The MOME Workstation as a Platform for Automatic Analysis of Measurement Data

Marek Dabrowski[1], Jaroslaw Sliwinski[1], Felix Strohmeier[2]

[1]Warsaw University of Technology, Institute of Telecommunications, Poland
[2]Salzburg Research, Austria
E-mail: [1]{mdabrow5,jsliwins}@tele.pw.edu.pl, [2]fstrohmeier@salzburgresearch.at

Abstract: The paper describes the Data Analysis Workstation developed by the IST project MOME (Monitoring and Measurement Cluster). The MOME project maintains the repository of information about publicly available measurement data, like e.g. packet traces, flow traces, routing data, and others. The collected information includes the detailed description of an assumed measurement scenario, as well as selected statistical parameters obtained by data analysis methods. The MOME repository aims at providing assistance to the research community in finding and retrieving raw measurement data, which is most appropriate from the point of view of different research objectives. In this paper we discuss the design and development of the MOME Data Analysis Workstation. It extends the MOME Database with capabilities for automatic statistical analysis of submitted measurement data. The included exemplary results of analysis performed on packet traces captured in a measurement site associated with MOME illustrate that the MOME Database provides the researchers with useful information about available measurement data.

1. Introduction

Traffic analysis and modelling is considered as important topic in research on IP and IP QoS (*Quality of Service*) networks. Despite of the significant work performed recently in this area (see e.g. [3][4]), the Internet traffic is still not well understood and considered as difficult to be described by mathematical formulas. Broad range of Internet research domains rely on well-suited measurement data. Therefore effective research on Internet traffic requires access to large repositories of real and representative measurement data, still being able to find appropriate data for the specific research task. One can observe that a number of research projects were started with the aim of collecting such data (see e.g. [5][6][7][8]), but typically each one does it on its own way, using different measurement tools. Due to the lack of standard formats for data storage and annotation, finding appropriate data among various uncoordinated repositories is often quite difficult.

Therefore, the IST MOME (*Monitoring and Measurement Cluster*) project [1] has developed a meta-database. The MOME meta-database does not perform measurements by itself. Rather, the main goal is to collect and disseminate information about measurement data available from different repositories. The defined meta-data model (i.e. format of stored data) allows the submitters to describe the measurement scenario and environment by a carefully selected set of attributes [2]. Notice that the information about measurement scenario can be complemented by certain statistical parameters, which may be important for assessing the contents of the stored data. In this paper we discuss the attributes for storing the analysis results, which have been included in the MOME meta-data model. We claim that these parameters are the most adequate from the point of view of describing the characteristic features of the captured traffic.

The meta-database approach has been previously studied by several projects dealing with measurement data. For example, the SIMR architecture [9] and CADIA Trends project [10] aim at collecting and annotating various measurement data. The MOME database not only stores the information, but can also perform automated analysis of submitted measurement data. The MOME Data Analysis Platform allows for executing selected analysis tools. New analysis tools can be quite easily integrated in the MOME Workstation, by implementing appropriate filters, which parse their output files and insert results into the database. The access to the results is available for all visitors browsing the MOME database. Together with meta-data describing the assumed measurement scenario, the analysis results should help them in assessing, if a particular raw data set is appropriate for given research targets.

The remainder of this paper is structured as follows. In section 2 we introduce the MOME meta-data model, with special focus on the attributes defined for storing the results of data analysis. Section 3 describes in detail the architecture of MOME Data Analysis Platform. The exemplary case study, which demonstrates the capability of MOME system for performing selected analysis tasks, is presented in section 4. Finally, section 5 concludes the paper.

2. The MOME meta-data model

In this section we describe the meta-data model, which was used as a base for implementation of the MOME meta-repository. The meta-data model consists of a set of attributes, which annotate the most important features of various measurement data. These attributes describe in detail the measurement scenario, like e.g. the type of measurement, type and format of stored data, type of capture platform. Additionally, the meta-data model comprises some statistical parameters, which represent the internal characteristics of measured traffic.

Notice that the statistical parameters selected for implementation of the MOME database correspond only to a small subset of known analysis methods, applicable for a given type of data. However, recall that the main goal of the MOME repository is to assist the researchers in searching for measurement data, which is most appropriate for their needs. Then, they can retrieve the data and perform further processing, using more advanced methods (like e.g. compare the captured traces with various analytical traffic models). Therefore, the attributes of MOME meta-data model should represent only the basic and most important information, which adequately describes the characteristic features of the actual data. In particular, it should allow assessing, if the data is representative for the conditions assumed for particular investigated traffic models. However, we stress that the MOME database is flexible, i.e. it allows for adding new fields, if the results of some new analysis methods will be required. Currently five different types of measurement data are considered and described below: packet traces, flow traces, QoS results, routing data and HTTP traces. In addition, the MOME database supports a generic type for web repositories having mainly a pointer to a base URL.

