

SaaS BILLING & METERING





TABLE OF CONTENTS

> 1. Introduction	1
> 2. To Build In-House or Outsource	2
2.1 In-House	2
2.2 Outsourcing	2
> 3. Physical Metrics	3
3.1 Overview	3
3.2 Potential Metrics	4
3.3 Enterprise Service Bus	4
3.4 Common Tools	5
3.5 Protocol Metrics	5
> 4. Business Metrics	5
4.1 Overview	5
4.2 Alignment of Billing and Cost	5
4.3 Use Cases, Expected or Not	6
4.4 Capturing Overhead	6
> 5. Cash Flows	6
5.1 PCI DSS	6
> 6. Summary	6
> 7. Glossary of Terms	7



1. INTRODUCTION

This paper is one of a series of papers that explore and discuss the technical architectural components of the Software-as-a-Service (SaaS) model.

SaaS shares the distinction of being both a business model and an application delivery model. SaaS enables customers to utilize an application on a pay-as-you-go basis and eliminates the need to install and run the application on the customer's own hardware. Customers generally access the application via a web browser or thin client over the Internet. SaaS is most often subscription based and all ongoing support, maintenance, and upgrades are provided by the software vendor as part of the service. Application customization capabilities, if available at all, are generally provided to all customers in a consistent manner. From the perspective of the software vendor, the SaaS model provides stronger protection of its intellectual property, operational control of the environment running the software, and generally a repeatable revenue stream from the service subscription fees. Software vendors have varying capabilities and applications can come in varying flavors but SaaS applications most typically support many unique customers using a single instance of that application – also known as multi-tenancy.

Billing in the SaaS environment is unlike traditional Application Service Provider (ASP) environments. SaaS businesses are generally high transaction, dynamic businesses. Customers may each have unique packages or pricing schedules. As Service Oriented Architecture (SOA) and Web 2.0 continue their growth, SaaS business models continue to become more complex.

In general, billing is measured around usage in terms business owners can understand. For example:

- > “How many invoices were emailed this month?”
- > “How many users paid their invoice through the SaaS system this month?”
- > “How many customer employees accessed the SaaS platform this month?”

These questions are very different when looking at physical metrics of usage. For example:

- > “Per customer, how many documents are we storing and how much disk space is being consumed.”
- > “Graph bandwidth usage between each of the SaaS application tiers as well as traffic to the internet.”
- > “Graph CPU and memory load of all servers to enable the SaaS provider to provision more hardware before utilization reaches capacity.”

Physical metrics are tied to the SaaS provider's accounts payable while the billing is tied to their accounts receivable. We will address these two separate concerns as their use and mechanism of measurement can be very different. Regardless, the overlap is inevitable and in fact, should be as tight as possible.

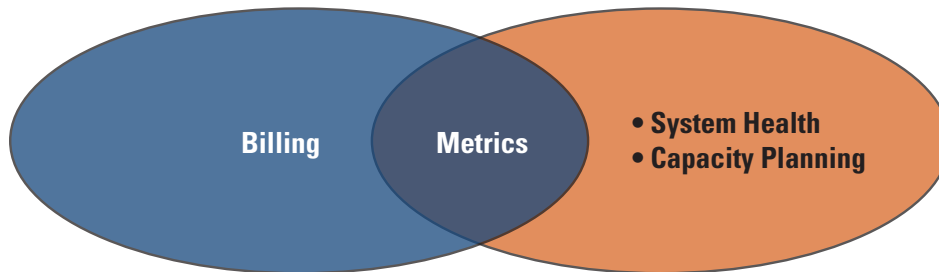


Figure 1

2. TO BUILD IN-HOUSE OR OUTSOURCE

2.1 In-House

It seems natural in an environment where many of your employees are developers to develop a billing system internally. In-house systems tend to provide a better alignment of customers resource usage to billing because they have the ability to connect to the SaaS platform at more points and collect more information. This additional information would then allow a closer alignment of expenses to billing. But it may not be that simple. In-house systems can also be lengthy development projects, expensive and risky. Systems already exist to fill this functionality and those developers might serve better to focus on the core SaaS business model.

