Lessons from History: Comparitive Case Studies

Mohsen Banan
http://mohsen.banan.1.byname.net/ContactMe

Version 0.2
First Published: August 4, 2000
Last Updated: July 7, 2000

A Component of *The LEAP Manifesto*

This article is one of a series of articles describing various aspects of the Mobile Messaging industry and the Lightweight & Efficient Application Protocols (LEAP) protocols. For the complete collection of articles see *The LEAP Manifesto* [5], available at
`http://www.LeapForum.org/LEAP/Manifesto/roadMap/index.html`. *The LEAP Manifesto* is also available at the Free Protocols Foundation website at
`http://www.FreeProtocols.org/LEAP/Manifesto/roadMap/index.html`.

Contents

List of Tables

1 Introduction

In general, protocols can have widely differing degrees of industry tenure. Some protocols achieve widespread adoption and usage, and persist as long-term industry standards. Others never achieve widespread acceptance, or else have very short lifetimes before becoming obsolete or being eclipsed by competing protocols.

There is no definitive set of circumstances which will guarantee either the success or failure of any particular protocol. Protocols succeed or fail as a result of a number of technical, cultural and business factors, each of which has complex effects and unpredictable results. The technical merits of the protocol play an important role; however, technical superiority alone is no guarantee of success.

The LEAP Manifesto is about a particular set of protocols: the Lightweight & Efficient Application Protocols, or LEAP. These protocols have been designed to address a particular industry need: the need for a set of highly efficient communications protocols. They are intended to be the foundation of an entirely new industry: the Mobile Messaging industry.

It is our hope and our intention that the protocols succeed in this purpose. It is our hope that these protocols become enduring industry standards, that they play a key role in the growth of the Mobile Messaging industry, and that they bring long-lasting benefits to the industry and the consumer.

Those factors which influence the success or failure of protocols are therefore of great interest to us. In this article, we identify and discuss those characteristics of protocols which contribute to their success or failure. In this regard we can learn a great deal from history. This article therefore also includes several case studies of successful and failed protocols.

2 Characteristics of Successful Protocols

Although there are no guarantees, history shows that successful protocols tend have certain characteristics in common. By a ”successful” protocol, we mean one which becomes accepted as an industry standard in the face of competing protocols, endures as an standard in the long term, and serves to promote the growth of the industry.

Some of the characteristics which improve a protocol’s chances are:

1. **Good Technical Design.** The more a protocol satisfies the technical requirements of the industry, the better its chances. This means that the protocol must primarily be an engineering construct, not a business one.
2. **Open Development and Maintenance Processes.** It is preferable for the protocol to be developed and maintained by open processes. This means that mechanisms should exist for public commentary on the protocol, and the protocol maintenance process should allow the participation of all constituencies that are affected by the protocol.
3. **Open Availability Process.** The protocol should be published and made available in a way that ensures that it is freely, easily and permanently accessible to anyone who wishes to use it.
4. **Freedom from Usage Restrictions.** There should be no restrictions on the use of the protocol. Anyone who wishes to base an application on the protocol should be able to do so without legal or financial hindrance.

Not all successful protocols have all these attributes. Nevertheless, the history of protocol development demonstrates that the more a protocol conforms to these attributes, the more likely it is to become an enduring industry standard.

The characteristics of various protocols are listed in Table 1. The protocols are arranged in groups, where each group represents a set of competing protocols. The table lists the major characteristics of each protocol, along with its ultimate success or failure. Note that there is only one eventual winner in each group.

Note how certain characteristics correlate well with eventual protocol success – for example, in all cases the winning protocol was free of usage restrictions. Other characteristics do not appear to influence the eventual outcome – for example, endorsement by a standards organization does not greatly increase a protocol's chances.

Use	Protocol	Usage/ Publications	Publications/ Availability	Open Source Implementation	Commercial Implementation	Development/ Maintenance	Standards Body Blessing	Long-term Success
Architecture	TCP/IP	Free	RFC / Open	Many	Many	Open	IETF (After)	Success
	SNA/DECnet	Licensed	Closed	None	Few	Closed	None	Fail
	ISO/OSI	Free	ISO/Limited	Few	Many	Limited	ISO	Fail
E-Mail	SMTP	Free	RFC / Open	Many	Many	Open	IETF (After)	Success
	X.400	Free	ITU/Limited	Few	Many	Limited	ITU	Fail
	MS:Mail CC:Mail	Licensed	Closed	None	Few	Closed	None	Fail
Security	S/MIME	Free	RFC / Open	Few	Few	Open	IETF	Fail
	PGP	Free	RFC / Open	Many	Few	Open	None	Success
WWW	HTTP/HTML	Free	RFC / Open	Many	Many	W3, Limited	None (W3 Later)	Success
Wireless	LEAP	Free	RFC / Open	Many	Few	Open	None	?
	WAP	Licensed/ Patented	Self/Limited	Few	Few (More)	Limited (WAP Forum)	None	?

