A Framework for Service Quality Assurance using Event Correlation Techniques

Andreas Hanemann
Munich Network Management Team
Leibniz Supercomputing Center
Barer Str. 21, D-80333 Munich, Germany
hanemann@lrz.de

Martin Sailer
Munich Network Management Team
University of Munich (LMU)
Oettingenstr. 67, D-80538 Munich, Germany
sailer@informatik.uni-muenchen.de

Abstract

Due to the increasing use of service level agreements for the provisioning of IT services, providers have to ensure that the guarantees for the quality of their offered services, expressed by quality of service parameters, are met. These guarantees are endangered by resource malfunctions in the provider's infrastructure or quality degradations of services which are supplied by subproviders. To optimize the resolution of customer reports about service quality degradations affecting the quality guarantees we propose to apply event correlation techniques.

In this paper we present a framework for this new kind of event correlation which is called service-oriented event correlation. The framework bridges the gap between the management of the infrastructure and the offer of services for the customers with respect to the service fault diagnosis. The application of event correlation approaches in the framework is examined in detail.

1 Introduction

In today's IT environments the offer of services with guaranteed QoS parameters has become a necessity for IT service providers[1]. For such a provider it is an important problem how to ensure that the agreements can be kept which are endangered by failures in the provider's resources including subservices which have been subscribed from other providers. Some of the possible resource problems may be partial (e.g., low quality of a subscribed subservice) or intermittent (e.g., sporadic failures in a fiber link) in nature. As time constraints like the Mean Time to Repair are often part of the service level agreements it has become a critical issue for a provider to quickly identify malfunctioning resources to shorten the fault resolution process.

In the area of network and systems management various approaches have been devised to deal with this issue. In this area a single fault can often cause a burst of subsequent failure events. To automatically cope with the amount of events and to retrieve meaningful information, event correlation approaches like rule-based reasoning, the codebook approach, or case-based reasoning have been devised. When implemented in today's commercial systems, the events called *resource events* denote resource failures mainly defined by the device vendors. This means that the events contain information about the network (e.g., link up/down) or end systems (e.g., authentication process crashed). However, these approaches fall short of accommodating customer trouble reports indicating a service quality degradation. As a result, the information about the service quality being affected by resource failures needs to be deducted and maintained by system administrators. As we have motivated in our previous work [8], *service events* denoting a service quality degradation need to be taken into account and have to be matched to the state of the underlying resources. An example for this is a slow data transfer being caused by a high link utilization. The high utilization cannot be regarded as a failure, but may lead to a violation of the agreed service quality. For coping with this issue, we have proposed to extend the use of event correlation techniques for the service events.

In addition to the resolution time minimization, the correlation of service events should lead to an effort reduction for the provider's incident management. Let us assume a situation where a provider offers services A and B and both services depend on another service C. In some situations it is then possible to identify a relationship between customer reports concerning services A and B, when it has been concluded that these services are affected by a malfunction of

[1] In this paper a *service* is defined as a set of functionalities that are offered by a *provider* to a *customer* with a guaranteed *Quality of Service (QoS)* specified in a *Service Level Agreement (SLA)*. The services are provided using other services called *subservices* and *resources* (e.g., network links, network components, end system memory, or end system processes).

service C. This relationship identification at an early stage of the event processing leads to an effort minimization for the provider as both kinds of customer reports can now be processed in an aggregated way.

A framework to receive these benefits by adapting event correlation techniques is subject to this paper. The main issues which need to be addressed by the framework are:

- How to transform customer reports about service quality degradations into events processable by the correlation engine?
- Which information is necessary about the service provisioning and especially about dependencies between services and subservices as well as between services and resources?
- Which event correlation techniques are suitable for the service-oriented scenario? How to minimize the effort for the correlation information maintenance with respect to frequent changes in the service provisioning?
- How can the provider get knowledge about service problems prior to the customers in order to gain more time for the problem resolution?