2.1. Packet traces

The packet traces consist of collections of time-stamped records of packets, captured in certain measurement point. Obviously, the availability of representative packet traces is crucial for developing and validating packet-level traffic models. Such traces are also essential in the relatively new area of research on multi-level and multi-timescale traffic models (see e.g. [11]).

Notice that each trace is captured at a certain point of a network, only within a finite time interval. The trace can be regarded as representative, if its relevant features sufficiently reflect the general characteristics, and can be confirmed by a trace collected at a different time, or in different network. For example, different traffic models are proposed for the access and core networks, which can differ essentially from the point of view of level of traffic aggregation. For evaluating the traffic model, developed for core or access network, the used packet traces must be representative for the assumed type of network. Additionally, the duration of the trace should be sufficiently long to allow evaluating the stationary behaviour of captured traffic (or, to reveal its non-stationary behaviour).

To allow assessing if the trace is representative and appropriate, the information stored in the MOME meta-database should include such information, as: type of measured network, measurement time, location of capture device, speed of monitored link, capture method, filtering and anonymisation rules, capture platform, and the storage format of collected data. This information gives us detailed description of the assumed measurement scenario. In addition, several attributes for storing *analysis results* were defined, as presented in Table 1.

Table 1. Packet traces analyses

Analysis name	**Data type**
Average packet inter-arrival time	Real
Average packet size	Real
Average packet rate	Real
Packet size histogram	Graph
Average bit rate	Real
Average bit rate per protocol	Graph
Average bit rate per application	Graph
Time-series of bit rates, calculated over non-overlapping intervals	Graph
Stationary variance of bit rate	Real
Required value of effective bandwidth	Graph
Required traffic descriptor parameters	Graph
Value of Hurst parameter	Real

The first group of attributes corresponds to calculation of the empirical statistical parameters. The input data consists of a statistical sample of observed packet lengths, or packet inter-arrival times. Based on this sample, one can easily calculate the following parameters: **average packet inter-arrival time**, **average packet size**, **packet size histogram**, **average packet rate** and **average bit rate**. Additionally, **average bit rate per protocol** and **per application** correspond to the split of the captured traffic between different transport protocols (TCP, UDP) and different applications (identified by the port numbers). These statistical parameters (and graphs) give us basic information about the volume and type of traffic carried on the monitored link.

The **time-series of bit rates** shows the plot of bit rate (calculated as total number of bits transmitted within the time interval, divided by the length of this interval), as a function of time. It gives us information, how the observed traffic fluctuated throughout the trace duration. Notice that the rates can be calculated over intervals of different length. In MOME database, a few typical values are considered: 10ms (to reveal very-short time scale traffic changes), 1 second (short time-scale), and 3 minutes (medium time-scale). From the set of obtained bit rate samples we can calculate the **variance of bit rate**. This parameter attempts to quantitatively estimate the level of variability of traffic observed within trace duration. The set of bit rate samples can be also used as input data for more advanced analysis tasks, like e.g. fitting the parameters of assumed analytical traffic models, and validating their correctness. Currently, such methods are not considered in the MOME database, but can be added in the future, if needed.

The **effective bandwidth** gives us estimation of the resource requirements of captured traffic. In IP QoS networks, such information is necessary for performing the admission control decision. The algorithms for measurement-based estimation of effective bandwidth were discussed for example in [3][12][13].

The **traffic descriptor** parameters provide an envelope, which describes the worst-case behaviour of the traffic arrival process, in a deterministic way. Typically, the *token bucket* mechanism is used for this purpose. The information about required values of token bucket parameters can help in assessing the worst-case variability of the traffic captured in particular trace.

The last considered attribute is the **Hurst parameter**, which evaluates the level of self-similarity of captured traffic. Traffic is regarded as self-similar, if it is highly variable on multiple time scales. Several known methods for estimating the value of Hurst parameter are based on analysing the plot of Index of Dispersion for Counts (IDC), analysing the variance-time plot, the R/S analysis, periodogram-based analysis in frequency domain, or wavelet-based analysis [3][14].