Table 1: Protocol Success Stories

In the remainder of this article we provide case studies of several successful and failed protocols.

3 Case Study I: The World Wide Web

Perhaps the best known example of a network industry based on a set of open protocols is the World Wide Web. In this section we identify the key factors that contributed to the extraordinary success of the Web Industry.

Our use of the LEAP protocols to catalyze the growth of the Mobile Messaging industry is to some extent modelled on the World Wide Web. We have identified many of the key factors that led to the success of the Web, and we have implemented those same factors in the case of the LEAP protocols and the Mobile Messaging industry. In many ways, we are attempting to reproduce the same circumstances in the Mobile Messaging industry that prevailed in the Web Industry, with the expectation that this will generate similar results.

Throughout this case study, as we discuss each factor that led to the success of the Web Industry, we will identify the parallel factor that we believe will lead to the success of the Mobile Messaging industry.

The factors that made the Web successful are summarized in Table 2, along with the corresponding factors for the Mobile Messaging industry.

3.1 Prerequisites

The Web Industry required that certain technological prerequisites be in place. These included such things as wired networks and multimedia PCs. The Web Industry could not come into widespread existence until these prerequisites were in place. For this reason the Web could not have materialized any earlier than 1993.

	Web Industry	**Mobile Messaging Industry**
Prerequisites	Wired Networks, Multimedia PCs, ...	Wireless-IP Networks, Palmtop PCs, ...
Open Standards	HTTP/HTML published as RFC-1945 & RFC-1866	ESRO/EMSD published as RFC-2188 & RFC-2524
Standards Organization	W3 Consortium	EMSD.org
Client Software	Browsers on all platforms freely available	User-Agent software for Windows CE etc. freely available
Server Software	Plug & Play, Freely available	Plug & Play, Freely available, or reasonable licensing
Service Providers	mosaic.com, etc.	ByName.net, etc.
Open-Source	Apache, Netscape	Ready for Open-Source

Table 2: Web Industry vs. Mobile Messaging Industry

Similarly, the Mobile Messaging Industry has its own set of technological prerequisites. These include widespread and reliable wireless networks, affordable and miniaturized modems, and affordable and miniaturized mobile devices. All of these prerequisites did not become fully satisfied until 1999. Therefore the Mobile Messaging industry could not have materialized any earlier than 1999.

3.2 Open Protocol Specifications

The Web protocols are open and are based on the HTTP and HTML specifications, initially published as Informational RFC-1945[4] and RFC-1866[3] respectively.

Likewise, our Mobile Messaging protocols are completely open and are based on the ESRO and EMSD specifications, published as Informational RFC-2188[2] and Informational RFC-2524[1].

Hypertext multi-media information retrieval systems based on proprietary protocols existed before the Web came about. Now that HTML is widespread these proprietary protocols are all considered irrelevant.

Similary, many wireless e-mail and mobile messaging systems based on proprietary protocols exist today. We believe that they will likewise be considered irrelevant when LEAP becomes widespread.

There is another interesting parallel between the Web and Mobile Messaging protocols. The IESG inserted a critical note into the HTTP RFC cited above. More recently, the IESG has done exactly the same thing to both Mobile Messaging RFCs cited above. We regard this inclusion of critical notes into foreign (i.e. not originated by the IESG) RFCs to be a manifestation of the IESG's prejudice against external protocols – a frame of mind sometimes referred to as Non Invented Here syndrome.

3.3 Open Standards Organization

The Web Industry had the W3 Consortium. The Mobile Messaging industry has ESRO.org and EMSD.org.

3.4 Widespread Client Software

At an early stage in the evolution of the Web, client software was made widespread for all major platforms.

Similarly, a key requirement for the development of the Mobile Messaging industry will be to ensure that implementations of the LEAP protocols are made available for all major handheld devices. Specifically, for:

- PalmPilot
- Windows CE
- EPOC

We have made our implementations of device (i.e. client side) software for Windows CE and Win-32 available through Neda's website.

3.5 Widespread Server Software

At an early stage in the evolution of the Web, server software was made widespread for most major platforms.

Similarly, a key requirement for the development of the Mobile Messaging industry will be the ready availability of LEAP server software for platforms such as the Sun Solaris and Windows NT.

Our implementation of LEAP server software is readily available for the Sun Solaris platform now, and will become available for Windows NT shortly.

3.6 Open-Source Software

The availability of open-source software is extremely important for the creation of network industries.

Reference implementations of Web protocols were made widespread at an early stage in the evolution of the Web Industry.

Open-source software will play a similarly crucial role in the Mobile Messaging industry. Full implementations of published LEAP protocols will be made available as free software in open-source form. Our strategy is to release the software in stages as it becomes ready and available. The first piece to be released is our Open C Platform [6]. The next piece will be the ESRO protocol engine, followed by the EMSD device-side protocol engine, followed by the Message Center EMSD protocol engine. Further, we will integrate the LEAP protocol engines with various other free software. The distribution mechanism for the software will be through http://www.MailMeAnywhere.org.