2.2 Outsourcing

A SaaS provider has two options, build an in-house solution or outsource. There are a myriad of reasons for outsourcing this function. It allows resources to be redirected to the core platform and provides a quicker time to market. On the downside, outsourcing can also be more costly in the long run as you have the overhead of the third party to pay. It is easy enough to locate a vendor with a specialized billing system for a SaaS platform. One important point, outsourcing billing does not resolve the very real need for internal physical metering beyond just what is billed by the SaaS provider to the customer.

3. PHYSICAL METRICS

3.1 Overview

Physical metrics are important because they allow the SaaS provider to keep capacity ahead of demand and identify hotspots. Physical metrics can, and should, take place at numerous locations to allocate cost.

By examining a complex system, a SaaS provider can extrapolate possible measurement locations based on their individual implementations. Figure 2 below illustrates a 3-tiered architecture with diverse geographic locations, an external logging facility, DMZ for secure FTP, clustered databases and an iSCSI SAN. Every yellow dot represents a possible measurement point.

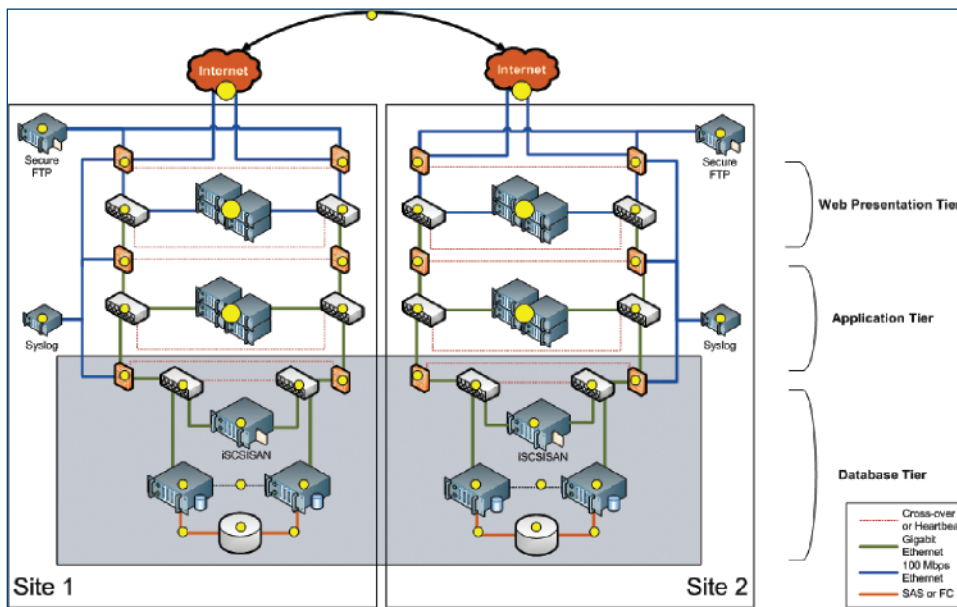


Figure 2



3.2 Potential Metrics

The following outlines some measurements to consider. If a third party provides the infrastructure, then many of these metrics will be provided by the third party:

Firewalls

- > Attacks / Threats
- > Traffic by Port
- > Traffic by Source

Routers

- > CPU load
- > Traffic by port
- > Traffic by Source

Switches

- > CPU load
- > Traffic by port

Servers

- > CPU load
- > Memory
- > Hard disk space
- > Network traffic

Database (including the server section above)

- > Request response time
- > Number of transactions/sec
- > Lock condition

3.3 Enterprise Service Bus

A common architectural paradigm for SaaS platforms is SOA. This is a natural fit as it makes integration into customer enterprise systems more fluid. Within a SOA framework is the concept of governance, metering and data interchange. These three combine in the Enterprise Service Bus. It is the glue that can be used to tie numerous systems together, connect to customers, enforce policies and enforce security. It can also save time by providing metrics of usage at numerous locations. Remember to consider your ESB when looking for measurement points.