2.2. Flow traces

The flow traces consist of a collection of records of distinct flows, observed in certain measurement point. Single flows can be defined as by IETF-IPFIX in [19]. The flow record contains the timestamps of flow arrival and departure, as well as the number of packets and bytes transmitted within its lifetime. The flow traces are useful in developing and validating the traffic models on the flow (or connection) level. Similarly to the case of packet traces, the capture location and duration should be considered for assessing the usefulness of the trace. Therefore, the following meta-data attributes have been defined: type of measured network, measurement time, location of capture device, speed of the monitored link, capture method, filtering and anonymisation rules, type of capture platform, and the storage format. The proposed list of adequate parameters corresponding to statistical analysis of flow records is presented in Table 2.

Table 2. Flow traces analyses

Analysis name	Data type
Average flow inter-arrival time	Real
Average flow duration	Real
Average number of packets per flow	Real
Average number of bytes per flow	Real
Average flow arrival rate	Real
Average traffic bit rate	Real

The **average flow inter-arrival time** and **average flow duration** describe basic flow-level characteristics of observed traffic. In the case of packet-based networks, the "size" of the flow must be described not only by its duration, but also by the total number of transmitted bytes and packets. Therefore, both the **average number of packets per flow** and **average number of bytes per flow** are included as information about typical size of observed flows. The **average flow arrival rate** corresponds to the intensity of new flow arrivals. Finally, the **average bit rate** gives us long-term estimation of utilised resources.

2.3. QoS results

In the case of QoS measurements, the raw data consists of a set, or time-series, of measured values of QoS metrics, like e.g.: one way delay, delay variation, packet loss ratio, etc. The singleton values correspond to the measurement taken on a single packet (probe packet in the case of active measurement, or data packet in the case of passive measurements).

The meta-data attributes should give precise information on the network, where the measurement was performed, and the time, when it was performed. Therefore, the defined meta-data model includes such information, like: the name and type of network, location of measurement equipment, method for time synchronisation between the transmitter and receiver, calculated metrics, and the storage format of the measurement results. The proposed additional attributes related with data analysis tasks are presented in Table 3.

Table 3. QoS results analyses

Analysis name	Data type
Minimum packet delay	Real
Average packet delay	Real
Maximum packet delay	Real
10-percentile of packet delay	Real
90-percentile of packet delay	Real
Histogram of packet delay	Graph
Minimum packet delay variation	Real
Average packet delay variation	Real
Maximum packet delay variation	Real
Packet loss ratio	Real

The analysed statistical sample consists of the set of singleton measurements of packet delay, or packet delay variation. The basic statistics calculated from this sample include: **minimum**, **maximum**, and **average** value, or certain **quantiles** of the distribution of the considered quantity (packet delay, or packet delay variation). Additionally, the **histogram can** give us estimation of the distribution of the packet delay or delay variation. Finally, the **packet loss ratio** completes the assessment of QoS delivered to the measurement traffic.

2.4. Routing data

The routing data consists of the snapshots of routing tables, or collections of routing update messages captured within particular measurement interval. The routing data may be useful for investigating the changes of logical inter-domain topologies, observing trends in number of advertised prefixes, or detecting abnormal behaviours of the inter-domain routing. The selected statistical attributes for routing data are presented in Table 4.

Table 4. Routing data analyses

Analysis name	Data type
Routing table size	Integer
Number of update messages	Integer

The **routing table size** (expressed in the number of prefixes) allows us for assessing the global state of network at a time of taking the capture. It can be useful for investigating the trends in routing table size and growth of the total number of advertised prefixes. The **number of updates** can be evaluated in the case of capturing the routing protocol messages.

2.5. HTTP traces

The HTTP (*Hyper Text Transfer Protocol*) is the application layer protocol, designed for the web application. The HTTP trace contains records of observed user HTTP requests and server HTTP responses typically collected by log-files in web- or proxy-servers. The information stored in such traces corresponds to the application (or session) layer traffic model. The statistical parameters selected for describing the contents of HTTP traces in the MOME database are shown in Table 5.

Table 5. HTTP traces analyses

Analysis name	Data type
Average HTTP request inter-arrival time	Real
Average response size in packets	Real
Average response size in bytes	Real
Average HTTP request arrival rate	Real

Several statistical parameters can be useful for evaluating the activity of web application users. The arrival process of user request is described by **average request inter-arrival time** and **average request arrival rate**. The basic parameters characterizing the server responses are **average response size**, measured **in packets** and **in bytes**.

3. Automated analysis of measurement data

Based on the proposed meta-data model, the MOME meta-repository has been implemented and deployed (see Fig. 1). It is available publicly via the MOME project webpage [1]. The main part of the system is the **Database**, which stores all the MOME meta-data. Users can browse the MOME database using the **Graphical User Interface (GUI)**, implemented on the MOME **Web-server**.

