From global measurements to local management

D 3.1: Design Space Analysis for Operational Support Tools

Date: 21st January, 2014

Version 2

Dissemination level: Public

The project is co-funded by Leone partners and by the European Commission in the 7th Framework Programme

ABSTRACT

This document reports the Leone project's work on the analysis, design, realisation and integration of operational support tools, which leverage measurements from large-scale distributed infrastructures.

Progress includes new visualisation and automated repair techniques, design and realisation of prototypical tools, and extensions of current management systems.

TARGET AUDIENCE

This document is targeted at a technical audience with some basic knowledge of routing and network monitoring, measurement and management.

ABOUT THE PROJECT

Project Coordinator: Philip Eardley, BT

Technical Manager: Sam Crawford, SamKnows

Start date of project: November 1st 2012

Duration: 30 months

Acknowledgement: Leone is a Collaborative Project under FP7 ICT-2011.1.1 Future Networks. The research leading to these results has received funding from the European Union Seventh Framework Programme [FP7/2007-2013] under Grant Agreement n° 317647.

ABOUT THE DOCUMENT

Editor: Stefano Vissicchio (UCL)

Due date of original Deliverable: 31st September, 2013

Original submission date: 16th September, 2013

Version 2 submitted on: 21st January, 2014

LIST OF AUTHORS

Authors (in alphabetical order):

Vaibhav Bajpai (Jacobs University Bremen)

Boris Banjanin (MG-Soft)

Andra Lutu (UC3M)

Maurizio Pizzonia (ROMA3)

Massimo Rimondini (ROMA3)

Jürgen Schönwälder (Jacobs University Bremen)

Claudio Squarcella (ROMA3)

Stefano Vissicchio (UCL)

EXECUTIVE SUMMARY

One of the main objectives of the Leone project is to provide network or service operators with tools to maximise the utility of the information collected through large-scale measurement platforms. Those tools are expected to help operators speed up troubleshooting, to detect performance problems, and then repairing them, while reducing management costs.

This document reports on the progress of the Leone consortium regarding the design, implementation and integration of those tools. Our contributions include visualisation prototype tools, extensions of management systems with additional abilities, such as anomaly detection and alarm triggering, and automated repair techniques.

The first contributions are techniques to effectively visualise traffic flows and routing information as reported by various measurement data sources. To improve the effectiveness of the visualisation, we studied the possibilities to convey graphically enriched representations of the paths on which traffic currently flows. Namely, we devised techniques, based on state-of-the-art graph drawing know-how, to simultaneously visualise traffic paths and their respective measured performance, e.g., delay. Moreover, we proposed algorithms to represent traffic paths at different levels of aggregation, and in combination with geographical information. For each technique, we realised a working prototype tool.

The visualisation tools facilitate the elaboration of effective strategies to fix performance problems and to take longer-term higher-level decisions, e.g., on commercial relationships with their respective customers and providers. Once those decisions have been taken, the configuration of network devices has to be changed accordingly. Similarly, network behaviour has to be modified to react promptly to major performance problems. We investigated algorithms and methodologies to support network operators to reconfigure devices. In particular, we studied new automatic and practical approaches to reconfigure network devices while ensuring network service continuity, overcoming the identified limitations of the state-of-the-art. We proved formally the correctness of our algorithms.

We envision visualisation and automatic repair techniques, as well as the results of measurement data analyses, to be integrated in current management systems. We analysed the possibilities for such an integration. On one hand, we evaluated the suitability of existing protocols, such as IPFIX and NETCONF for the integration with generic management systems. On the other hand, we have already developed extensions of a specific management system, i.e., the MG-Soft Net Inspector software. These extensions enable the import, conversion and elaboration of measurement results obtained by probes of the SamKnows measurement infrastructure using well-known data formats (CSV) and protocols (SFTP).

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

AS	Autonomous System
BGP	Border Gateway Protocol
IGP	Interior Gateway Protocol
CSV	Comma Separated Values
DSL	Digital Subscriber Line
FTP	File Transfer Protocol
HVP	High Visibility Prefix
ID	Identifier
IE	Information Element
IPFIX	Internet Protocol Flow Information Export
JSON	JavaScript Object Notation
KPIs	Key Performance Indicators
LMAP	Large-Scale Measurement Platform
LVP	Low Visibility Prefix
MA	Measurement Agent
NETCONF	Network Configuration Protocol
NMS	Network Management System
NOC	Network Operations Centre
QoE	Quality of Experience
SFTP	Secure File Transfer Protocol
SITN	Ships-in-the-Night
SQL	Structured Query Language
XML	Extensible Markup Language

1 INTRODUCTION

The goal of the work performed within work package 3 of the Leone project is to devise tools to support operators in effectively exploiting measurement data from probes distributed throughout an operator's network, for example embedded in every home gateway. The particular challenges are the large-scale of the measurement data and the richness of the information provided, compared with the traditional data managed by a network operator.

We have been working on several approaches:

- Visualisation techniques, which could help a network operator identify faults and issues that would otherwise be hard to identify amongst the large amount of collected data
- Extending an existing network management tool, MG-Soft's Net Inspector, with the measurement data and automatically identifying anomalies
- Techniques to smoothly alter intra-domain routing in order to repair around faults identified using the techniques in the two bullets above.

Together these build into an integrated framework for Quality of Experience (QoE)-oriented troubleshooting.

