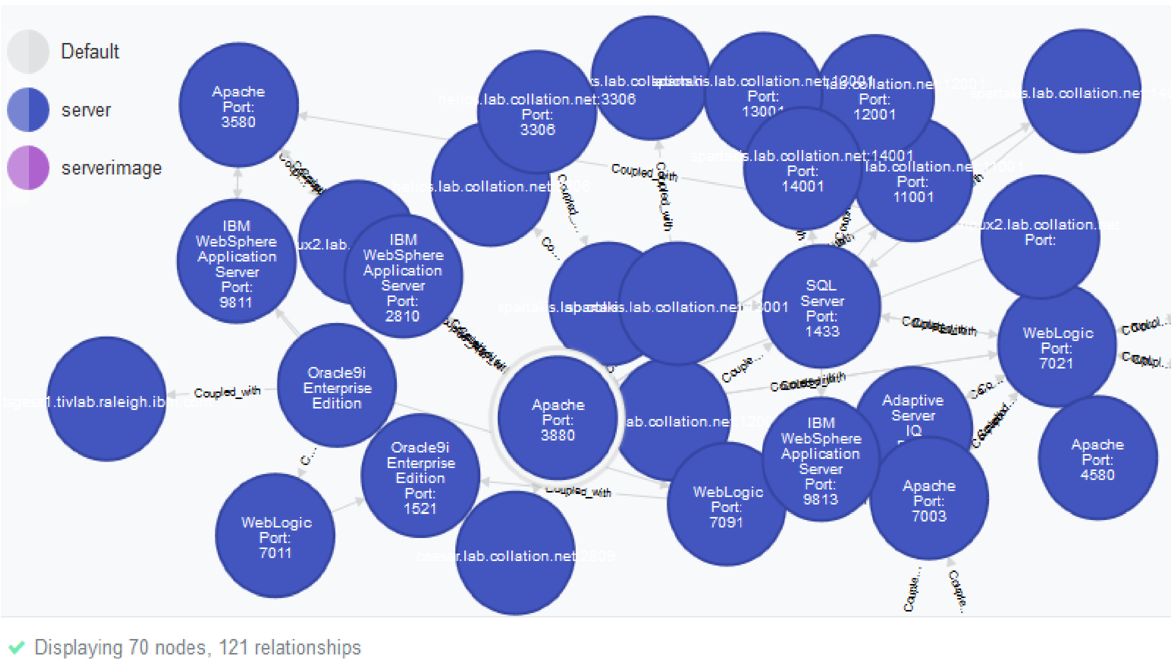

Figure 8: Loaded components and direct dependencies into Neo4j

The image above shows the discovered servers including the intuitionistic fuzzy direct impact loaded into the Neo4j database. For calculating indirect dependencies in server networks, graph databases suit perfectly well, since the given data is already in shape of a connected network and actions like path-finding, which are required for the impact calculations, are already implemented in the used graph database Neo4j. Being able to calculate the indirect dependency index for the discovered network, the impact of any component to any other can be expressed as intuitionistic fuzzy indirect impact by either getting the direct coupling for adjacent servers or calculating the indirect coupling based on the chosen IFS operations. To present the results to the user, the Neo4j browser is used, where a temporary graph is inserted into the database, which forms a star showing the chosen service in the center and all other components connected to it with the calculated indirect coupling levels (Figure 9).