The user can request analysis of raw data, stored in some public repository or available directly from the measurement site associated with MOME. The data analysis is performed on the **MOME Data Analysis Workstation** (‘data analysis’ block on Fig. 1). The raw data file is retrieved from its original location and temporarily stored it on a local disk. After performing the analysis, the results are inserted into appropriate fields in the MOME Database.

The MOME system assumes that external, freely available data analysis tools can be used for performing actual operations on the measurement data. Each tool must be adapted to the MOME Workstation by implementing appropriate filters, which are able to read and parse the analysis results. At the beginning stage of the project, only few exemplary analysis tools were selected. They allow performing basic analysis tasks for packet-level traces, stored in the popular format of *libpcap* [15] However, new analysis tools can be quite easily added, depending on the needs.

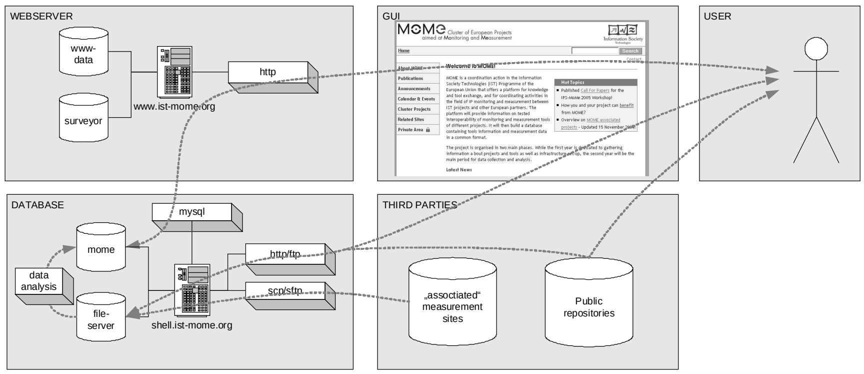

Fig. 1. Deployment of the MOME meta-repository

3.1. Executing data analysis tools

Fig. 2 illustrates step-by-step the process of executing data analysis task. When an authorised user browses the database, the **Analysis Request Manager (ARM)**, which is a part of MOME GUI, displays the data analysis tasks available for particular measurement data. If the user requests the analysis (step 1), the ARM inserts a new pending request into the MOME database.

The software modules, which constitute the MOME **Data Analysis Workstation,** are implemented as independent processes and can run concurrently on the MOME server. The **Autonomous Process Manager (PM)** is implemented as a PHP CLI (*Command Line Interface*) script and is periodically executed on the MOME Database server. The PM searches the database for new pending analysis tasks. If PM finds out that new analysis task has been requested, it starts the **Download Manager (DM)**, providing it as parameter the link to the file, which should be analysed (step 2).

The DM, which is also implemented as a PHP CLI script, retrieves the file identified by the provided link and saves it on the local storage (step 3). Notice that multiple DMs can be started at the same time, to download different remote files. After the DM has retrieved the file, the PM can change the status of the meta-data entry in the MOME database to "*analysis pending*, *local copy*" and execute the **Analysis Manager (AM)** (step 4). The parameters, which he provides to the AM, are: the type of requested analysis, and the name of the local file with raw data. The status of the appropriate entry in the meta-database is now changed to "*analysis in progress*".

The **Analysis Manager (AM)** is implemented as a PHP CLI script. Only one instance of the AM runs at a time. The AM executes the external analysis tool (step 5), providing the name of the local file to be analysed. Therefore the tool needs to be installed with appropriate privileges for running on the MOME server. After finishing the actual analysis work, the analysis tool saves the results into the output file (step 6). The AM parses this file and inserts appropriate values into the MOME database (step

7). Finally, it sets the data status to "*analysis done*" and finishes the data analysis procedure.

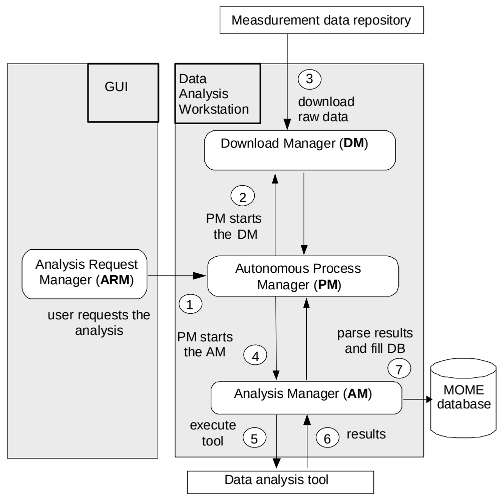

Fig. 2. Operations of the MOME Data Analysis Workstation

3.2. Adding new analysis tools

Fig. 3 shows the internal structure of the AM module. Recall that this module manages the execution of data analysis tools integrated in the MOME Workstation. The '*check available tools*' function finds available analysis tools and selects the tool that is capable to perform the requested analysis. The '*execute analysis tool*' function starts the actual analysis tool, providing as parameter the name of the local file with raw measurement data. The analysis tool performs the requested operations and saves results to the local output file. This file must be read by the function '*parse results, plot graphs*'. This function parses the results, extracts appropriate data and, if necessary, generates graphs using the *GNUPlot* tool. Finally, the results are inserted into the appropriate fields of the MOME database ('*update database*' function).