The rest of the paper is organized as follows. In Section 2 related work is presented to examine the contributions and limitations of the state-of-the-art. The framework for the service-oriented root cause analysis is presented in Section 3, while metrics to show the benefit of the approach in a real-world scenario can be found in Section 4. Conclusion and future work can be found at the end of the paper.

2 Related Work

In this section related work is referenced to show the current state-of-the-art for the IT service fault resolution. It comprises process management frameworks and event correlation techniques. Other related work relevant for components of the framework will be referenced during its presentation in Section 3.

2.1 IT Process Management Frameworks

The IT Infrastructure Library (ITIL) [18] is a continuously evolving collection of best practice documents with regard to the service management of IT service providers. It defines the process sets for service support and service delivery including incident and problem management processes. Process descriptions are derived from expert knowledge and written more or less in prose describing what has to be to done. No methodology is provided how the tasks can be performed especially no technique is proposed to actually perform a service fault diagnosis.

The enhanced Telecom Operations Map (eTOM) [6] published by the TeleManagement Forum is a business process framework for the telecommunications industry. eTOM is customer-centric and covers a broad range of important processes including processes for strategy, infrastructure, product, and operations. The Service Problem Management (SM&O-A) process deals with the diagnosis of service problems. However, this process is not described in a formal way, and neither input and output parameters, nor the linking of processes are described explicitly. Event correlation is briefly mentioned as a technique for dealing with problems on the resource level. Therefore, it remains unsolved how a service fault diagnosis should be performed.

2.2 Event Correlation Techniques

For applying event correlation for the framework, we examine existing event correlation methodologies for their reuseability for service-oriented event correlation.

Model-based reasoning: In model-based reasoning (MBR, [12, 17]) each component of an infrastructure is modeled with respect to its attributes, behavior, and relation to other models. The behavior of the whole infrastructure is a result of the interaction of the component models where each of them can either be a representation of a physical entity or a logical entity. The event correlation is a result of the collaboration of models.

This approach does not propose a detailed technique to correlate the events. Therefore, real-world systems like the GTE Impact system [12, 13] are often designed as a hybrid model-based/rule-based (see below) system.

For the service-oriented event correlation this approach is useful if it is possible to model each service as a logical entity. The behavior of a service may be difficult to describe in some scenarios as it is depending on the actual customer behavior.

Rule-based reasoning: In rule-based reasoning (RBR, [12, 17]) a set of rules is used to actually perform the correlation. The rules have to form "*conclusion* if *condition*". The condition contains received events together with information about the state of the system, while the conclusion may consist of actions which lead to changes of the system and can be input to other rules.

The rules in an RBR system are more or less human readable, so that their effect is supposed to be intuitive. Fast algorithms like the RETE algorithm [4] exist to actually perform the correlation.

In practice, the rule sets may become quite large which may lead to unintended rule interactions and makes it difficult to maintain the system. In addition, the system is going to fail if an unknown situation occurs which has not been covered by the rules so far.

Due to the correlation performance of rule-based reasoning algorithms this approach is useful for the service-oriented event correlation. However, the effect of the drawbacks of the approach has to be minimized, i.e. how to allow for a good maintainability of the system and how to deal with unknown situations.

Codebook approach: Like RBR the codebook approach [14, 19] proposes a correlation algorithm. This approach uses experience from graphs and coding. The input of this technique is a dependency graphs consisting of events and root causes as nodes and directed edges to represent the dependencies. After a graph optimization has been performed, the graph is transformed into a correlation matrix. The columns in the matrix represent the root causes, while the rows represent the events. In its simplest form the matrix cell entries can either be 1 or 0, denoting the presence or absence of a relationship between event and root cause. Values between 0 and 1 may be used to indicate the strength or likelihood of the dependencies. Techniques from coding theory can be applied for optimization. For example, some event rows may be deleted if the events do not lead to a discrimination of the root causes.