The following scenario illustrates our approach. Suppose a network operator measures one of the QoE metrics developed by Leone (and described in Deliverable D2.1), for example the one-way delay metric using PNPM ping, by using probes (also called Measurement Agents, MAs) located at the customer premises. The collected data feeds into the network operator's existing Network Management System (NMS), where an algorithm detects an anomaly and triggers an alert via the same means as the operator already uses. The operator takes advantage of the Leone visualisation tools to investigate the problem; the visualisation integrates other data such as BGP routing data, traceroute, and other Leone QoE-related metrics. The operator is alerted to an increase of the one-way delay and, by using the visualisation to explore how performance and routing has evolved, the operator can identify that the problem is caused by one of its providers reconfiguring its routing. The operator can contact the provider, but rather than simply waiting for the problem to be fixed, it can also apply a temporary workaround by reconfiguring its own network so that a different egress path is taken to avoid the under-performing ISP. Today, such a reconfiguration is usually considered too risky but Leone's automatic repair methodology will allow a seamless transition to the new routing configuration.

Generalising the example above into a methodology, we envision that the operator uses its network management system to analyse data coming from the measurement network, detects a measurement that does not conform to the expected QoE (metrics are described in Deliverable D2.1), devises a reconfiguration that will fix the problem, and automatically applies it. The whole process is described in the following diagram.

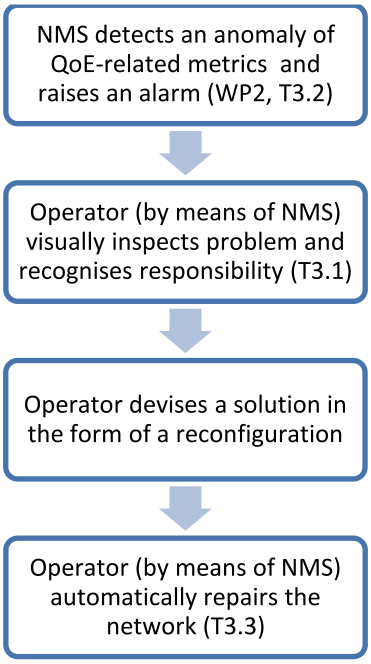

Figure 1: How to combine the various operational tools

In Section 2 we describe our work on visualisation. We have worked on three techniques and developed a prototype tool for each:

1. Simultaneously visualise routing paths and the corresponding performance measurements
2. Visualise paths at different level of abstractions, and
3. Simultaneously visualise routing and geographical information

Each involves routing plus performance information, such as Leone's QoE metrics (described in Deliverable D2.1).

In Section 3 we describe our work on network management integration. We have studied the integration of Leone's measurement data into one existing commercial network management tool, namely Net Inspector. We believe this is the first integration of large-scale QoE measurement data into a real NMS (see Section 3.2) where an operator can apply the typical tools that are used to manage its local network (charts, thresholds, alarms, logging, etc.). In Section 3.1 we studied the suitability of the existing protocols NETCONF and IPFIX for the control of the probes (MAs) and the collection of the measurement results. As explained in Section 3.1, we are now instead developing an HTTP-based protocol; that work is described in Deliverable D1.1.

In Section 4 we study "automated repair". What repair is appropriate is highly dependent on the specific performance problem, the deployed technologies and the constraints (such as the possibility to use given portions of the network as a backup, the kind of configuration changes allowed outside the management windows, etc.). We studied one type of repair – changing the intra-domain routing – and how to automatically deploy new routing configurations whilst avoiding transient loops, packet losses and service disruptions.

In our on-going work, we intend to test the developed prototypes within the Leone trial (described in Deliverable D4.2) and get feedback from the operators involved in Leone, in order to improve our research.

2 VISUALISATION

Network intermediate systems, such as routers, act at two levels:

- On the routing level, they agree on the paths to be used for user packets, and
- On the forwarding level, they forward packets to their respective next-hop.

Routing paths can have a significant influence on the forwarding performance, hence on the Quality of Experience (QoE) of users.

State-of-the-art network visualisation provides tools that are based on simple visualisation techniques (based on forces or on a tree layout). They cannot show large-scale measurement data that is equipped with geographic location and evolves over time. Furthermore, no visual association with an administrative authority, that can be identified by means of the Autonomous System or the BGP protocol, is usually provided.

We considered techniques for an effective visualisation of routing and forwarding paths, which aim to help network operators assess the impact of computed routing paths on QoE. To this end, we explored three techniques to:

1. Simultaneously visualise routing paths and the corresponding performance measurements
2. Visualise paths at different level of abstractions, and
3. Simultaneously visualise routing and geographical information

The three techniques are detailed in the following subsections, together with their application contexts. The techniques are based on state-of-the-art graph-drawing methods, such as interactive clustering and mixed geographical/abstract views. We advance the state-of-the-art of network visualisation, as the devised techniques are the first to adopt sophisticated graph-drawing algorithms to visualise measurements obtained from large-scale measurement infrastructures.

To date we have visualised public data (e.g. data coming from RIPE Atlas or BGP routing data). The integration will be performed in the second period of the project. This was a pragmatic decision in order to test our algorithms before data was available from the Leone trial (work package 4). We now plan to integrate TPlay (described in Section 2.2) with data coming from the trial and to make it available within the network management software Net Inspector (see Section 3.2). Thus we will visualise some of the Leone metrics developed within work package 2.