The actual analysis operations are performed by the external data analysis tools. Any freely available analysis tool can be quite easily integrated in the MOME system, if it meets the following requirements: can be executed on the Linux system, can read input data from the file and write output data to the text file. Notice that particular analysis tools may be dedicated for specific types of measurement data (e.g. packet or flow traces), and specific input data formats (e.g. *pcap* in the case of packet traces, or *NetFlow* in the case of flow traces).

Adding new analysis tool to the MOME Data Analysis Workstation only requires implementing two new functions in the AM: "*execute analysis tool*" and "*parse results, plot graphs*" (see Fig. 3). Those functions act as an interface to the external tool and are implemented separately for each tool supported by the system.

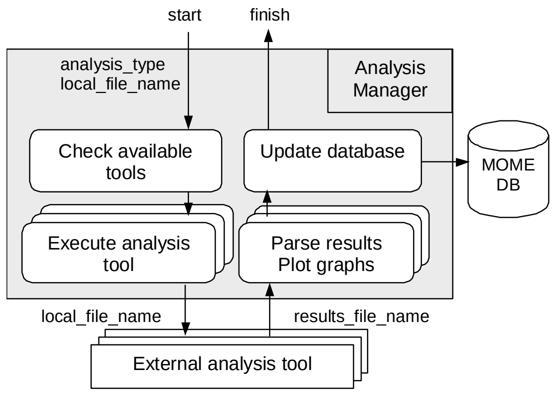

Fig. 3. Operations of the Analysis Manager

3.3. Basic analysis tasks for packet traces

As a first tool for integrating into the MOME Data Analysis Workstation, a simple utility called *tcpdstat* [16] has been chosen. *Tcpdstat* performs basic analysis of packet-level traces captured by tools like *tcpdump* [15] or *tcpdpriv* [17]. As input data, it reads the binary files saved in the *libpcap* format. The statistics calculated by *tcpdstat* include: file size, measurement start-time and end-time, total number of captured bytes and packets, average and peak bit rate and standard deviation, number of observed distinct flows, average number of packets per flow, top 10 flow sizes, number of observed distinct IP addresses, packet size distribution, average rate, split between different protocols and applications.

The *tcpdstat* saves the results in a form of human-readable ASCII file (see Fig. 4). Some of the values calculated by *tcpdstat* can be directly used for filling appropriate attributes of the MOME meta-data model (see section 2.1). For example, the **average bit rate** can be directly extracted by parsing the *tcpdstat* output file. Notice that the **average packet rate** can be easily calculated, based on other values produced by *tcpdstat*. Additionally, based on the textual data parsed from the *tcpdstat* output, MOME Workstation creates the graphs of **packet size histogram**, and **average bit rate per protocol** and **per application**. The *GNUPlot* tool, executed on the MOME server, produces the actual graph, which is then stored as a PNG picture in the MOME database.

```
TotalTime: 899.82 seconds
TotalCapSize: 166.31MB  CapLen: 96 bytes
# of packets: 3103414 (1490.21MB)
AvgRate: 13.89Mbps  stddev:1.10M   PeakRate: 18.61Mbps

### IP flow (unique src/dst pair) Information ###
# of flows: 279098  (avg. 11.12 pkts/flow)
Top 10 big flow size (bytes/total in %):
  7.6%  2.9%  2.8%  2.1%  1.3%  1.3%  1.2%  1.2%  1.1%
1.1%

### IP address Information ###
# of IPv4 addresses: 190536
Top 10 bandwidth usage (bytes/total in %):
 29.7% 13.2%  9.1%  7.8%  4.9%  4.5%  3.4%  3.1%  2.9%
2.9%
### Packet Size Distribution (including MAC headers) ###
<<<<
 [   16-   31]:       8285
 [   32-   63]:     948296
 [   64-  127]:     835986
 [  128-  255]:     165792
 [  256-  511]:     158287
 [  512- 1023]:     137072
 [ 1024- 2047]:     849696
```

Fig. 4. Fragment of exemplary results file of tcpdstat

4. Exemplary results of analysis performed by the MOME Data Analysis Workstation

In this section we demonstrate the capabilities of the MOME Data Analysis Workstation for analysing the data obtained in a measurement site “associated” with MOME. By “associated” measurement site we mean that the results of performed measurements are automatically uploaded and annotated in the MOME meta-repository. We show that the results provided by simple analysis tools integrated in the MOME system can provide interesting information for the researchers visiting our meta-repository.

