A Platform for Predicting Real-time Service-level Metrics from Device Statistics

Rerngvit Yanggratoke*, Jawwad Ahmed†, John Ardelius‡, Christofer Flinta†,
Andreas Johnsson†, Daniel Gillblad‡, and Rolf Stadler*‡
*ACCESS Linnaeus Center, KTH Royal Institute of Technology, Sweden Email: {rerngvit,stadler}@kth.se
†Ericsson Research, Sweden Email:{jawwad.ahmed,christofer.flinta,andreas.a.johnsson}@ericsson.com
‡Swedish Institute of Computer Science (SICS), Sweden Email:{john,dgi}@sics.se

***Abstract*—Predicting performance metrics for cloud services is critical for real-time service assurance. We demonstrate a platform for estimating real-time service-level metrics. Statistical learning methods on device statistics are used to predict metrics for services running on these devices.**

I. Background

Understanding and predicting the performance of telecom cloud services is intrinsically hard. Such services involve large and complex software systems that run on general-purpose platforms and operating systems, which do not provide real-time guarantees. In our companion paper [1] presented at IM2015 we described a novel approach for predicting real-time service-level metrics from device statistics.

The approach is based upon statistical learning whereby the behaviour of the target system is learned from observations. Specifically, we collect statistics from a Linux kernel of a server machine, which we call X, and predict client-side metrics for a video-streaming service (VLC), which we refer to as Y, e.g., video frame rates and audio buffer rates. The fact that we collect thousands of kernel variables, while omitting service instrumentation, makes our approach service-independent and unique.

In [1], we present details of the above approach and its evaluation in an offline setting. In particular, we collect traces produced on our testbed under different load patterns, we apply well-known statistical learning methods on the traces to produce models for predicting service-level metrics, and we evaluate these models against test data from the traces. The traces are publicly available [2].

II. Testbed

This demonstration includes a platform that implements the above approach in an online setting, which is illustrated in Figure 1. The platform is an extension of the setup described in [1]. A management station provides access to the KTH testbed over the Internet and displays measurements and predictions from the platform running on the testbed.

The testbed is shown in Figure 2. The setup includes four parts, namely, a server cluster that provides the video-on-demand service over HTTP, a client machine that runs video-on-demand sessions, a load generator that creates the aggregate demand of a set of VoD clients, and an analytics engine that produces model predictions from well-known statistical learning methods, e.g., regression tree, linear regression, and random forest.

The server cluster consists of six machines, one load balancer machine, three machines that each host a web server and a transcoder, and two networked storage machines. Each machine i in the server cluster runs a sensor that periodically reads out the vector X_i and streams it to the analytics engine. The client machine runs a VLC client, whose sensor extracts service-level events. At the start of every second, the sensor collects the events from the last second, computes the Y metrics and streams them to the analytics engine. The load generator machine dynamically spawns and terminates VLC clients, depending on the specific load pattern that is executed during the demonstration. Receiving a VoD session request from a client, the load balancer server forwards the HTTP request to a backend web server. To respond to a request forwarded from the load balancer, the web server spawns a transcoding instance to transcode a selected video, whereby the raw video content is retrieved over the network from one of the networked storage machines.

III. Demonstration

The demonstration shows the predictions of service-level metrics and the accuracies of those predictions in real-time. The management station displays real-time measurements of video frame rates and audio buffer rates from a sampled VLC client. At the same time, it shows the predictions for these metrics based on the current device statistics collected from the platform. Predictions of the service statistics are given for different online learning methods and load patterns.

References

[1] R. Yanggratoke, J. Ahmed, J. Ardelius, C. Flinta, A. Johnsson, D. Gillblad, and R. Stadler, "Predicting real-time service-level metrics from device statistics," in *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on*. IEEE, 2015.

[2] ——, "Linux kernel statistics from a video server and service metrics from a video client." 2014, distributed by Machine learning data set repository [MLData.org]. http://mldata.org/repository/data/viewslug/realm-im2015-vod-traces.