2.1 Combined visualisation of routing paths and network measurements

While routing and forwarding information can be analysed independently, a combined visualisation of data from both planes can offer the optimal support to network operators for operational decisions. However, the interplay between the routing and the forwarding plane is difficult to be rendered in a visually understandable and user-friendly form. In addition, routing and forwarding data are typically collected from physically different vantage points, which collect information in different formats and on different time scales.

To overcome these difficulties, we leveraged data aggregation and clustering techniques. We devised a graphical metaphor which we implemented in a visualisation tool called “Hydra”. The metaphor is conceived to convey the round-trip delay measured by geo-located probes (like SamKnows probes) and the inter-domain routing paths gathered by BGP collectors (like the RIPE RIS ones) in a combined graphical representation. A snapshot taken from the “Hydra” tool is shown in Figure 2.

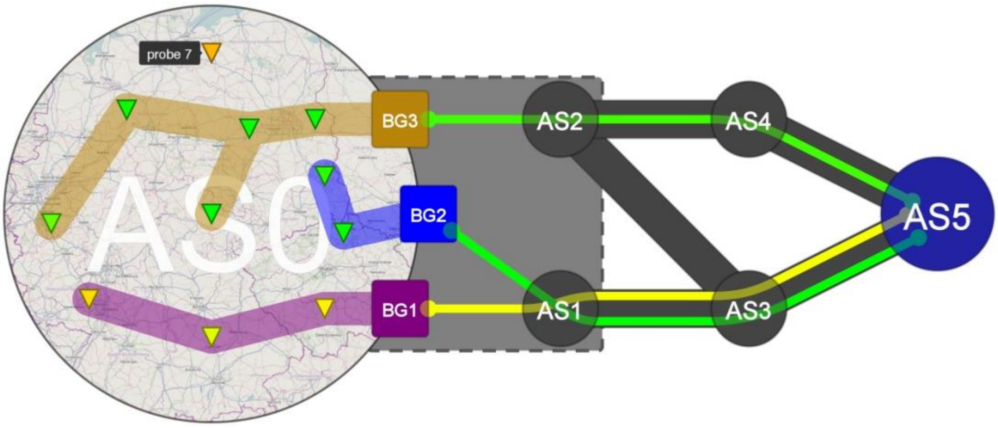

Figure 2: Example visualisation with the Hydra tool

Our metaphor highlights both the topology at the Autonomous System level (right part of the drawing) and the latency associated with each AS-path (left part of the drawing, which shows a detailed view of AS0). Furthermore, it shows how probes are partitioned into clusters associated with each Border Gateway, based on observed patterns in the measurement results. The visualisation also allows the user to explore the dynamics of the correlation between the two types of information with an intuitive animation. Note that this visualisation approach can be exploited with any end-to-end metric that results in a numeric quantity, like most of those developed within work package 2.

Additional references for this work are reported in Appendix A.

2.2 Visualising forwarding paths at different levels of aggregation

To support the in-depth analyses of routing and forwarding paths, we explored the possibility to visualise information at different levels of abstraction. We especially focused on traceroute measurements, that track routing paths at the router level. Currently, however, the exploitability of such a rich source of data is impaired by the lack of a proper abstraction and the presence of incomplete pieces of information.

To overcome this obstacle, we designed a visualisation tool, which we called “TPlay”. TPlay exploits clustering visualisation techniques to allow its users to look at traceroute measurements at different levels of abstraction. A snapshot of the tool is depicted in Figure 3.

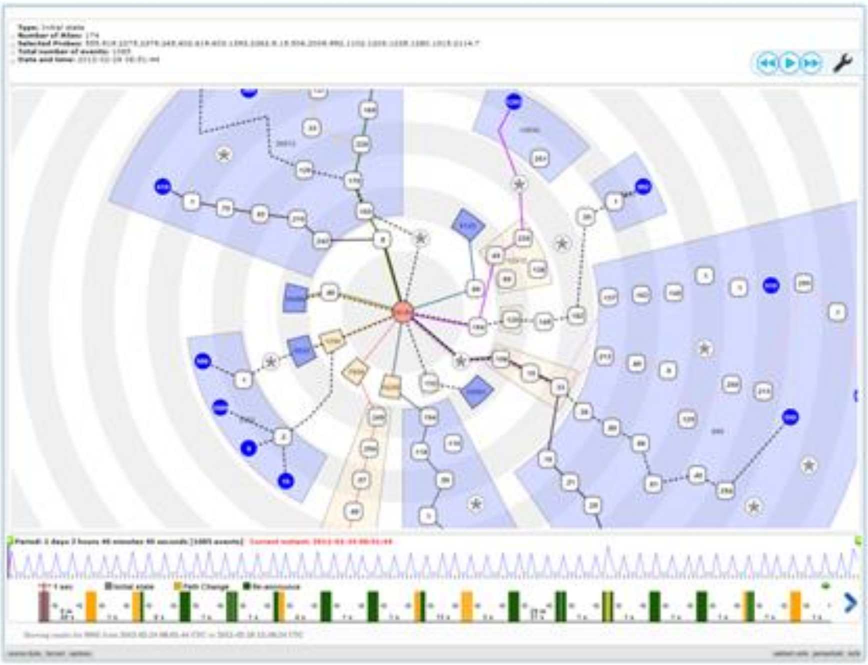

Figure 3: Example visualisation with TPlay

In the TPlay tool, by default, hops in forwarding paths are grouped according to the AS to which they belong, and are not explicitly displayed. The user can then arbitrarily choose to expand an AS and look at the detailed traceroute paths traversing it. The tool performs a node expansion with a smooth and intuitive animation. The layout of the drawing is radial, with the target AS of the traceroute being represented in the centre. This enables easy comparison between the paths from different observation points (e.g., measurement probes) and offers an intuitive representation of topological distances. Furthermore, the visualisation offers the possibility to observe the evolution of the paths over time using easy-to-understand animations.