4.1. Capturing measurement data at the MOME associated sites

As a reference and for demonstration, a monitoring and meta-data storage process for packet traces has been implemented in the premises of Fachhochschule (FH) Salzburg. The information from the packet headers is captured, made anonymous, compressed and uploaded together with necessary meta-data to the MOME database. This approach provides an automatic way to continuously get up-to-date measurement data to the MOME database. The procedure is divided into the following steps:

(1) Header capture + anonymisation (*tcpdpriv* [17])
(2) Data compression (*bz2*)
(3) Meta data construction
(4) Data upload (trace + meta data), by *ftp*
(5) Meta data entry to MOME database

The capturing in step (1) can be done according to different schedules and filters. A first scenario assumes a full packet capture process, starting daily at 02:00 until 23:59, with traces stored in hourly intervals. Data transfer to the analysis station is scheduled for the time period between 00:00 and 01:59. Note that the data upload is never performed during the capturing process, to avoid interferences with the monitored traffic. Due to the low access link speed (4Mbps), the amount of a full data capture can be handled without losses. Additionally, header extraction and compression reduce the accumulating amount of data usually below 500MB per day. This scenario will be adapted based on the produced analysis results.

While steps (1) - (4) are performed by the monitoring host, the last step is done at the MOME database server in regular intervals. The following figure shows the network set-up as it was installed at the reference capture point at the FH Salzburg. Due to the simple installation without major changes in the operational network, a set-up with a mirror port on the operational edge router has been preferred against other installations, e.g. with Linux routers or splitters. Only outgoing traffic is monitored.

The automatically generated meta-data is listed in Table 6. Exemplary values are given for a specific monitoring scenario at FH Salzburg.

After the meta-data entry is available in the MOME database, analysis requests can be set on these traces by using the web interface. In the case that all automatically uploaded monitoring traces are dedicated for analysis, the analysis requests can be set in the database automatically.

Table 6. The meta-data values generated for the trace collected in measurement site “associated” with MOME

Common Attributes		**Packet Trace Specific Attributes**	
Attribute	**Generated Value**	**Attribute**	**Generated Value**
StartTime	local start time of the measurement trace	NetworkType	"WAN Access Network"
EndTime	local end time of the measurement trace	CollectorLocation	"AT, Salzburg, FH Salzburg"
FileSize	comressed file size in Bytes	TrafficType	"operational unidirectional outgoing traffic, educational institution"
Description	"Trace FH Salzburg"	LinkProtocol	X.21
DataLocation	link to the uploaded file	LinkSpeed	4 Mbps
FileCompression	bz2	CaptureMode	mirror port
MD5Sum	the md5sum	FilterRules	None
SubmissionDate	local time of data upload	NumberPackets	counted number of packets
LastUpdate	database insertion date	TraceAnonymisation	-P99 -A50
Tool	tcpdpriv		
DataAvailability	local copy	CapturePlatform	Linux 2.6.9
SubmitterID	a dedicated id	DataFormat	libpcap
		AdditionalInfo	2x2Mbps X.21 connections

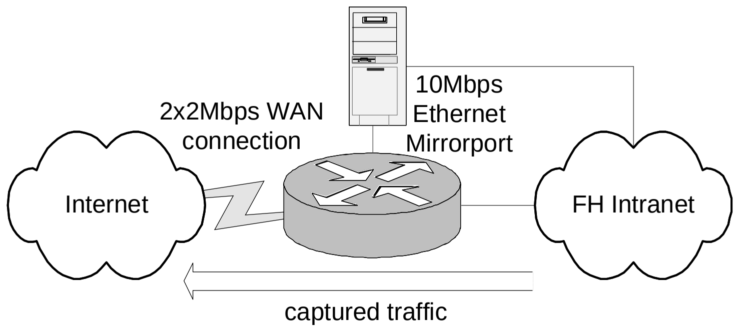

Fig. 5. Network Set-up

4.2. Analysing packet traces captured at MOME associated sites

Below we present the results of analysis performed on the 1-hour long traces from the measurement site in FH Salzburg, captured during three days on the 20th February (Sunday), 28th of February (Monday), and 1st of March (Tuesday) 2005. The traces were automatically uploaded and annotated in the MOME meta-repository. Then, the trace analysis was performed by the MOME Data Analysis Workstation, as described in section 3. The analysis results can be accessed directly via the MOME repository GUI (see Fig. 6). The average values of bit rate, packet rate, packet size and inter-arrival time are directly displayed on the screen, together with other detailed information about the trace. The plots of packet size histogram, as well as per-protocol and per-application distribution of bit rate, are displayed after clicking on the *'graph'* button.