This approach has an advantage in comparison with RBR as it can - in some situations - deal with unknown combinations of events. These combinations can be mapped onto known combinations by using the Hamming distance. In contrast, the RBR approach may allow for a greater flexibility than the encoding schema.

This technique is also useful for the service-oriented event correlation as an efficient correlation algorithm is provided.

Case-based reasoning: In contrast to both techniques presented before the case-based reasoning approach (CBR, [16, 17]) needs no prior knowledge about the infrastructure. It contains a database of cases which have occurred before together with the identified root causes. While the first root causes have to be identified by hand, an automated matching to prior cases is performed at later stages.

The ability of this approach to learn from prior cases is useful for the service-oriented event correlation, even though case-based reasoning algorithms are less efficient than the algorithms for the techniques presented before.

Hybrid approach for dynamic situations: In [11] a hybrid approach which combines RBR and CBR techniques has been proposed to deal with highly dynamic situations (e.g. battlefield scenarios). In the proposed architecture an RBR and a CBR system run in parallel. The RBR engine uses temporal and spatial dependencies to correlate reported events, while the CBR engine makes use of prior situation templates. As the approach has not been implemented so far, details about the collaboration of the approaches are not available yet. According to the authors this work has been the first attempt to combine RBR and CBR techniques in the network and systems management domain.

3 Service-Oriented Event Correlation Framework

Our framework for the service-oriented event correlation is depicted in Figure 1. Here, we chose a data flow oriented representation in order to illustrate the main components of the framework as well as the information flow taking place among them. Newly introduced event correlations components are represented as dark gray boxes and the corresponding event correlation flows as solid arrows.

3.1 Input/Output Components

The *Customer Service Management* (CSM, [15]) is designed as an interface for the exchange of management information between customer and provider. In case of a service degradation, a customer can report this via the interface, while the provider forwards information about the current service status and scheduled recovery measures. An *Intelligent Assistant* is part of the *CSM* to generate service events from the customer reports. This is done by traversing a query tree to collect the information needed from the customer. This tree is composed of queries (e.g., how the customer accesses the service) and tests. The latter are either tests of the service or one of its subservices or tests of resources. They are performed by the *QoS probing* or *resource probing*, respectively.

The *QoS probing and measurement* performed by the corresponding component is done at the service access point. This means that the service is regarded from the customer's perspective without the need to know details about the inner structure of the service provisioning. The reason for this is to have a QoS definition independent from the provider's service realization which is demanded by the customers in order to easily compare the offers made by different providers. Important QoS parameters for the offered services are tested on a regular basis, which is also needed for generating customer reports. Tests on demand are also possible to allow for detailed problem examination.