At the moment, TPlay shows only topology derived from traceroutes and evolution over time. We plan to equip the tool with the capability to show (for each probe) the round-trip time associated with traceroutes (for each hop) and one other end-to-end numerical metric (for instance most of those developed within work package 2 would be suitable).

We designed an algorithm for the TPlay system that can be used to draw generic sparse graphs. This was motivated by a study we made on data coming from RIPE Atlas (similar in structure to that which will be produced by Leone) that shows how the union of a number of traceroutes gives graphs whose density (number of edges divided by the number of nodes) is usually quite low.

The TPlay algorithm creates a drawing whose nodes are logically assigned to rectilinear levels in which the probes are on the upper levels, and the target is in the bottom layer. The drawing presented to the user is a circular distortion of this output so that the target is in the middle of the screen, in order to give a more user-friendly view. To support the grouping nodes by the Autonomous System to which they belong, it produces drawings in which nodes belonging to the same group are close each other (see Figure 4).

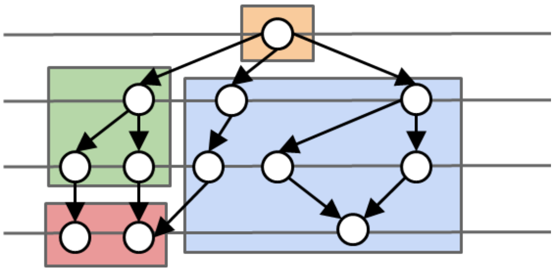

Figure 4: Nodes belonging to the same group are laid out close each other

The algorithm strives to produce a planar drawing if possible. To do this, it applies a sophisticated planarity technique relying on the PQ-tree data structure (Booth and Lueker, 1976) that implicitly represents all possible permutations of a set of elements under contiguity constraints that can be incrementally specified.

Appendix A contains additional references for both the tool and the techniques used in it.

2.3 Concurrent visualisation of topology and geography

We also explored techniques to reconcile the representation of topological and geographical information. Such a reconciliation presents a peculiar graph-drawing problem. Indeed, user-friendly representations of topological information can be effectively achieved through algorithms that arbitrarily arrange nodes in the available drawing space. On the contrary, geographical information is bound to the physical location of the represented entities, and can therefore lead to less readable drawings (e.g., with many nodes concentrated in close locations and others placed far away).

Within our study, we defined a visualisation methodology that is general enough to be independent of the specific semantics of the data being visualised, and that can be effectively applied to analyse routing information. The methodology is based on an innovative 2.5D representation consisting of two flat surfaces layered in a 3D space. The lower layer, the *geographical layer*, displays the actual location of the visualised entities (e.g., routers, hosts, etc.) and their connections, superimposed on a geographical map. The higher layer, the *logical layer*, draws the same information using a graph drawing algorithm that produces a nicer and more understandable representation. Lines between the two layers help match the entities visualised on the two layers. To represent possible uncertainties in the geographical locations, a cone can be projected from an entity on the logical layer to an area on the geographical layer. The visualisation can be dragged to centre the layers on a particular area, and each entity can be easily selected. An example of the application of this technique, described more extensively in Appendix B, is presented in Figure 5.

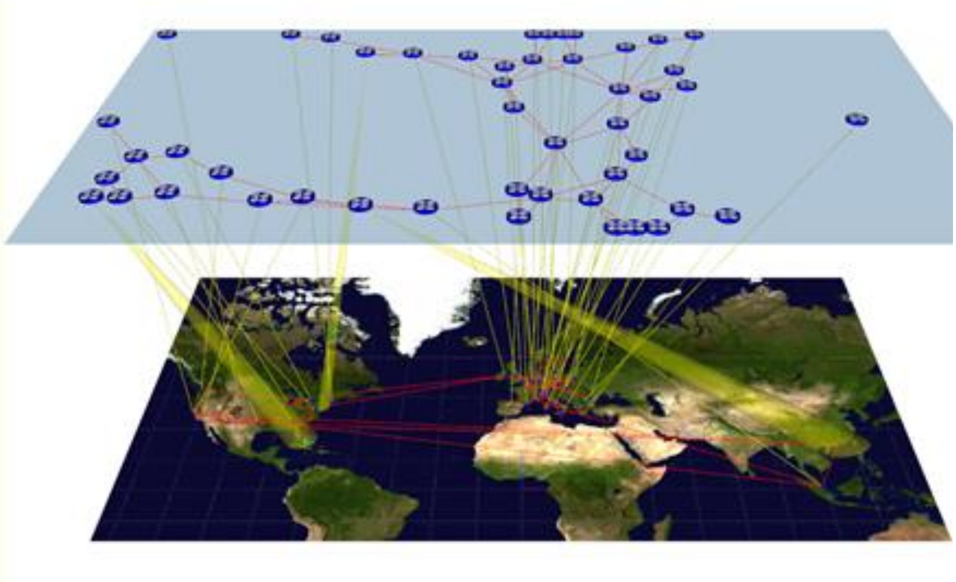

Figure 5: Example of concurrent topological and geographical visualisation

3 NETWORK MANAGEMENT INTEGRATION

In the following, we describe our progress towards integrating the measurement infrastructure with standard management systems (Section 3.1) and with the MG-Soft Net Inspector software (Section 3.2).

