A Semantic Monitoring and Management Framework for End-to-end Services

John Keeney,
Owen Conlan

FAME & KDEG
School of Computer
Science & Statistics,
Trinity College Dublin,
Dublin, Ireland.

John.Keeney@cs.tcd.ie
Owen.Conlan@cs.tcd.ie

Viliam Holub,
Miao Wang

FAME & PEL
School of Computer
Science & Informatics,
University College Dublin,
Dublin, Ireland.

Viliam.Holub@ucd.ie
miao.wang@ucdconnect.ie

Laetitia Chapel

FAME & Hamilton Institute
National University
of Ireland,
Maynooth, Ireland.

Laetitia.Chapel@nuim.ie

Martín Serrano,
Sven van der Meer

FAME & TSSG
Waterford Institute
of Technology,
Waterford, Ireland.

jmserrano@tssg.org
vdmeer@tssg.org

Abstract— Modern distributed applications and communication services have become increasingly complex, composed of diverse heterogeneous sub-systems, and it is progressively more unrealistic that the users of these systems will be able to manage them in a holistic end-to-end manner. In particular, it is increasingly difficult to understand how such systems operate, the meaning of errors, and how they can be manipulated in a managed way, cognisant of the end-to-end nature of these systems. This paper describes an approach to semantically enrich monitoring information, events and faults, and management actions in such a way that they can be presented to a manager in manner that can be understood and leveraged. This work is based on one of the key scenarios FAME research project.

I. INTRODUCTION

Typical large enterprise and communications systems contain many physically distributed hardware and software components that communicate across different physical and overlay networks to satisfy client service requests. A key issue with such systems is the lack of structured approaches to monitor diverse and heterogeneous shared systems as a holistic homogeneous end-to-end service. This work aims to address three of the major challenges of managing modern end-to-end information and communication services: the inability to harmonise different management data from constantly evolving networks of different devices, systems and services in a cost-efficient and timely manner; the inability to express and fuse, in an interoperable way, end-to-end service, system and network information; and, the inability of (perhaps non-technical) service consumers and supplier-consumers to meaningfully participate in the quality of experience-based control loop of modern end-to-end services.

Monitoring data harmonisation requires mechanisms for mapping and federation of the large volume of low level data produced by monitoring systems into semantically meaningful information and knowledge that may be discretely consumed. Any complex system has domain experts who are very familiar with how the constituent parts of the system can be managed,

and are particularly aware of the end-to-end operating constraints of those constituent parts. Encoding this expertise in a manner that can be utilised by other stakeholders enables other stakeholders of other systems to monitor other parts of the system and relate their operation to their own system. It also enables common knowledge to be shared and reused across multiple domains. Personalised visualisation of harmonised monitoring data is viewed as key to including non-technical service consumers and provider-consumers in the end-to-end service quality of experience-based control loop [1]. Human interactions with networks are inherently complex; therefore, the governance of such complex systems must be a dynamic and converging two-way process. However, the relevance of particular network and/or management information or mechanisms is dependent on the user, network, and possibly environmental context, and hence must be properly contextualised [2] in a way that the consumer of those contextual data appreciates its relevance. This has led to the popularity of personalised “dashboard” type monitoring applications for visualising key performance indicators of managed systems [3]. The personalisation of visual state and context representations is most important where visualisations are used as a communication tool between different knowledge domains [4].

This paper describes aspects of the work in progress as part of the FAME project, which explores how monitoring at the network level can provide knowledge that enables enterprise management systems to reconfigure software components to better adapt applications to prevailing network conditions. This reconfiguration may, for example, involve redeployment of application components in different locations in order to alleviate congestion detected within particular parts of the network. In particular this paper discusses the gathering of end-to-end monitoring data for the network, the services and the servers; and the aggregation, enrichment, correlation, abstraction and visual presentation of this monitoring data to the multiple stakeholders involved in producing and consuming end-to-end enterprise services.