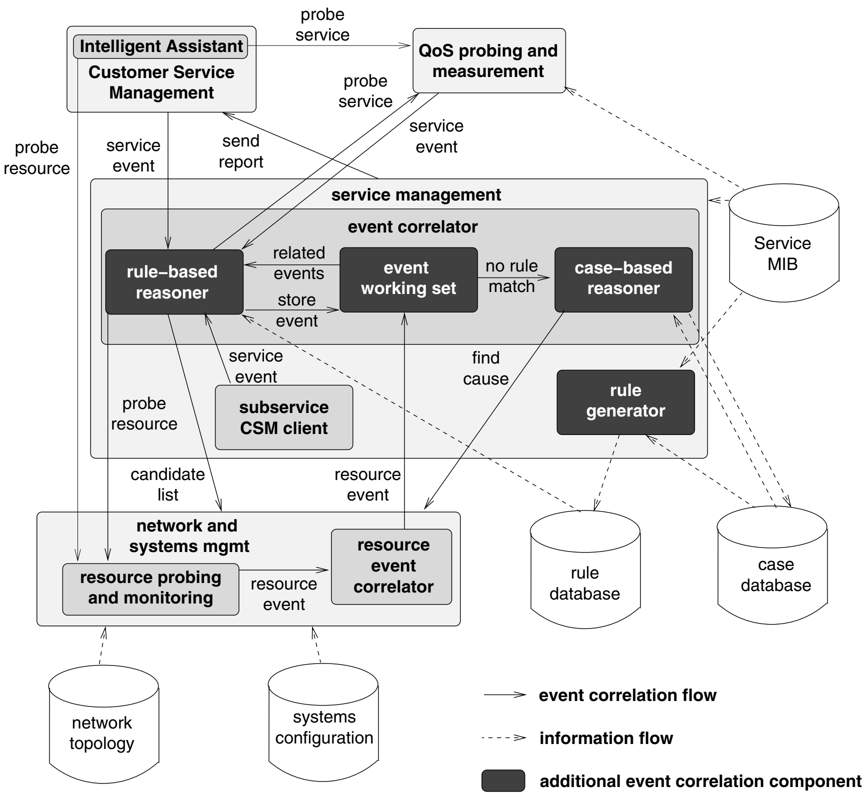

Figure 1. Framework for service-oriented event correlation

A QoS definition and corresponding measurement procedure appropriate for this component has been developed by Garschhammer [5].

The *network and systems management* component represents a management system like HP OpenView or IBM Tivoli. It contains the *resource probing and monitoring* which has to be in place to actively test resources either in regular time intervals or on demand and to passively monitor their behavior. Malfunctions are reported as resource events to the *resource event correlator* which denotes an event correlator designed for dealing with these events (using one the techniques reviewed above). The knowledge needed for the correlator, i.e. dependencies on the resource level, is contained in the network topology database as well as in the systems configuration database.

The *service management* is responsible for the provisioning of the offered services. A Management Information Base called "Service MIB" is accessed to retrieve necessary information about the service provisioning. It contains e.g., the dependencies on the service level and between services and resources. The QoS parameters for the services are also contained in the MIB which is important for the *QoS probing and measurement*. The *event correlator* for the service events is part of the *service management*. While a lot of work has already been carried out for the SLA definition and monitoring (see e.g. [17, 1]), a detailed methodology for a dependency modeling is still missing, even though an approach for an ISP scenario exists [2]. Methods to find the dependencies in an automated way have been proposed in [3, 7].

3.2 Event Correlator

We propose the use of a hybrid architecture for the *event correlator* dealing with service events and aggregated resource events. It consists of a *rule-based reasoner* and of a *case-based reasoner*. They use rules and cases stored in the rule database and in the case database, respectively. The idea behind this architecture is to cover as many situations as possible by the use of rules, but to apply experience gained from prior cases otherwise. The architecture can also be regarded as model-based.

Even though the number of service events encountered in an event burst may not be as high as the number of events in a similar situation on the resource level, a rule-based reasoning approach is used because efficient algorithms are applicable to actually perform the correlation. As many rules as possible are generated automatically from the knowledge about the service provisioning, especially from the different kinds of dependencies. This issue, which is addressed by the *rule generator*, is essential to reduce the effort necessary for frequent rule updates in rapid changing complex environments.

When applied in practice, it is often difficult to always have accurately defined rules due to the complexity of today's service provisioning. In situations, where the *rule-based reasoner* cannot cope with an event (determined by the time that has been passed since the event arrival), it is forwarded to the case-based reasoner. The case-based reasoner tries to find a match to prior cases or, if this is not possible, the root cause has to be found by the operation staff responsible for the *network and systems management*. The case together with its solution is stored in the case database. To improve the effectivity of the rules, these cases are used to generate new rules which is also a task for the *rule generator*.

This approach differs from the hybrid approach for highly dynamic situations as the case-based reasoner in our architecture is used only as a backup method to improve the modeling. For the correlation in the approach for highly dynamic environments it permanently tries to match the current system state to states which have been seen before.