3.7 Service Providers

At an early stage in the evolution of the Web, its industry leaders actively participated in the creation of web-based information and services. In a sense they acted as early Service Providers.

Similarly, it will be necessary to provide an initial set of Subscriber Services based on the LEAP protocols. To satisfy this need we are hosting a number of LEAP-based subscriber services.

4 Case Study II: Pretty Good Privacy

The history of the PGP (Pretty Good Privacy) data encryption system provides an excellent example of the success of a protocol developed entirely outside the traditional Standards Organization processes. PGP was essentially the creation of a single man: Phil Zimmermann. Armed with a vision and a belief in its value, Zimmermann single-handedly made PGP the dominant consumer encryption application – displacing the IETF alternatives in the process.

PGP is a protocol for electronic data encryption, which at the time of its development and deployment was in direct competition with S/MIME, a protocol being developed for the same purpose by the IETF. The key differences between PGP and S/MIME are summarized in Table **??**.

One of the major advantages that PGP enjoyed was being the creation of a single person. Small groups have an inherent advantage over large ones in any cooperative venture; as the size of a group grows, the difficulties of communication and coordination become increasingly challenging. Phil Zimmermann took this advantage to the limit: as a one-man operation, he enjoyed maximum efficiences of communication and coordination.

This is to be contrasted with S/MIME, which was developed by the IETF using classical Standards Organization processes. Under these processes, protocols are developed by committees, in the form of IETF Working Groups. Though this is a very reasonable way to conduct cooperative effort, it inevitably suffers from the friction associated with group action: communication overhead, time required to resolve misunderstandings or disagreements, and so on. Phil Zimmermann enjoyed an agility and an efficiency of action that the IETF process could not possibly match.

Both PGP and S/MIME were open protocols, without any usage restrictions, and so neither protocol had any advantage in terms of openness. Both protocols were also eventually published as RFCs. However, it is interesting to note that S/MIME, enjoying its privilege as an in-house IETF protocol, sailed through an early RFC publication process. PGP, on the other hand, was not published as an RFC until much later, when it had become clear that PGP had achieved widespread acceptance.

A further significant difference between PGP and S/MIME was the extent to which the two protocols were implemented in the form of open-source software. Both protocols were implemented as open-source; however PGP had much wider open-source support than S/MIME. This undoubtedly contributed significantly to the success of PGP, and this attests to the power of open-source in encouraging protocol acceptance.

S/MIME was developed and endorsed by the IETF, a formal Standards Organization. PGP enjoyed no such endorsement, and was developed entirely independently of any formal standards body. In spite of this, PGP has now become the *de facto* world-wide standard for electronic data encryption.

PGP is certainly not an isolated case. HTTP, the protocol which forms the basis of world-wide Internet communications, was also developed and achieved prominence independently of any formal standards body. These and many other protocols have become industry standards despite their lack of official endorsement.

The conclusions that we can draw from the history of PGP and other protocols are that standards organizations do not have an exclusive monopoly on creativity and innovation, and that official endorsement is not a prerequisite for protocol success.

The case of PGP and many other protocols supports our view that in general, innovation comes from small groups or individuals with vision, and not from committees, working groups, and Standards Organizations.

Phil Zimmermann has been an inspiration to every individual or small group with an idea they believe in, but who find themselves at odds with an entrenched Standards Organization. We believe the history of LEAP will provide similar inspiration.

References

[1] M. Banan. Neda's Efficient Mail Submission and Delivery (EMSD) Protocol Specification Version 1.3. RFC 2524 (Informational), February 1999.

[2] M. Banan, M. Taylor, and J. Cheng. AT&T/Neda's Efficient Short Remote Operations (ESRO) Protocol Specification Version 1.2. RFC 2188 (Informational), September 1997.

[3] T. Berners-Lee and D. Connolly. Hypertext Markup Language - 2.0. RFC 1866 (Historic), November 1995. Obsoleted by RFC 2854.

[4] T. Berners-Lee, R. Fielding, and H. Frystyk. Hypertext Transfer Protocol – HTTP/1.0. RFC 1945 (Informational), May 1996.

[5] Mohsen Banan. *Lightweight & Efficient Application Protocol (LEAP) Manifesto.* Technical Report 108-101-01, LEAP Forum, Bellevue, WA, January 2000. Online document is available at http://www.leapforum.org/LEAP/Manifesto/completeManifesto.

[6] Neda Public Document. *Open C Platform.* Neda Published Document 103-103-01, Neda Communications Inc, Bellevue, WA, October 1996. Online document is available at http://www.mailmeanywhere.org/sw.free/neda/foundations/ocp/OCP-MulPub/accessPage.html.