

Banners Network Performance Boosting Case Study

By Moshe Kaplan, RocketTier





Startup Your Engines

Banners Network Performance Boosting Case Study

by Moshe Kaplan, RocketTier

Abstract

A banner network firm was facing its software system technological limitations. The business projections of the company required the system to support 200 times more events, when it couldn't handle half the current events.

Our solution, based on our methodology and algorithms, is designed to solve such performance bottlenecks and scale out limitations.

We provided an immediate relief solution in a short time frame by redesigning and implementing next generation solutions employing grid and in-memory databases. This allowed our client reaching the business requirements and achieving a long term scalable solution, high availability, reduced hardware costs.



Startup Your Engines

Contents

Introduction	4
RockeTier Methodology	4
Affiliate Marketing Overview	5
Affiliate Marketing Software Performance Challenges	6
Bottleneck Detection	7
Solutions Rating	7
The Roadmap	8
Scale Up and Scale Out	9
Bottom Line	10



Startup Your Engines

Introduction

Software performance problems have two traditional solutions. The first is investing money on new and improved hardware with additional resources (CPUs, memory etc.). This may be very expensive, since in order to double the performance you require 4 times the money, which means spending x400 money for a x20 performance improvement.

Twice faster machines are 4 times more expensive

The other solution is to rewrite the entire system based on new technology, but this involves a great risk, since it requires great deal of time and money and by the time you get a new working version the market moves on.

Whole system migration to a new technology is risky

There is also the consideration of the downtime during which your system is unavailable due to the technology or hardware migration.

RockeTier Methodology

Based on our vast experience in software development, system architecture and business requirements we, at RockeTier, designed a novel methodology to release performance bottlenecks, which achieves better performance in shorter time frames and reduced risk. The methodology includes 5 steps in the road to boost your system's performance.

RockeTier methodology achieves better performance in shorter time frames and reduced risk

The First step is to analyze the system architecture and the main business processes, and only then detect the main hardware bottlenecks and the related business process causing them, thereby focusing the effort where it would create the greatest return.

The second step is to rate the bottlenecks by importance and provide immediate practical recommendations to improve system performance and calm your clients down.

In the third step we implement the recommendations and **provide immediate effective relief to the system** in a short time frame (small code changes and system tuning to give you time to design and implement bigger changes without losing clients).

The next step is to plan a road map and when necessary, redesign next generation solutions, using robust and scalable solutions such as grid, cloud computing and in-memory database. This is done ONLY for intensive business processes, in order to save time and money on irrelevant code changes.

The final step is to scale up and scale out when redesign is necessary.

Our methodology is to boost system performance by focusing only on the critical components and enable an evolutionary migration to a scalable platform based on the agile methodology, thus reducing overall risk by avoiding full rewrite of the system or spending a fortune on new hardware.

Focus on critical components to achieve better performance and reduce risk

To better explain our methodology we will use a case study. This case study is based on a project we carried for an internet application company operating an affiliate marketing solution that needed a massive performance boost.

Affiliate Marketing Overview

Affiliate Marketing, a.k.a Banners Network, is an online advertising channel; in which advertisers pay publishers for promoting their products or services on the publisher's web site. Advertisers pay their publishers only when the new client introduction results in a sale or a lead, making it a low-risk, high-reward environment for both parties. Every impression (viewing of an ad), click or sale is recorded by the affiliate management company differently.

Banners Network is an online advertising channel

The affiliate management company records the number of impressions per banner, even though the publisher doesn't pay for impressions, because the publisher wants to know the conversion rate from impressions to actual transactions.

Banner network billing systems record every impression although charge is done on actual actions like sales and clicks

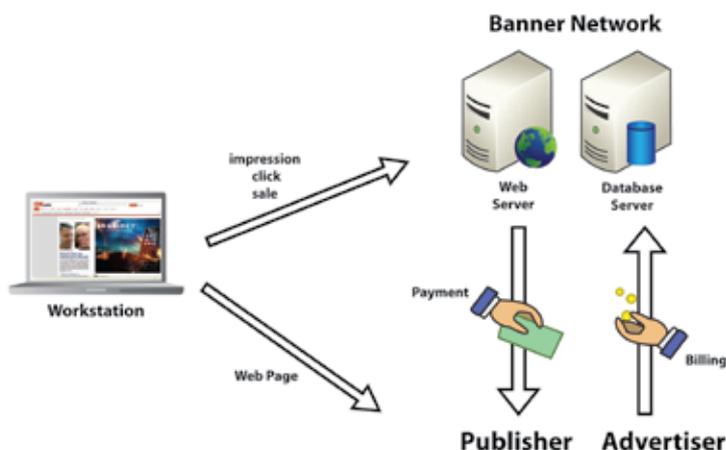


Figure 1 - Banner networks major business processes

Using the recorded transactions, the affiliate management company can ensure fair commission payment for the publisher, and all of the administration and verification necessary to ensure quality sales and leads for the advertiser.