The *event correlator* receives service events from the *CSM*, the *QoS probing and measurement*, and the *subservice CSM client*. The latter denotes a component which receives management information about subservices provided by subcontractors. The *resource event correlator* in the *network and systems management* sends resource events to the *event correlator*, while the result of the event correlation, a candidate list of possible resource failures, is transferred back to the *network and systems management*.

The correlation on the service level should be performed as a two-step process due to the different kinds of dependencies involved. At first, service events should be correlated with other service events. The rules for performing this step are related to the dependencies from services to subservices. The output of this step are services which may be affected by a resource failure. Their resources are identified in a second step by using the dependencies between services and resources. They are correlated with the aggregated resource failures received from the *network and systems management*.

As events and especially service events in most cases denote that something does not work properly, it is useful to integrate additional events about properly working services and resources. This is helpful to improve the correlation result by avoiding to have too many possible root causes. For doing so, the rule-based reasoner requests the *QoS probing and measurement* and the *resource probing* to probe services or resources, respectively.

Attached to the rule-based reasoner is the *event working set* storing events which have not finally been correlated. An event is contained in the working set until it is either correlated or the time window for a successful correlation has been exceeded. In the latter situation the event is forwarded to the case-based reasoner.

4 Assessment Metrics

The benefit of the approach needs be measured in concrete scenarios by applying one or more metrics. The aim of the provider is to improve its profit by lowering the cost for service fault management. The cost savings are a result of prevented SLA violations and the effort reduction in the event processing. On the other hand, the costs for maintaining the event correlation components need to be taken into account.

As the financial consequences are difficult to determine during the use of the approach, simpler methodologies should be used. A simple indicator for the effort reduction would be the percentage of service events that have been correlated to other events and do not need to be treated as isolated events anymore. Problems of this metric are the treatment of false positives (i.e. incorrect correlation of not relelated events) which have to be corrected after the final root cause identification by backtracking. In addition, the event generation from the providers own service surveillance can influence the result, e.g. if a lot of trivial events are generated which can be successfully correlated. A disregard of provider-generated events in this metric is also not desirable as the treatment of theses events is also important to the well-functioning of the fault management.

An indicator for the prevention of SLA violations is the mean time to identify the problems' root causes. To allow for a comparison before and after the installation of a service-oriented event correlation, a set of representative cases has be treated with and without the correlation. Besides the use of the mean time a percentage of root causes where the identification took longer than a predefined time interval, or the median of the identification times could be used.

5 Conclusion and Future Work

In this paper a framework has been presented to leverage the service-oriented event correlation and therefore benefit from the accelerated detection of root causes leading to a service degradation as well as the effort reduction for dealing with customer reports. It uses a combination of rule-based and case-based reasoning techniques which have been identified to be applicable for the service-oriented root cause analysis.

An application of the framework for large-scale services offered by the Leibniz Supercomputing Center to its customers in the Munich scientific community is currently carried out to measure the benefit of the approach.

The output of the service-oriented event correlation is a resource which has been detected to be the problem's root cause. Such a resource problem can be taken as input for an impact analysis [9] where the dependencies for services and resources are used to identify services affected by the resource problem. In addition, an SLA database allows for finding out affected customers and therefore to classify the importance of the problem resolution. The classification is useful to select an adequate measure to deal with the problem. Together with the impact analysis the work presented in this paper can be regarded as a framework covering the whole area of service fault management.

References

[1] P. Bhoj, S. Chutani, and S. Singhal. Sla management in federated environments. In *Proceedings of the 6th IFIP/IEEE International Symposium on Integrated Network Management (IM 1999)*, pages 293–308, Boston, Massassuchets, USA, May 1999. IFIP/IEEE.

[2] D. Caswell and S. Ramanathan. Using service models for management of internet services. In *HP Technical Report HPL-1999-43, HP Laboratories*, Palo Alto, California, USA, March 1999.