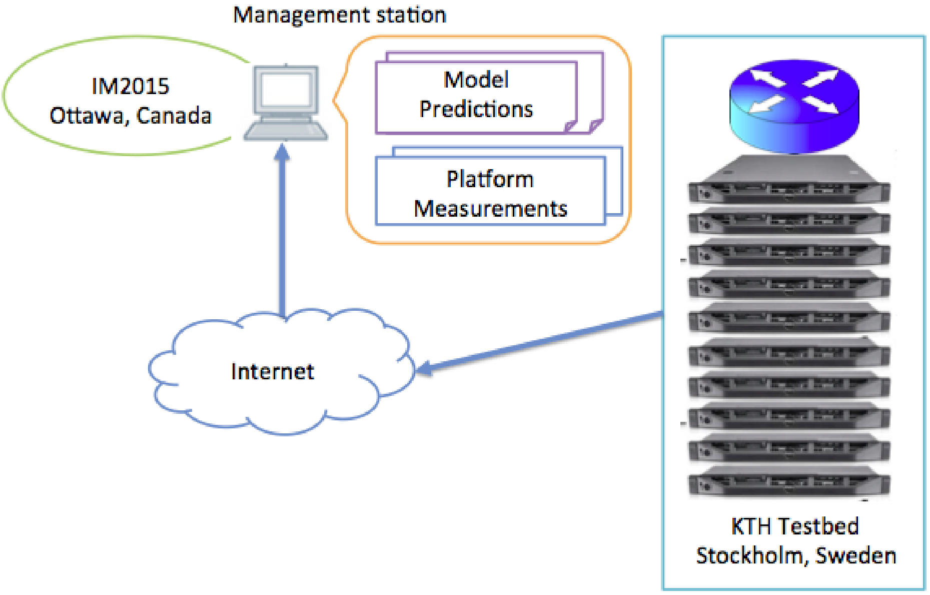

Fig. 1. Demonstration setup

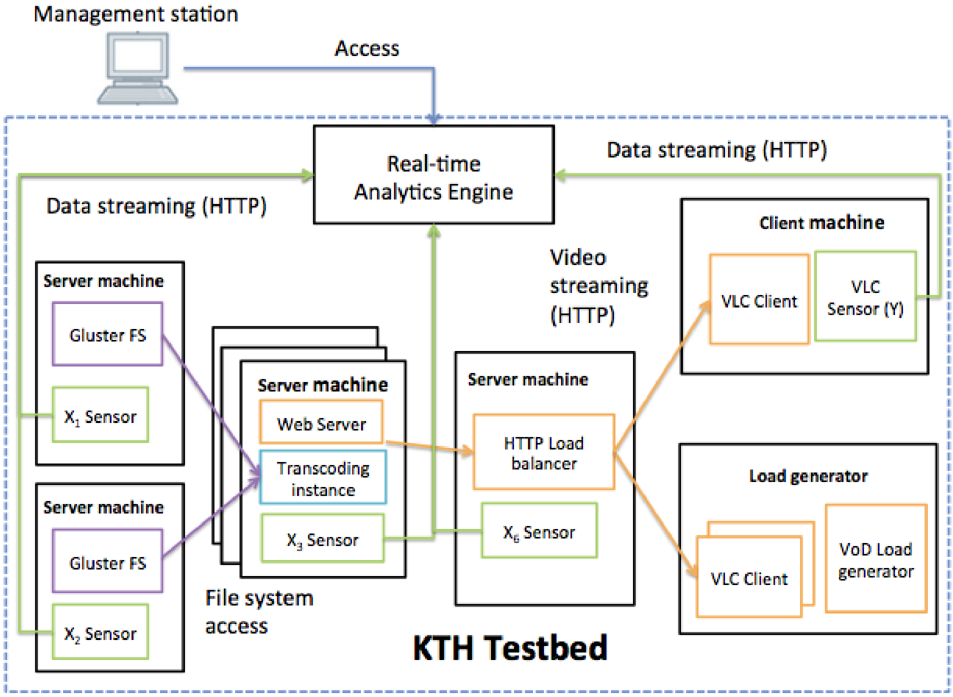

Fig. 2. Testbed setup for predicting service-level metrics Y from device statistics $X = [X_1, X_2, ..., X_6]$