

Foreword

On May 10, 1869 the tracks of the Union Pacific and Central Pacific Railroads were joined to create the Transcontinental Railroad. The first public railway, the Liverpool and Manchester railway, had opened less than forty years earlier on a track only thirty-five miles long. A journey from New York to San Francisco could now be completed in days rather than months.

The railroad was the Internet of its day. The benefits of fast, cheap, and reliable transport were obvious to all, but so was the challenge of building an iron road thousands of miles long through mountains and over rivers. Even though few people doubted that a railroad spanning the North American continent would eventually be completed, it took real vision and courage to make the attempt.

The railway age is often compared to the Internet since both technologies played a key role in transforming the economy and society of their day. Perhaps for this reason both began with public skepticism that rapidly changed into enthusiasm, a speculative mania, and despair. If the railway mania of 1845 had been the end of railway building, the transcontinental would never have been built.

The building of an Internet infrastructure for business applications represents the transcontinental railroad of the Internet age. Every day millions of workers sit at computers generating orders, invoices, and the like that are then sent on to yet more millions of workers who spend most of their time entering the information into yet more computers.

Ten years ago practically all businesses worked that way. Today a significant proportion of business takes place through the Web. Consumers are used to ordering

Foreword

books, clothes, and travel online and expect the improvements in customer service that this makes possible.

During the dotcom era it was fashionable to speak of “e-commerce” as if the Internet would create a completely new and separate form of commerce that would quickly replace traditional businesses. Despite the failure of many companies founded on this faulty premise, Internet commerce has never been stronger. In the peak year of the dotcom boom, 2000, VeriSign issued 275,000 SSL certificates and processed 35 million payments in transactions totaling \$1.3 billion. In 2003 VeriSign issued 390,000 certificates and processed 345 million payments in transactions totaling \$25 billion.

It is now understood that defining a class of “e-commerce” businesses is as meaningless as trying to distinguish “telephone commerce” or “fax commerce” businesses. Businesses of every size and in every sector of the economy are now using the Internet and the Web. It is no longer unusual to find a plumber or carpenter advertising their services through a Web site.

It is clear that the emerging Internet business infrastructures will eventually connect, and electronic processes that are largely automated will replace the current fax gap. It is also apparent that when this connection is finally achieved it will enable a transformation of commerce as fundamental as the railroads. Before this can happen, however, two key problems must be solved.

The first problem is complexity. Despite the many practical difficulties that had to be overcome to make online retail successful, a standard business process for mail order sales had been established for over a century. Allowing orders to be placed through a Web site rather than by mail or telephone required a modification of an established process that was already well understood. Coding systems to support business-to-business transactions is considerably more challenging than producing a system to support online retail. Business-to-business transactions vary greatly as do the internal processes that enterprises have established to support them.

The railroad engineers addressed a similar problem through standardization. An engine built to a bespoke design had to be maintained by a specialist. If a part broke it might be necessary for a replacement to be made using the original jigs in a factory a thousand miles away. The theory of interchangeable parts meant that an engine that broke down could be repaired using a part from standard stock.

Software reuse strategies that are desirable in building any application become essential when coding business-to-business applications. In addition to taking longer and costing more to build, a system that is built using bespoke techniques will be harder to administer, maintain, and extend.

Software patterns provide a ready-made template for building systems. Industry standards ensure that software will interoperate across implementation platforms.

Interoperability becomes essential when the business systems of one enterprise must talk to those of a partner.

The second challenge that must be addressed is security. A business will not risk either its reputation or its assets to an online transaction system unless it is convinced that it fully understands the risks.

It is rarely sufficient for an electronic system to provide security that is merely as good as existing systems deliver. Whether fairly or unfairly, electronic systems are invariably held to a higher standard of security than the paper processes they replace.

Today, rail is one of the safest ways to travel. This has not always been the case. Railway safety was at one time considered as intractable a problem as some consider Internet safety today. Victorian trains came off their rails with alarming frequency, bridges collapsed, cargoes caught fire, almost anything that might go wrong did. Eventually engineers began to see these accidents as failures of design rather than merely the result of unlucky chance. Safety became a central consideration in railway engineering rather than an afterthought.

The use of standardized, interchangeable parts played a significant role in the transformation of railway safety. Whether an engineer was designing a trestle for a bridge or a brake for a carriage, the use of a book of standard engineering design patterns was faster and less error prone.

The security field has long accepted the argument that a published specification that has been widely reviewed is less likely to fail than one that has had little external review. The use of standard security protocols such as SSL or SAML represents the modern day software engineering equivalent of the books of standard engineering parts of the past.

This book is timely because it addresses the major challenges facing deployment of Internet business infrastructure. Security patterns describe a means of delivering security in an application that is both repeatable and reliable. It is by adopting the principle of standardized software patterns that the engines to drive Internet business will be built.

In particular this book makes an important case for taking a proactive approach to security rather than relying on the reactive security approach common in the software industry. Most security problems are subject to a 'last mover advantage'—that is, whichever side made the last response is likely to win. Relying on the reactive approach to security cedes the initiative and thus the advantage to the attacker. A proactive approach to security is necessary to ensure that problems are solved before they become serious.

The recent upsurge in spam and spam-related frauds (so called spam-scams) show the result of relying on a reactive approach to security. By the time the system

has grown large enough to be profitably attacked, any measures intended to react to the challenge must work within the constraints set by the deployed base.

Despite the numerous applications of email and the Web, email remains at base a messaging protocol, the Web a publishing protocol. The security requirements are well understood and common to all users. The challenges faced in creating an Internet business infrastructure are considerably more complex and difficult. It is clear that every software developer must have access to the best security expertise, but how can that be possible when the best expertise is by definition a scarce resource?

Reuse of well-understood security components in an application design allows every application designer to apply the experience and knowledge of the foremost experts in the industry to control risk. In addition it allows a systematic framework to be applied, providing better predictability and reducing development cost.

Computer network architectures are moving beyond the perimeter security model. This does not mean that firewalls will disappear or that perimeter security will cease to have relevance. But at the same time what happens across the firewall between enterprises will become as important from a security perspective as what takes place within the security perimeter. It is important then that the end-to-end security model and its appropriate application are understood.

In the railway age businesses discovered that certain important security tasks such as moving money from place to place were better left to specialists. The same applies in the computer security world—it is neither necessary nor desirable for every company to connect to payments infrastructures, operate a PKI or other identity infrastructure, manage its own security infrastructure, or perform one of hundreds of other security-related tasks itself. The use of standard Web Services infrastructure makes it possible to delegate these security sensitive tasks to specialists.

The benefits of an infrastructure for business applications are now widely understood. I believe that this book will provide readers with the tools they need to build that infrastructure with security built into its fabric.

—Judy Lin
Executive Vice President, VeriSign