[3] C. Ensel. New approach for automated generation of service dependency models. In *Network Management as a Strategy for Evolution and Development; Second Latin American Network Operation and Management Symposium (LANOMS 2001)*, Belo Horizonte, Brazil, August 2001. IEEE.

[4] C. Forgy. Rete: A fast algorithm for the many pattern/many object pattern match problem. *Artifical Intelligence Journal*, 19(1):17–37, 1982.

[5] M. Garschhammer. Dienstguetebehandlung im Dienstlebenszyklus: von der formalen Spezifikation zur rechnergestuetzten Umsetzung - in German. PhD thesis, University of Munich, Department of Computer Science, August 2004.

[6] enhanced Telecom Operations Map (eTOM), The Business Process Framework for the Information and Communications Services Industry. Technical Report GB 921 Approved Version 3.0, TeleManagement Forum, June 2002.

[7] M. Gupta, A. Neogi, M. Agarwal, and G. Kar. Discovering dynamic dependencies in enterprise environments for problem determination. In *Proceedings of the 14th IFIP/IEEE Workshop on Distributed Sytems: Operations and Management*. IFIP/IEEE, October 2003.

[8] A. Hanemann, M. Sailer, and D. Schmitz. Assured service quality by improved fault management - service-oriented event correlation. In *Proceedings of the 2nd International Conference on Service-Oriented Computing (ICSOC04)*, New York City, New York, USA, November 2004. ACM.

[9] A. Hanemann, M. Sailer, and D. Schmitz. Towards a framework for failure impact analysis and recovery with respect to service level agreements. In *Proceedings of the 9th IFIP/IEEE International Conference on Integrated Network Management (IM 2005)*, Nice, France, May 2005.

[10] H.-G. Hegering, S. Abeck, and B. Neumair. *Integrated Management of Networked Systems - Concepts, Architectures and their Operational Application*. Morgan Kaufmann Publishers, 1999.

[11] G. Jakobson, J. Buford, and L. Lewis. Towards an architecture for reasoning about complex event-based dynamic situations. In *Proceedings of the Third International Workshop on Distributed Event Based Systems (DEBS 2004)*. IEE, May 2004.

[12] G. Jakobson and M. Weissman. Alarm correlation. *IEEE Network*, 7(6), November 1993.

[13] G. Jakobson and M. Weissman. Real-time telecommunication network management: Extending event correlation with temporal constraints. In *Proceedings of the Fourth IEEE/IFIP International Symposium on Integrated Network Management*, pages 290–301, Santa Barbara, California, USA, May 1995. IEEE/IFIP.

[14] S. Kliger, S. Yemini, Y. Yemini, D. Ohsie, and S. Stolfo. A coding approach to event correlation. In *Proceedings of the Fourth IFIP/IEEE International Symposium on Integrated Network Management*, pages 266–277, Santa Barbara, California, USA, May 1995. IFIP/IEEE.

[15] M. Langer, S. Loidl, and M. Nerb. Customer service management: A more transparent view to your subscribed services. In *Proceedings of the 9th IFIP/IEEE International Workshop on Distributed Systems: Operations & Management (DSOM 98)*, pages 195–206, Newark, DE, USA, October 1998. IFIP/IEEE.

[16] L. Lewis. A case-based reasoning approach for the resolution of faults in communication networks. In *Proceedings of the Third IFIP/IEEE International Symposium on Integrated Network Management*, pages 671–682, San Francisco, California, USA, April 1993. IFIP/IEEE.

[17] L. Lewis. *Service Level Management for Enterprise Networks*. Artech House, Inc., 1999.

[18] Office of Government Commerce (OGC), editor. *The Business Perspective*. IT Infrastructure Library (ITIL). The Stationary Office, Norwich, UK, 2003.

[19] S. Yemini, S. Kliger, E. Mozes, Y. Yemini, and D. Ohsie. High speed and robust event correlation. *IEEE Communiations Magazine*, 34(5), May 1996.