Figure 1: Monitoring and Management Framework

II. A FRAMEWORK FOR END-TO-END SERVICE MONITORING AND MANAGEMENT

This section describes a testbed monitoring framework established as a part of the FAME project, according to the architecture shown in Fig 1. In this testbed an end to end video streaming service was established, serving standard and high definition video files, with the initial contact with the service via a standard web-server.

A. Services

The testbed described here operates as a standard *Apache web server*¹ on a Linux server, which redirects RTSP video streaming requests to a *Darwin Streaming Server*² installed on another remote server. The statistics collecting utility, *collectd*³, continuously gathers server- and application-level statistics from both testbed machines. This information, combined with logging outputs were streamed into the monitoring framework.

B. Anomaly Detection

Another aspect of the work described here is the use of machine learning techniques to identify abnormal behaviour and probable errors, based on the analysis of the hardware- and network-level monitoring information. This work attempts to characterise the software operating on network devices (and some simpler models of the service providing servers) as linear systems. This work makes use of Kalman filtering to infer the parameters of the system by formulating the system as a ℓ_1 least squares, or LASSO, problem [5]. Under mild conditions, the algorithm yields consistent estimates and correctly identifies patterns in the inputs, while providing accurate

estimates of the parameters of the system. Depending on the number of parameters being estimated, and the diversity of the load on the system, this approach allows the parameter approximation mechanism to self-train in a short period of time. Once trained, this system can then detect anomalous behaviours in the networking equipment or server. A problem may present as unusual patterns in the low-level monitoring information, for example, previously unseen excessive CPU usage with very low throughput may indicate a software error in a network router, or unusual request string lengths or formats may indicate a security attack on an application server. Reports of such anomalies are then passed to the monitoring framework where they can be exploited to assist in correlating other monitoring readings and events about the entire end-to-end service being supported. It is important to note that the anomaly detection system by its nature cannot derive the cause of any anomaly, or if the anomaly is indeed an error, only that something unusual has happened. However, such reports can be very beneficial when correlated with other monitoring information or to highlight portions of logs.

C. Run-time Correlation Engine - RTCE

The Run-time Correlation Engine (RTCE) [6][7] is a framework that collects, correlates, and presents events produced by complex distributed software systems. Its main goal is to significantly speedup log analysis and assist in monitoring live applications. Correlation is based on a symptom database that provides a mechanism for matching known problems in large volumes of data. It supports capturing expert knowledge about known issues represented in a XML symptom database, which is then applied to the analysis of logged events. The framework presents a number of advantages for such tasks: 1) high performance event correlation (tens of thousands of events per second [7]); 2) low CPU and memory consumption [7]; 3) regular expression-based filtering that allows event categorisation and grouping, and filtering out

¹ http://projects.apache.org/projects/http_server.html

² <http://dss.macosforge.org/downloads/DarwinStreamingSrvr5.5.5-Linux.tar.gz>

³ <http://collectd.org/>

noise events; 4) aggregation of historical as well as live events; and 5) the ability to distribute the correlation tasks across a hierarchy of RTCE instances, to distribute the correlation load, and localise the network load.

D. INFLECTIONS & Semantic Attribute Reconciliation Architecture – SARA

Ideally the monitoring user would prefer complex monitoring information to be presented in a way they can understand and explore. SARA [8][9] provides mechanisms for semantically lifting, reconciling and mapping heterogeneous information sources into a common semantic structure that is personalised to the user. SARA allows domain experts (even those with no computer coding experience) to encode domain concepts and their accompanying rules, and provides a means to apply expert oriented subjective analysis and trends, in the form of semantic attributes. Semantic attributes [1] are discrete encodings of domain expertise, abstractions and simplifications that can be combined together and personalised to support user exploration of an information domain. This enables a novice user to access and explore an information space in accordance with one or several experts' perspectives [8]. SARA provides an authoring tool to enable the domain experts to create and combine semantic attributes. One of the key successes of SARA is that these semantic attributes may be further personalised to the user's specific needs by adjusting their parameters.