The Leone architecture, as described in Deliverable D1.1, includes the key components of Controller, Collector and Measurement Agent (MA). The question is what protocols should be used to transport the instruction from the Controller to the MAs, and to send the measurement results from the MAs to the Collector. Section 3.1 describes our study of the suitability of two existing protocols, NETCONF and IPFIX, which are used by several management systems today.

It is possible that large-scale measurement systems will be deployed as a new tool, completely independent of all a network operator's existing tools for test and diagnostics, and for design and planning. However, it would seem much better if, instead, large-scale measurements supplemented existing tools and helped them be more accurate, powerful and better able to help a network operator improve QoE. Rather than a generic study of this issue, instead we are taking an exemplar approach and working on the practical integration of our measurement infrastructure, as being developed for our trial (see Deliverable D4.2) with Net Inspector, which is an existing commercial network management system from MG-Soft (one of Leone's partners). Section 3.2 discusses our progress.

3.1 Integration with standard management systems

In this section we explore the integration of the Measurement Agents (MAs) with standard management systems. In particular, we explored the integration via two standard protocols, namely NETCONF and IPFIX. NETCONF is a generic network configuration protocol standardised by the IETF while IPFIX is a generic data pushing protocol, also standardised by the IETF. There are several implementations of these protocols available and hence it appears attractive to investigate whether these protocols can satisfy the Leone requirements, or if any specific features are missing. Other network management protocols defined by the IETF, such as the Simple Network Management Protocol (SNMP) were not considered because SNMP is often deployed in read-only mode and blocked by firewall configurations. NETCONF also provides more complex operations to manipulate the configuration of a device including support for transactions.

We studied the suitability of the Network Configuration Protocol (NETCONF) as an interface between management systems and MAs of a large measurement infrastructure. NETCONF provides mechanisms to install, manipulate, and delete the configuration of network devices. The IETF is considering NETCONF as one of the candidate protocols that can be used to configure the MAs in a large-scale measurement platform (LMAP). This would require that a MA runs a NETCONF server, while the LMAP Controller runs a NETCONF client.

We explored the technical challenges of using NETCONF. They encompass

- Connection Initiation. Typically MAs are in the home network which is behind a firewall. Hence, MAs need to initiate a connection to the Controller. This requires a standardised call-home mechanism, which is currently not supported in NETCONF, although work is under way.

- Definition of Server/Client Roles. Contemporary large-scale measurement platforms use proprietary protocols, where the Controller is a server, while the MAs are clients. This conflicts with the NETCONF LMAP model. The model may also require optimisations in situations where configuration updates need to be delivered.
- Pushing measurement results. NETCONF is not designed as a data-push protocol. Event notifications can be moulded to provide the capability, but the approach is not scalable for LMAP.

YANG is the data modelling language used to specify NETCONF models and protocol operations. We started to evaluate the suitability of YANG as the data model to configure and schedule measurements within a LMAP, and to work on better alternatives to it. More detailed references to those considerations are reported in Appendix A.

In addition, we analysed the possibility of using IPFIX as an interface between the MAs and the management system. IPFIX is a unidirectional, transport-independent export protocol for binary data records, with a focus on network measurement and operations applications. The structure of the data records is described in-band by Templates, which refer to Information Elements (IEs) from a common data model managed by the Internet Assigned Numbers Authority (IANA). The basic IEs cover most Layer 3 and Layer 4 measurement needs, and the Information Model can be extended.

IPFIX organises data records into Messages. A Message is a sequence of Sets preceded by a Message Header which, among other things, includes an Observation Domain ID (identifying where the records in the Message were measured) and an Export Time (when the Message was originally sent).

A Set contains Records preceded by a Set Header, which contains a Set ID identifying the type of Records the Set contains. Template Sets, identified by a special Set ID, contain Templates, which are sequences of IE identifiers and lengths; these define the fields of the Records they describe. A Template's ID matches the Set ID of the Sets containing records described by the Template. Since many records may be described by a single Template, IPFIX’s data representation is more efficient than those based on inline record structures (e.g., XML, JSON).

In order to use IPFIX to report results from the MA, we would need an IPFIX Template.

Part of the information can be conveyed using the fields in the IPFIX header, namely:

- Information about the MA: The MA identifier can be sent in the Observation Domain field of the IPFIX header
- Information about the time of the report: The Export Time field that can be used to convey this information

The information describing the test is included in a Template Set that contains multiple IEs for each of the different pieces of information we need to convey. This includes:

- An identifier of the metric used for the test. In order to convey this we need to define a new IE, for example: *metricIdentifier*
- An identifier of the scheduling strategy used to perform the test. Again, this will be a new IE, for example: *testSchedule*
- An identifier of the output format, for example: *outputType* is needed

- An identifier of the environment, notably, whether cross traffic was present during the execution of the test. A new IE is needed for this, for example: *testEnvironment*
- The input parameters for the test. Most of these can be expressed using existing IEs, such as *sourceIPv4Address*, *destinationIPv4Address*, etc.

The information describing the test results varies widely with each test, but can include the time each packet was sent and received, the number of sent and lost packets, or other information. Again, most of these can be expressed using existing IEs, and some new ones can be defined if needed for a particular test. More details about this can be found in Appendix A.

The results of our investigations were presented to the IETF at the 86th IETF meeting (March 2013).