Affiliate Marketing Software Performance Challenges

Successful affiliate management systems handle billions of events per day and hold records of all transactions in the system. Therefore, the system has to work extremely efficiently, which entails serious performance challenges in most affiliate management companies.

We detected a performance pattern in banner network systems by examining the major bottlenecks in the systems' components and designed a specific solution based on our unique methodology and algorithms.

The affiliate management system of our case study consists of three components. The management system is used for business purposes like defining rules, executing payments, producing reports, etc. The second component is the real-time system responsible for recording the transactions. Both of these components run together on the web server – the IIS server. The third component is the database server that holds the management and transaction information. We worked according to the Pareto principle, also known as the 80/20 rule. This rule states that 80% of the performance issues result from 20% of the business processes.

The system accumulates the transactions information by counting every impression and recording clicks or sales on the banner directly into the database.

Banner networks billing systems process every day billions of events

Banner network impressions counting cannot be done using conventional methods.



Figure 2 - Case study system architecture



Startup Your Engines

Bottleneck Detection

The RocketTier 5 steps methodology

We started by checking the DB and the web server performance counters, using the windows system performance monitor PerfMon (it is also possible to use diagnostic products such as dynaTrace and/or Precise) and found that the DB CPU usage is 100% at peak hours.

The counting of one million events per day directly into the database brought the CPU utilization to 100%, while it needs to deal with approximately 2.5 million events per day.

We examined the queries sent to the DB using database tracing and found that 99.6% of them are validation queries to avoid forgery and update of impression queries, which meant that the validation queries had to be simplified.

Detect the suspected bottlenecks using performance counters

Solutions Rating

Based on the detection of the performance bottleneck we offered several solutions, including tuning SQL queries, fixing programming errors, stored procedures implementation and massive solutions, like in-memory databases and grid platforms implementation, which we'll discuss in the following sections.

Provide creative solutions and evaluate for each the cost/performance ratio

Immediate Effective Relief

The immediate relief boosted the system to over 2 million events per day and included tuning SQL queries, implementing stored procedures, using PHP compiler, removing includes files and fixing programming errors. At this point we doubled the system performance in just 3 months.

This performance boost wasn't enough, because the business projections of the affiliate management company required the system to support 20 million views per day by the end of 2008 and 200 million by the end of 2009, which required redesign of the way the program uses the database server, **or alternatively purchase 100 new servers.**

The 100% performance boost provided the company the time needed to design its **road map** for the major performance improvement required to keep up with its business projections.

Quick action: gain 25% performance boost in few weeks to meet immediate business needs

The Roadmap

Our first step on the roadmap was to minimize the access number to the database.



Act smart: provide an evolutionary long term road map that will boost your system in factor of 20.

Figure 3 - Impressions counting business process

The main business process occurring was validation of the banner through the database every time a visitor accesses a banner ad in order to avoid forgery. With the right software design we can avoid sending similar validation queries to the database.

The solution we used was implementing caching algorithms that keep the validation data in the application server, thereby avoiding unnecessary database querying. The validation caching algorithms reduced almost 50% of the CPU utilization.

The next step was to minimize the number of DB update queries due to impression counting, clicks or sales recording. The solution is accumulating the transactions over a period of time in an in-memory database in the application server. A cyclic process is responsible to update these accumulated transactions in the database sever and delete them from the in-memory database.

At this point the system could handle 20 million impressions per day, which meant a huge performance improvement and reaching the business projections of the company for that year using the existing hardware.



Figure 4 - The impressions business process after implementing caching mechanism reducing database access by factor of 100

However, the company still had bigger business projection of 200 million events per day for 2009, so we designed a scalable model using a scale out grid mechanism. This scale out grid mechanism includes a seventh layer load balancer and application servers. Each application server supports a different set of banners and the load balancer directs the banner impressions request to the appropriate application server.

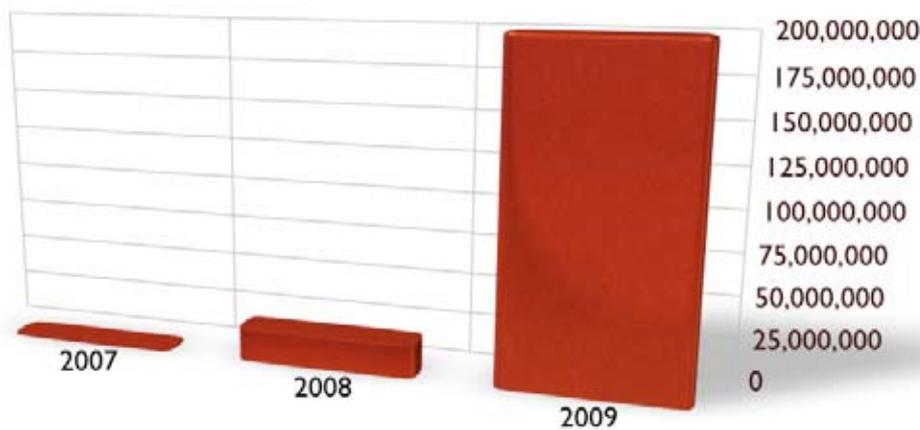


Figure 5 - Case study banner network 2007-2009 business objectives measured by number of impressions