INFLECTIONS is a framework that lies between SARA and client visualisation tools to support exploration of semantically enhanced information. It supports prioritisation of semantic and numerical information into quantised views, thereby providing layered access to the information. The labelling of the layers also introduces further meaningful semantics: e.g., all network traffic below a given speed may be grouped into layers named 'Slow'. Several layers may be selected by the client, thus showing a depth of information, e.g. all 'Slow' and some 'Medium' layers. A key benefit of INFLECTIONS is the ability to manage this depth in order to support exploration and manage the end user's attention, thereby mitigating cognitive overload. INFLECTIONS also has an authoring environment to enable a domain expert define reusable views of the information with sequenced filtering rules based on the values of the data and associated semantic attributes from SARA. This enables the domain expert to highlight trends and features of the information that may be pertinent to the end user.

E. Semantic-based Service Control Engine - 2SCE

The Semantic-Based Service Control Engine (2SCE) represents an intermediate solution for efficient management, in a semantic-driven way, of application and network devices based on semantically encoded monitoring information and events. Events are considered as triggers for new service actions so that underlying infrastructure and platforms can modify their performance according to agreed or defined operation requirements.

Fig. 2 illustrates the operational components of 2SCE. The technology independent components contain a set of functional decision objects directly involved in the application of end-to-

Figure 2: Semantic-based Service Control Engine - 2SCE

end decisions. The technology dependent components represent the functional enforcement objects necessary to interact with external systems and frameworks from which information is extracted and upon which actions are enforced.

The Event Conflict Check Components handle the identification of events and potential conflicts and validate the format and identifying event features in the event message. Once each event is checked, and its semantics and context is classified it is passed to the Objects Consumer Manager, which interacts with the object manager modules. These include: the Repository Access Components, where Operation and Configuration Rules and Event identifiers are stored; and the Authentication Check Components, where the integrity of the information model and data model are verified.

Once triggered by a trigger event via the Event Conflict Check Components, rule conditions referring to monitored objects are retrieved by the Decision making Objects Manager. If successfully evaluated the Object Consumer manager then delegates the performance of a (possibly composite) action to the relevant Action Consumer. Actions Consumers compose and forward any policy enforcement message to the appropriate managed object.

In this work 2SCE is driven from both high-level semantically enhanced information taken from SARA and lower-level correlated information drawn from RTCE, both in terms of trigger events and information to be used in policy condition clauses. The set of actions available to 2SCE for use in its policy scripts is defined as semantically encoded higher-level composite actions. These actions will be drawn from the action sets of the underlying end-to-end services in a manner whereby they can be semi-automatically decomposed to enforceable management actions, as described in [10][11].

To test the reliability of the 2SCE system a set of policies for application and service management have been generated and loaded by both 2SCE and a comparable Policy-based Management System (PBSM [12][13]). As shown in fig 3 and 4 below, for the same policy set, an initial prototype implementation of 2SCE demonstrates reduced the CPU usage, similar memory usage, and although not shown here, reduced

time requirements. Fig 3 shows CPU usage percentage vs. the number of operations and rules (service policies) that has been created and processed by PBSM and 2SCE. Fig. 4 shows memory usage in megabytes vs. the number of operations and rules when an application or service is using the loaded rules.

Figure 3. CPU Usage Comparison PBSM Vs 2SCE.

Figure 4. Memory Usage Comparison PBSM Vs 2SCE.

(Tests run on laptop with an Intel Core Duo Processor @ 2.20GHz AT/At compatible; 2GB RAM; Microsoft Windows XP 5.00.2195 SP4; JRE 1.5.0_06 from Sun Microsystems; AppPerfect DevSuite 5.0.1 - AppPerfect Java Profiler)

III. CONCLUSIONS

In this work we focus on the issues involved in simplifying the end-to-end monitoring and management of complex networking and services. In particular we concentrate on how monitoring information and management tasks can be gathered, correlated and described in an abstracted manner so that the person overseeing or managing the entire process need not be an expert in all high-level and low-level aspects of its operation. This is particularly important as the end-to-end service, and the systems upon which it depends grows bigger, more complex, and more difficult to manage.

One of the main advantages of such a framework is that it provides the ability to harmonise data from different devices and services to support a holistic end-to-end view of the service. It also supports on-the-fly monitoring and manipulation of the service and its constituent parts in a user-

friendly and usable manner. This approach also enables diverse expensive expert knowledge to be codified and reused in a cost effective and flexible manner.