We concluded that while NETCONF and IPFIX can be made to work, they both require a number of protocol extensions. On the other hand, it seems that none of the existing large-scale measurement platforms natively supports NETCONF and IPFIX. Furthermore, home routers running embedded versions of Linux (e.g., OpenWrt) are prime targets for hosting measurement agents and they usually do not support NETCONF and IPFIX but they have rich support for HTTP (both clients and servers). In addition, RESTful protocols running over HTTP are known to integrate very easily into other infrastructures. As such, it was decided to investigate how a RESTful protocol could address the Leone requirements. It was also decided to develop a protocol-neutral Information Model, as standardising this would be valuable even if different measurement systems used different protocols. Further details can be found in Deliverable D1.1.

3.2 Integration with MG-Soft Net Inspector

The impact of large-scale measurements can be hindered by the complexity of accessing data. In this project, we started one of the first experiments of integration of such measurements into a commercial network management system, namely Net Inspector. We are tackling three challenges:

1. How to import measurement results in the format used by the Leone infrastructure and translate them into the data model used in the Net Inspector application
2. How to automatically discover Leone probes (MAs) from measurement results, and
3. How to extend the existing anomaly detection capabilities of Net Inspector, in order to best detect anomalies in the Leone measurement results

To date we have made good progress on the first two areas and have a design for the third. We are performing extensive verifications using real measurement results.

The next step, as well as completing this work, is to study how Net Inspector and TPlay can complement each other. One idea is that a Network Manager would start in Net Inspector, select a set of managed devices/probes (perhaps those detected as having an anomaly), and then switch to the TPlay view, which would visualise the routing, forwarding and performance metric for those specific devices/probes and with all the correct context automatically loaded. This approach is viable since both applications share the same source data model (measurement probes and their identification numbers, target measurement servers, etc.). Such integration will provide a more complete image of the managed system.

We now provide more details on some critical points in our extension.

3.2.1 Data import and translation

The results of measurements performed by the Leone probes are exported as Comma Separated Values (CSV) files. Since each measurement can consist of one or more metrics, the format of the exported CSV files varies from measurement to measurement. Hence, the import procedure needs to be configurable.

To this end, we extended the Net Inspector software to support *data import models***.** Data import models define the interpretation of fields in CSV files and their translation to the Net Inspector Performance Manager data model. Namely, each data import model:

- Refers to a given measurement type
- Defines which fields in the CSV file reporting measurement results identify the references to the probes generating the results (probe IDs), the measurement instance, and the timestamps
- Defines metric-dependent descriptions, such as name, description, data type and unit
- Specifies the KPI to be registered in the MG-Soft Net Inspector Performance Manager data model. Typically, indeed, a new KPI is registered for each measurement metric

In our implementation, data import models are written using XML notation. The current measurement files are transferred from Leone's results collector to the Net Inspector application using one of the standard file transfer protocols, such as FTP or SFTP. After translation, the results are stored in Net Inspector's bundled SQL database, and can be visualised by using all the visualisation capabilities built into Net Inspector.

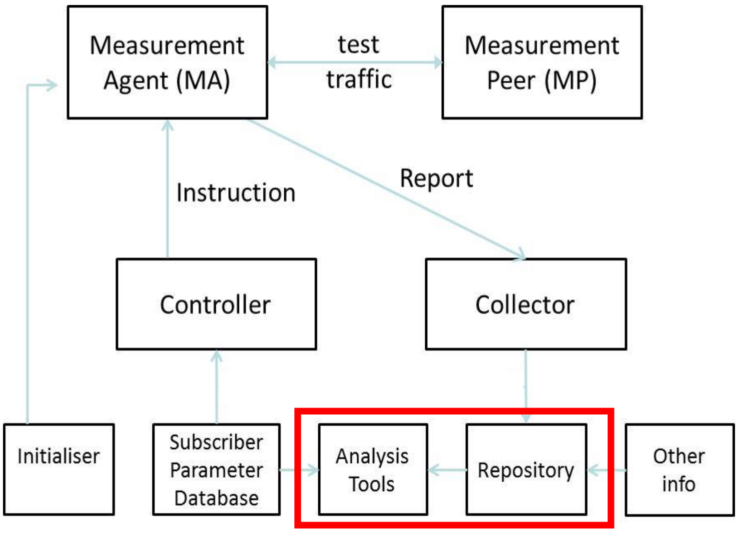

Figure 6: The Leone architecture (annotated)

Referring to the Leone architecture presented above, the integration conducted so far fits into the areas highlighted in Figure 6. In our implementation, the Repository is effectively the database of measurements received from the MAs. The arrow pointing towards the Analysis Tools is the transfer of compressed CSV files over a well-known protocol (SFTP). The Analysis Tools node is the Net Inspector software.

3.2.2 Automatic discovery of SamKnows probes

To better integrate the MG-Soft software with (possibly multiple) measurement infrastructures, we realised the support for the automatic discovery of measurement probes. In particular, probes are discovered when measurement results are imported into Net Inspector.

Two critical tasks are performed during automatic probe discovery, namely:

- Definition of the scope of the probe ID. Information is added about which measurement infrastructure a probe belongs to. An extended ID introduces the capability to import measurement results from several different measurement platforms
- Automatic creation in Net Inspector of a managed object for each probe. This avoids manual configuration

3.2.3 Anomaly detection support

By ‘anomaly’ we mean a measurement result that falls outside a valid range, as defined by thresholds. Since manually configuring thresholds is typically impractical (due to the large number of probes), we studied techniques to automatically compute thresholds.