Traffic rate, averaged over entire trace duration, in bit/s	2442547
Average packet inter-arrival time in sec	0.003186
Average packet size in bytes	972.7
Average packet arrival rate in pkt/sec	313.89
Histogram of packet sizes	graph
Bandwidth use per-protocol	graph
Bandwidth use per-application	graph

Fig. 6. The representation of analysis results for exemplary packet trace annotated in the MOME meta-database

The presented case study validates the design and implementation of the MOME Data Analysis Workstation. In addition, it demonstrates how MOME meta-repository can provide the research community with relevant, pre-processed information about the gathered traces. For example, Fig. 7a and Fig. 7b present the daily plots of average bit rate, and packet rate, respectively. This kind of plots can be quite easily obtained from the analysis results available in the MOME database. As one could expect, the traffic peaks during the afternoon hours are clearly visible. One can also observe the fluctuations in traffic load on different days of week. The researcher looking for traces needed for validation of some traffic model can use these analysis results, combined with the detailed description of the measurement scenario (see Table 6), for selecting the most appropriate trace among the collection available in the MOME database. For example, he can choose to download and process further only those traces, which were captured during higher utilisation period of the FH Salzburg WAN connection.

Another MOME data analysis results are demonstrated by the graphs on Fig. 9. They show the distribution of measured bit rate, split between different applications (recognized by port numbers in the packet header). Fig. 9 corresponds to one of the FH Salzburg traces, while Fig. 9 to one of the traces from the public repository [18]. As a simple example of usage of such information, consider a researcher looking for traces needed for his work on modelling traffic related with the audio streaming application (in particular, *realaudio*), or with peer2peer applications (*Kazaa*, or *Napster*). Thanks to the graph from Fig. 9, he can conclude that the FH Salzburg trace is not useful for him, because such applications are not represented in this trace. On the other hand, one can find the traffic related with the considered applications in the MAWI trace (see Fig. 9b).

Summarising, the MOME repository can assist the researchers in finding and retrieving the traces, which are most appropriate for their needs. Although the currently integrated analysis tasks are quite simple, the demonstrated scenarios show that they can achieve the main goal, which is providing the researchers with high-level information about the contents of the trace.

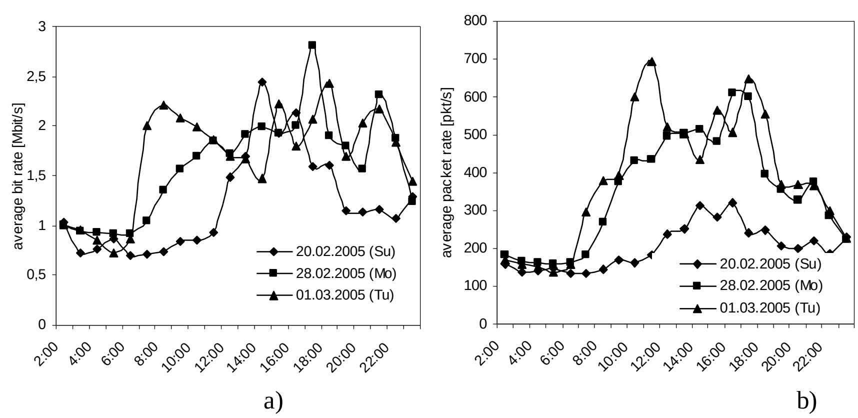

Fig. 7. The plots of: a) average bit rate and b) average packet rate, calculated for 1-hour long traces collected during 20 Feb, 28 Feb and 01 Mar 2005

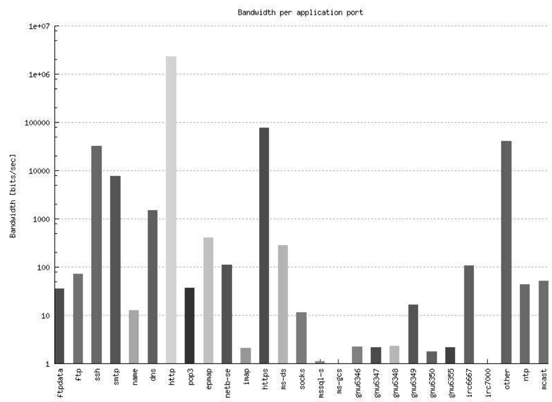

Fig. 8. The average bit rate of traffic related with different applications in the FH Salzburg trace from 20.02.2005, 14:00-15:00