ACKNOWLEDGEMENT

This material is based upon works supported by the Science Foundation Ireland (Grant No. 08/SRC/I1403) as part of the Federated, Autonomic End to End Communications Services Strategic Research Cluster (www.fame.ie).

REFERENCES

- [1] Hampson, C.: "Semantically Holistic and Personalized Views Across Heterogeneous Information Sources", in proceedings of the Workshop on Semantic Media Adaptation and Personalization, (SMAP07), London, UK, December 17-18, 2007
- [2] Novak, J.: "Helping Knowledge Cross Boundaries: Using Knowledge Visualization to Support Cross-Community Sensemaking", in proceedings of the Conference on System Sciences, (HICSS-40), Hawaii, January 2007
- [3] Palpanas, T., Chowdhary, P., Mihaila, G.A., Pinel, F.: "Integrated model-driven dashboard development", in the Journal of Information Systems Frontiers, vol 9, no. 2-3, Jul 2007
- [4] Burkhard, R. A.: "Learning from Architects: The Difference between Knowledge Visualization and Information Visualization", in proceedings of the Conference on Information Visualisation, (IV 2004), London, UK, July 2004
- [5] Chapel, L., Leith, D.: "Sparse Input Matrix and State Estimation for Linear Systems", to appear in proceedings of the 49th IEEE Conference on Decision and Control (CDC), Atlanta, GA, USA, Dec 15-17, 2010
- [6] Holub, V., Parsons, T., O'Sullivan, P., Murphy J.: "Run-time correlation engine for system monitoring and testing", in proceedings of the 6th International Conference on Autonomic Computing, (ICAC), Barcelona, Spain, Jun 2009.
- [7] Wang, M., Holub, V., Parsons, T., Murphy, J., O'Sullivan, P.: "Scalable run-time correlation engine for monitoring in a cloud computing environment", in proceedings of the 17th IEEE International Conference and Workshops on Engineering of Computer-Based Systems, (ECBS), Oxford, UK, March 2010
- [8] Hampson, C., Conlan, O.: "Supporting Personalised Information Exploration through Subjective Expert-created Semantic Attributes", in proceedings of the IEEE International Conference on Semantic Computing (ICSC), Berkeley, CA, USA, September 2009
- [9] Conlan, O., Keeney, J., Hampson, C., Williams, P.: "Towards Non-expert Users Monitoring Networks and Services through Semantically Enhanced Visualizations" to appear in proceedings of the 6th International Conference on Network and Service Management (CNSM 2010) (formerly MANWEEK), Niagara Falls, Canada, 25-29 October 2010
- [10] Keeney, J., Conlan, O., O'Sullivan, D., Lewis, D., Wade, V.: "Towards the Visualisation of Collaborative Policy Decomposition", in proceedings of the 9th IEEE Workshop on Policies for Distributed Systems and Networks (POLICY 2008), Palisades, NY, USA, 2-4 June 2008
- [11] Keeney, J., Lewis, D., Wade, V.: "Towards the use of Policy Decomposition for Composite Service Management by Non-expert Endusers" in proceedings of the 5th IEEE/IFIP International Workshop on Business-driven IT Management (BDIM 2010) at NOMS 2010, Osaka, Japan, 19 April 2010
- [12] Serrano, J.M., Serrat, J., Strassner, J., Ó Foghlú, M.: "Facilitating Autonomic Management for Service Provisioning using Ontology-Based Functions & Semantic Control", in proceedings of the 3rd IEEE International Workshop on Broadband Convergence Networks (BcN 2008) at NOMS 2008, Salvador de Bahia, Brazil, 7 April 2008
- [13] Serrano, J.M., Justo, J., Marin, R., Serrat-Fernandez, J., Vardalochos, N., Jean, K., and Galis, A.: "Framework for managing context-aware multimedia services in pervasive environments". International Journal of Internet Protocol Technology (IJIPT), vol. 2, no. 1, January 2007