We especially focused on *threshold calculation* and *event generation*.

Currently, our implementation supports three threshold computation methods, which may be combined together:

- Static threshold setting based on probe installation attributes (type of network access, such as: fibre, DSL, satellite, etc.; nominal bandwidth, …) and preconfigured threshold profiles
- A dynamic self-learning method based on the mean value and the standard deviation of past measurement results (in the last 48 hours) as calculated for each probe, metric and measurement instance
- A dynamic self-learning method limited by preconfigured “good” and “bad” threshold limits. A “bad” threshold limit prevents the threshold adapting too far due to an anomalous condition (otherwise a permanent issue that gradually developed over 48 hours might lead to no anomalies). A “good” limit prevents the system reporting false alarms after the threshold has adapted to an unusually excellent set of results.

Event generation occurs when a measurement falls beyond a threshold, and indicates that an anomaly has been detected. Net Inspector includes, as standard, the generation of events to raise and clear alarms. This has been extended with additional higher and lower severity threshold levels. To support these new functionalities, the data import model has been extended to define attributes, such as the error codes, alarm messages, and alarm severity, of the events generated by the measurements.

Our current extension to Net Inspector includes:

- Extension of the measurements model definition, in order to define threshold calculation methods and event parameters
- Hour and day aggregation of measurement results (minimum, maximum, average, square sum of samples, number of samples)
- Calculation of the mean value and standard deviation for a time interval based on aggregated values
- New three-threshold-levels event generation method

4 AUTOMATED REPAIR

Large-scale measurement infrastructures can be exploited to timely detect potential performance problems. Consider the example in **Error! Reference source not found.**7. The figure depicts a few routers of a single Autonomous System, called AS X, along with internal traffic flows to a given interdomain destination, external to the AS. To reach a given destination, two interdomain paths can be used, but the delay on path P2 is much higher than the delay on P1. If the network operator of AS X detects that some routers are using path P2, it could move traffic to path P1, in order to improve the quality experienced by its customers. We call this "automated repair".

The information about differences in delay on the paths could come from Leone metrics such as the 'Happy Eyeballs' TCP Connection Establishment metric, or more generally from any other Leone metric (see Deliverable D2.1) that can act as a discriminator between more and less desirable interdomain paths.

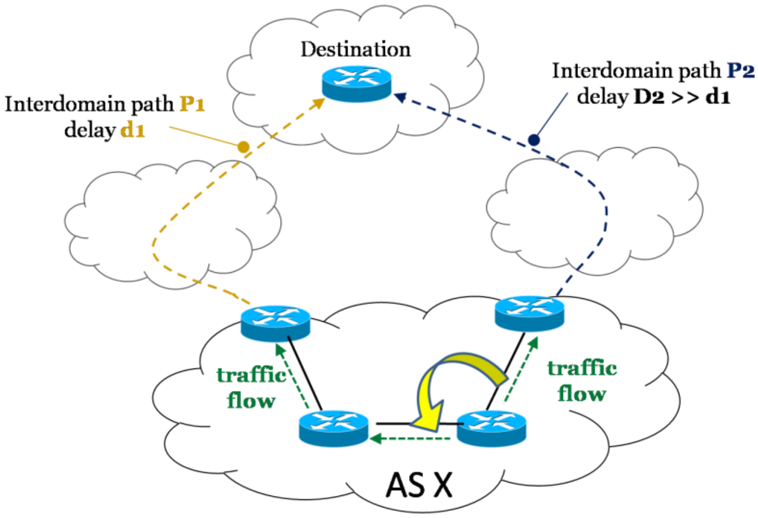

Figure 7: Automated repair use case

We have studied how to move traffic flows by reconfiguring some devices, e.g., routers, in the network. A large-scale system that regularly measures performance can give sufficient information to allow network performance problems to be automatically fixed before users are (highly) impacted.

Unfortunately, reconfigurations can be the source of long and serious network disruptions, because of transient inconsistent states in network devices. In the following, we report the results of our investigation of disruption-free reconfiguration techniques that can be leveraged for automatic repair.

We studied network layer reconfigurations because of their large spectrum of applicability and their independence from specific application level mechanisms. Among the advantages of such a choice, is that we are able to support automated repair in transit networks (i.e. which do not contain the communication end-points). We focused on how to deploy automated repair without packet losses and service disruptions. Note that the final decision to go ahead with a reconfiguration should often be left to a human network operator to confirm, since there are multiple non-technical constraints such as business goals, agreements with neighbouring networks, authority of different NOCs, intra-domain metrics to be optimised for a given customer traffic, etc.

Our approach is to modify the configuration of Interior Gateway Protocols (IGPs), in order to change the relative preference of internal forwarding paths. In flat IGP configurations, the forwarding paths are computed as the shortest paths on the weighted graph representing IGP adjacencies with their respective weights. Hence, basic IGP reconfigurations consist of modifying router configurations to impose a new set of IGP weights.

In the example in Figure 7, an IGP reconfiguration can be used to change IGP weights so that the path consisting of routers v, u and w is shorter than the path made of v and z. As a side effect, this would make v forward traffic over the interdomain path P1.