Scale Up and Scale Out

Grid Computing allows us to manage and scale application using distributed application servers. This technology enables application growth through a single, consistent model, keeping both equity and Op-Ex low.

The designed solution supports 200 times the original capacity, which sums up to 200 million events per day. To reach the same results with a single server that is 200 times stronger would cost about 10 million USD. Our solution requires only 10 commodity servers that cost no more than 35,000 USD.

Scale out solution may achieve better performance in fraction of the cost of scale up architecture

This solution supports semi-linear growth (by simple scale out) and high availability requirements, enabling the company to meet a future target of 1 billion events per day or even more.

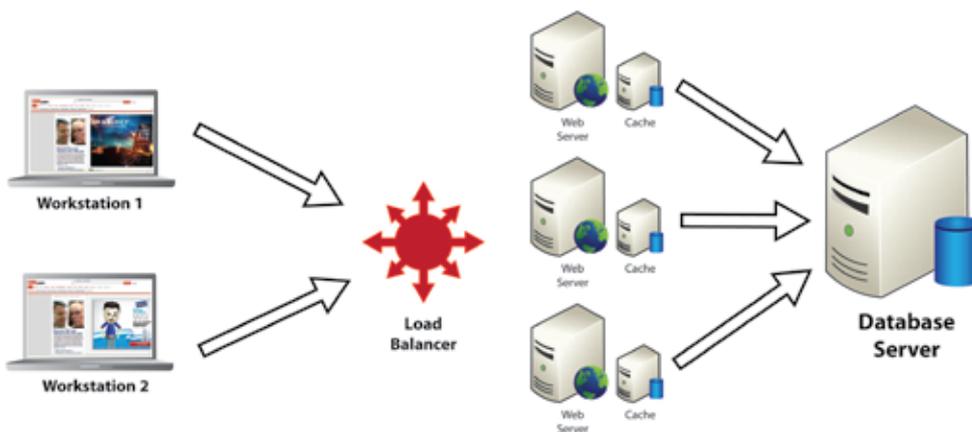


Figure 6 - Grid based architecture enabling data sharding, caching and load balancing

Bottom Line

Our solution advantages are better client experience by reduced web page load time, reduced database commit frequency as well as IT and business benefits including scaling enablement, lower hardware cost, reduced floor space, reduced CO2 emissions, reduced energy consumption. Most importantly these advantages enable our client meeting its growing business requirements in short time while maintaining the overall budget low.



Startup Your Engines

RocketTier at a Glance

RocketTier is a software solutions company, which utilizes its knowledge and skills to help companies from both the enterprise sector and the start-up industry. RocketTier has numerous success stories in solving customers' system performance bottlenecks and scale out limitations, providing immediate improvements and workarounds in a short time frame and, when necessary, redesign and implementation of the next generation solutions employing grid and/or in-memory databases in the Web 2.0, Telecom and finance markets.

About the Author

Moshe Kaplan is an expert in complex systems performance issues, scale up and scale out architectures. Mr. Kaplan is RocketTier Co-Founder and Chief Executive Officer and serves as a board member at The Israeli Association of Grid Technologies (IGT). Moshe Kaplan has over 10 years of experience in providing software solutions to the enterprise, academic and defense sectors. He is an expert in architecture, design, development and implementation of large scale software systems. Mr. Kaplan holds M.Sc. degree in Engineering (Algorithms) and B.Sc. degree in Computer Science and Physics from Tel Aviv University. Moshe Kaplan served as head of department in top IT unit at the IDF.



International Headquarters: RocketTier Technologies Ltd.

13th Floor, Avgad Tower, 5 Jabotinsky st, Ramat Gan, Israel | **Tel:** +972.54.229.1978 | **Fax:** +972.3.617.9362

Email: info@rocketier.com | **Blog:** <http://top-performance.blogspot.com> | **Web:** <http://www.rocketier.com>



All rights are reserved to RocketTier Technologies Ltd. You may not modify, copy, reproduce, republish, upload, post, transmit, or distribute, in any manner, this paper, including text, graphics, code and/or software without written permissions from RocketTier Technologies Ltd.