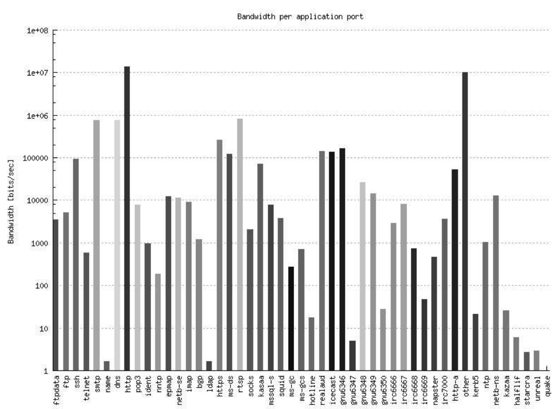

Fig. 9. The average bit rate of traffic related with different applications in the MAWI trace from 31.01.2005, 14:00-14:15

5. Conclusions

In this paper we presented the automated measurement data analysis approach assumed by the MOME Workstation. The goal of this approach is to enhance the existing MOME measurement data catalogue, storing general information about measurement data, like: time, duration or file sizes of measurement traces, with additional attributes describing the contents of the measurement trace itself. A framework providing the measurement data handling tasks, like downloading and the database access has been presented. It is extensible for different analysis algorithms on different datasets. The predefined analysis results will help researchers to find traces they want to use for further studies.

Acknowledgment

The work was supported by the IST Coordination Action MOME (*Monitoring and Measurement Cluster*), IST-2003-001990.

References

1. IST-MOME web site, http://www.ist-mome.org
2. P. Aranda-Gutierrez et al., „MOME: An advanced measurement meta-repository", in proceedings of *3rd International Workshop on Internet Performance, Simulation, Monitoring and Measurements*, IPS-MoMe 2005, Warsaw, March 2005
3. P. Tran-Gia, N. Vicari (eds.), "Impacts of New Services on the Architecture and Performance of Broadband Networks", COST257 Final Report, compuTEAM, Wuerzburg, 2000
4. M. Menth (ed.), "Analysis and Design of Advanced Multiservice Networks Supporting Mobility, Multimedia and Interworking", COST279 Midterm Report, Aracne, Roma, 2004
5. The Internet Traffic Archive web site, http://ita.ee.lbl.gov/index.html
6. The NLANR Network Traffic Packet Header Traces web site, http://pma.nlanr.net/Traces/
7. The NLANR Measurement and Network Analysis web site, http://watt.nlanr.net/active/maps/ampmap_active.php
8. The RIPE NCC Routing Information Service Raw Data, http://data.ris.ripe.net/
9. M. Allman, E. Blanton, W. Eddy, "A Scalable System for Sharing Internet Measurements", in P*roceedings of Passive and Active Measurements*, PAM 2002, Fort Collins, March 2002
10. The CAIDA "Correlating Heterogeneous Measurement Data to Achieve System-Level Analysis of Internet Traffic Trends" project, http://www.caida.org/projects/trends/
11. P. Salvador, A. Pacheco, R. Valadas, "Multiscale Fitting Procedure using Markov Modulated Poisson Processes", *Telecommunication Systems Journal*, 23(1-2):123-148, June 2003
12. C. Courcoubetis, V.A. Siris, G.D. Stamoulis. "Application of the Many Sources Asymptotic and Effective Bandwidths to Traffic Engineering", *Telecommunication Systems*, 12(2-3):167-191, 1999
13. M. Dabrowski, F. Strohmeier, "Measurement-based Admission Control in the AQUILA Network and Improvements by Passive Measurements", *LNCS 2698*, W. Burakowski, B. Koch, A. Beben (eds.), Springer 2003
14. D. Veitch, P. Abry, "A wavelet based joint estimator of the parameters of long-range dependence", *IEEE Transactions on Information Theory*, vol. 45 no. 3, pp.878-897, 1999
15. The *tcpdump* web site, *http://www.tcpdump.org/*
16. The *tcpdstat* web site, *http://staff.washington.edu/dittrich/talks/core02/tools/tools.html*
17. The *tcpdpriv* web site, *http://fly.isti.cnr.it/software/tcpdpriv/*
18. The MAWI working group traffic archive, *http://mawi.wide.ad.jp/mawi*
19. J. Quittek, T. Zseby, B. Claise, S. Zander, "Requirements for IP Flow Information Export", RFC3917, IETF, October 2004