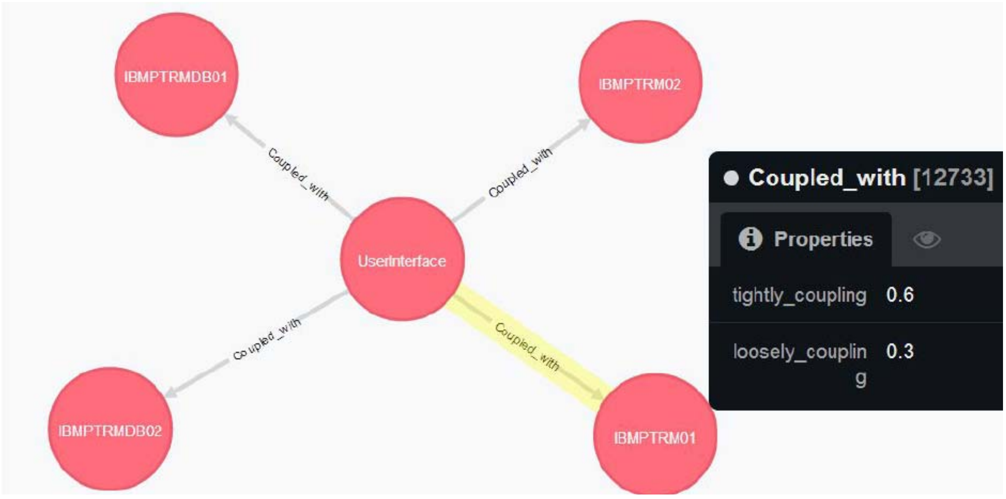

Figure 9: Star Representation of Indirect Dependencies

5 Conclusions

This approach maps service quality impacts to the idea behind intuitionistic fuzzy dependencies, where the level of tightly coupling between components corresponds to the intuitionistic fuzzy degrees of truth and falsity of the dependency impact and the loosely coupling index assesses the resilience capabilities of a service. 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 [15], including proposed models of QoS ontologies [16] or concepts that are based on Fuzzy Performance Relation Rules [1] or assessing on business and monetary impact information [17] on service levels. 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 intuitionistic fuzzy impact assessments. Combining well-grounded 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. The presented framework means transformation of availability and performance data into knowledge about the real-time status of business services that allows understanding and communicating the true impact of incidents on the business and vice versa.

References

[1] Joshi, K. P., Joshi, A., & Yesha, Y. (2011) Managing the Quality of Virtualized Services, *Proceedings of the SRII Service Research Conference.*

[2] Hui Li (2009) Challenges in SLA Translation – SLA@SOI European Commission Seventh Framework Programme (2007-2013) SAP Research, Dec. 2009

[3] Schütze, R. (2013) Intuitionistic Component Failure Impact Analysis, *Notes on Intuitionistic Fuzzy Sets*, 19(3), 62–72.

[4] Dhama, H. (1995) Quantitative models, *Journal of Systems and Software*, 29, 65–74.

[5] Fenton, N., & Melton, A. (1990). Deriving software measures. *Journal of Systems and Software*, 12(3), 177–187.

[6] Alghamdi, J. S. (2007). Measuring software coupling. *Proceedings of the 6th WSEAS International Conference on Software Engineering, Parallel and Distributed Systems*.

[7] Quynh, P. T. (2009) Dynamic Coupling Metrics for Service–Oriented Software, *International Journal of Electrical & Electronics Engineering*, 3(5), p. 282.

[8] Yang, H. Y. (2010) *Measuring Indirect Coupling*, Doctor Thesis at University of Auckland, New Zealand.

[9] Atanassov, K. (1999) Intuitionistic Fuzzy Sets: (Studies in Fuzziness and Soft Computing), Springer, Heidelberg.

[10] Zadeh, L. (1994) Soft Computing and Fuzzy Logic, *IEEE Software*, 11(6), 48–56.

[11] Kolev, B., & Ivanov, I. (2009) Fault Tree Analysis in an Intuitionistic Fuzzy Configuration Management Database, *Notes on Intuitionistic Fuzzy Sets*, 15(2), 10–17.

[12] Kosko, B. (1986) Fuzzy cognitive maps, *International Journal of Man-Machine Studies*, 24, 65–75.

[13] Stylios, C. D., Georgopoulos, V. C., & Groumpos, P. P. (1997) The use of Fuzzy Cognitive Maps in Modeling Systems, *Proc. of 5th IEEE Mediterranean Conf. on Control and Systems*, 518–527.

[14] Neo4j, World's Leading Graph Database. `http://neo4j.com/product/`

[15] Robak, S., & Pieczynski, A. (2008) Fuzzy Modeling of QoS for e-Business Transactions Realized by e-Services, *Journal of Applied Computer Science*, 16(1), 69–79.

[16] Sora, I., Todinca, D., & Avram, C. (2009) Translating user preferences into fuzzy rules for the automatic selection of services, *Proc. of 5th Int. Symp. Applied Computational Intelligence and Informatics (SACI'09)*, 497–502.

[17] Kieninger, A., Berghoff, F., Fromm, H., & Satzger, G. (2013) Simulation-based Quantification of Business Impacts Caused by Service Incidents, *Proc. of the 4th Int. Conf. on Exploring Service Science*. (Portugal) 2013 LNBIP, No.143. Springer. 170–185.