We explored two different approaches to perform IGP reconfigurations: improved *IGP convergence* and *Ships-in-the-Night (SITN)*. Each approach comes with its own set of pros and cons, which makes it suitable for different automated repair cases. The first is very natural for small configuration changes, such as the shutdown of a link or a router, but becomes impractical when the reconfiguration involves several nodes or when changes have to be applied on a per-flow or per-path (instead of per-link or per-node) granularity. The more general SITN approach can be used in those cases, at the cost of increasing the time needed for the reconfiguration and requiring more resources on the network devices.

Next, we plan to work on a refined implementation and a more complete validation of the proposed reconfiguration techniques. For the validation of our techniques, we envision to rely on simulation platforms and testbeds reproducing real (or realistic) ISP configurations.

4.1 IGP reconfigurations via improved IGP convergence

As mentioned above, a basic approach to move traffic flows (e.g., for automated repair) is to change IGP weights. This makes the IGP converge on the new configuration, e.g., re-compute the new forwarding path according to the new weights. Unfortunately, IGP convergence is not harmless as it comes at the cost of temporary inconsistencies and sub-optimality. We focused on link-state IGPs as they are currently used in most networks. In link-state IGPs, transient forwarding loops can be created during protocol convergence, and can trigger packet losses and service unavailability.

Prior research has proved that the state of a given link can be modified while avoiding forwarding inconsistencies without changing protocol specifications. We complemented these findings by studying the more general problem of gracefully modifying the state of an entire network node, while minimising the induced operational impact. As opposed to the single link-state modification problem, our problem is k-dimensional for a node of degree k. We showed that the interplay between operations applied at the granularity of a node can lead to loops that do not occur in the single-link modification problem. We also devised a polynomial-time algorithm that computes sequences of weight to be configured on the links of the updated node. Our algorithm basically trades the speed of convergence for guarantees of forwarding correctness. In order for the operational impact to be minimal, the algorithm minimises the length of the weight sequences it computes.

By the means of simulations based on real network topologies, we have been able to show the efficiency of our algorithm and the practicality of our approach, as the size of the computed weight sequences are often very limited in practice.

Additional details on this work are reported in Appendix A.

4.2 IGP reconfigurations via the Ships-in-the-Night technique

A more general reconfiguration approach relies on the so-called Ships-in-the-Night (SITN) technique. Essentially, this technique consists of simultaneously running two IGP instances, one with the initial configuration and the other with the final configuration, on all the routers in the network, and in progressively switching the preferred IGP from the initial to the final one, on a per-router basis.

Unfortunately, inconsistent IGP preferences on different routers can give rise to long-lasting network disruptions. Previous work has developed algorithms to avoid forwarding loops during SITN reconfigurations of a network running a single link-state IGP. However, many enterprise networks use multiple IGPs, connected through the route redistribution mechanism.

We studied the problem of performing safe transitions between route redistribution configurations. We investigated anomalies occurring in those reconfigurations, showing that many long-lasting forwarding loops can and do occur if naive techniques are applied. We devised new sufficient conditions for anomaly-free reconfigurations, and we leveraged them to build provably correct reconfiguration procedures. By enabling lossless and fast route redistribution reconfigurations, our procedures are suitable for both short-term (e.g., local repair or traffic engineering) and long-term (e.g., requirement changes) goals.

Additional details on this work are reported in Appendix B.

CONCLUSIONS

An appropriate combination of visualisation and algorithmic techniques, integrated with existing management systems, is often needed for operational support in the complex network ecosystems that operators typically face.

The design space analysis that we performed during the first 10 months of the Leone project highlights promising directions to exploit measurements from large-scale infrastructures for operational support. In this analysis, we proposed several techniques for the effective visualisation of substantially different information obtained from measurement infrastructures, and we implemented prototypes of them. Moreover, we devised algorithms and procedures to react to performance problems via automated (or semi-automated) repair approaches, and we formally proved their correctness. Finally, we extended existing management systems to prepare for the incorporation of our large-scale measurement information.

Our next steps are continue the work on the implementation, evaluation, optimisation and integration of the various techniques.

APPENDIX A

This appendix contains references to our published results.
The corresponding papers are also attached in the following order.

Visualisation

1. Giordano Da Lozzo, Giuseppe Di Battista, Claudio Squarcella. "Visual Discovery of the Correlation between BGP Routing and Round-Trip Delay Active Measurements". In Proc. IMC Workshop on Internet Visualization (WIV), 2012.
2. Massimo Candela, Marco Di Bartolomeo, Giuseppe Di Battista, Claudio Squarcella. "Dynamic Traceroute Visualization at Multiple Abstraction Levels", In Proc. 21st International Symposium on Graph Drawing (GD), 2013. To appear.

Network management integration

3. Juergen Schoenwaelder. "Considerations on using NETCONF with LMAP Measurement Agents". Internet-Draft, 2013.
4. Juergen Schoenwaelder. "A YANG Data Model for LMAP Measurement Agents". Internet-Draft, 2013.
5. Marcelo Bagnulo, Brian Trammell. "An LMAP application for IPFIX". Internet-Draft, 2013.

Automated repair

6. Francois Clad, Pascal Merindol, Stefano Vissicchio, Jean-Jacques Pansiot, Pierre Francois. "Graceful Router Updates for Link-State Protocols". In Proc. IEEE ICNP, 2013. To appear.

APPENDIX B

This appendix contains still unpublished works, e.g., under review in scientific conferences.

The work on a 2.5D Visualisation System (see Section 1.3) is attached as the first one. It is followed by the two still unpublished works on automated repair, namely the work on routing reconfigurations with route redistribution (see Section 3.1.2) and the one on BGP reconfigurations (see Section 3.2).