3.4 Common Tools

There are numerous tools available for measuring the physical metrics of a system. Strongly consider using one of these tools rather than developing from scratch.

3.5 Protocol Metrics

The most common data collection protocol is SNMP (Simple Network Monitoring Protocol). Utilizing a common monitoring platform in conjunction with SNMP will allow you to monitor your firewalls, routers, switches and server's health and utilization easily. The SNMP protocol itself has a number of security flaws and special care should be taken to contain all SNMP traffic inside your firewalls. SNMP does have the ability to actively manage your devices but current best practices dictate SNMP be used in a read-only manner.

Metrics from custom applications may require that you build special hooks into your custom software. For example, you may want to know how long PDFs are taking to build. Each build could be tracked in a database and a histogram provided later on. This type of monitoring will not be handled readily by SNMP and a database specific to billing may be required.

4. BUSINESS METRICS

4.1 Overview

Clients want metrics directly related to their business and expect that the SaaS provider will be able to translate the resource consumption and cost associated with each business use case into dollars and cents. Once those use cases are labeled with monetary figures then a simplified general billing model can be produced. Obviously, in making these conversions from technical resource consumption to use case scenarios, it is necessary to make assumptions about overall volume combined with the economy of scaling, average complexity of each of the use cases and the efficiency of the SaaS provider system. These are all the concern of the SaaS provider.

4.2 Alignment of Billing and Cost

It is vital that the SaaS provider's costs and billing are as closely aligned as possible.

When building a SaaS platform, you will inevitably have users who find unexpected uses for specific functionality. If the contract between the SaaS provider and the customer is very narrow, only covering very specific events, then there may be considerable SaaS resources expended which are tied to non-billable events.



4.3 Use Cases, Expected or Not

A comprehensive use case catalog is extremely valuable to the SaaS provider. Some of this will come from the basic design or intended purpose of the SaaS solution. As the platform matures the customer will find additional “unexpected” uses or paths through the system. It is important to identify these extraneous uses and their impact on the system as they may significantly impact the SaaS provider’s expenses while working around the SaaS billing model.

Refine your contracts on a regular basis with this information to keep your expenses aligned with consumption. This is an incredibly powerful tool to enhance your product roadmap.

4.4 Capturing Overhead

Whether you are paying server engineers, collocation facilities and buying your own servers or utilizing a hosting partner, there is overhead before there is a single customer. Allocating this expense into billing requires making assumptions about customer volume. The important thing here is to build into the SaaS provider’s business model a periodic review of the allocation of overhead as the business grows.

5. CASH FLOWS

5.1 PCI DSS

Initially created by the major credit card companies, PCI DSS has become the de facto industry standard for the security around electronic billing. If credit cards or ACH are involved, it is highly likely that the settling bank will request a PCI audit be performed. PCI is a very comprehensive standard stretching from physical security to defensive coding practices. If you intend to build an in-house billing system, prepare for PCI from the start. If you intend to outsource your billing, look for a PCI certified vendor.

6. SUMMARY

Comprehensive metrics can serve numerous purposes, capacity planning, system health and billing. It is important to align the physical usage with billing in order to protect the SaaS enterprise from unexpected usage patterns. As SaaS businesses turn towards capturing their share of the value created by their presence, they need to remain extremely agile. Often times, SaaS companies will start-up with seed money, knowing they are initially going to operate at a loss. Do not be complacent about building in the framework for a proper billing system. Eventually, billing will be the life-blood of your SaaS operation.



7. GLOSSARY OF TERMS

Term	Description
SOA	Service Oriented Architecture
ESB	Enterprise Service Bus
PCI DSS	Payment Card Industry Data Security Standard



Worldwide Headquarters

Progress Software Corporation, 14 Oak Park, Bedford, MA 01730 USA
Tel: +1 781 280-4000 Fax: +1 781 280-4095
www.progress.com

For regional international office locations and contact information, please refer to www.progress.com/worldwide

© Copyright 2008 Progress Software Corporation. All rights reserved. Progress is a registered trademark of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. or other countries. Any other trademarks contained herein are the property of their respective owners.