OPEN SYSTEMS IN DIGITAL CONVERGENCE[1]

Ken Krechmer

Lecturer, University of Colorado

757 Greer Road
Palo Alto, CA 94303-3024
Tel: 650-856-8836
Fax: 650-856-6591
email: krechmer@csrstds.com

ABSTRACT

The greater the degree of digital convergence the higher the potential for monopoly behavior. Open systems minimize the possibility of monopoly behavior. But what are open systems? This paper explores the different meanings and criteria suggested by the term open systems and the many ways openness is viewed. It specifically discusses open systems, open architectures, open source, open interfaces and open standards to better understand each. Identifying and agreeing upon what constitutes openness is an important step to avoid any disadvantages of digital convergence.

KEY WORDS: standards, standardization, open systems, open architectures, open source, open interfaces, open standards

INTRODUCTION

Digital convergence potentially offers many desirable attributes: easier use, lower cost, ubiquity, greater interactivity, more lifelike video and more realistic audio. The many different communications channels we use today (telephone, internet, cellular, broadcast TV, cable, satellite and wireless) may one day convergence into a single interface for all communications. However, converging all the communications of a home or office onto a single communications pathway might also decrease overall flexibility and reliability. Such a loss of flexibility and reliability could occur in a market dominated by a single supplier, where there are no alternative communications pathways.

Initially, the passage of information necessary for a free society only required freedom of the press. Later, broadcasting opened new pathways for society to receive information. Then came telephony (wired and much later wireless) communications. Next the internet emerged, providing society's first interactive information pathway. Now, digital convergence offers the possibility to bring together these independent pathways that keep society connected into a single stream, more interactive, more dynamic and more ubiquitous than any information pathway of the past. For digital convergence to be acceptable to society, these important pathways that inform, entertain and interact with us must remain open. Digital convergence only increases the importance of open systems to a free society.

As with the evolutionary hierarchy of life, the more successful communications systems are those which are the more flexible and adaptable. *Open* is usually used to describe ICT (Information and Communications Technology) systems or aspects of such systems that are perceived by the observer to be more flexible or adaptable. Of course, any understanding of *flexible* and *adaptable* is strongly skewed depending on the vantage point of the viewer. Original equipment manufacturers, software developers, end users, public standardization development organizations (SDOs) or private ones (consortia), judicial systems, national and local governments and international organizations all have their own views of what criteria constitute "openness as applied to ICT systems."

The terms open architecture, open source, open interfaces and open standards are frequently used to describe open systems. There are many cases in the literature where different criteria are related to these terms. The purpose of this chapter is to identify the criteria commonly used to describe openness, and to describe the criteria most closely relate to each of the above terms for openness. Perhaps when everyone agrees on the meaning of open systems, it will be possible to achieve them, and strengthen digital convergence

SOME DEFINITIONS

Open System embodies the terms open architecture, open source, open interfaces and open standards to support an Open Systems Environment (Grey, 1991; Hugo, 1991). Originally the IEEE standard POSIX 1003.0 defined open systems. Currently the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) standard

TR 14252 defines an Open System Environment as "the comprehensive set of interfaces, services and supporting formats, plus user aspects, for interoperability or for portability of applications, data or people, as specified by information technology standards and profiles" (Critchley and Batty, 1993). A 1994 report sponsored by the US National Science Foundation described an open data network as being open: "to users, to service providers, to network providers and to change" (NRENAISSANCE Committee, 1994). To maintain interoperability and portability, an open system needs to be responsive to change.

Open Architecture. The definition of an architecture used in ANSI/IEEE Std 1471-2000 is: "the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution." Based on this definition of architecture, two current definitions of an open architecture emerge:

- Open architecture is a type of computer architecture that allows users to upgrade their hardware in all of the computer components (for example the IBM PC has an open architecture). This is the opposite of a closed architecture, where the hardware manufacturer chooses the components, and they are not generally upgradeable (for example the AMIGA-500 home computer had a closed architecture; Wikipedia 2001-2005, http://en.wikipedia.org/wiki/Open_architecture).
- An architecture whose specifications are public. This includes officially approved standards as well as privately designed architectures whose specifications are made public by the designers (Webopedia 2001-2005, http://webopedia.com/TERM/O/open_architecture.html).

These two definitions are interrelated: the open architecture (modular system) suggested in the first requires the public specifications mentioned in the second. But is possible to have public specifications and yet a closed system, at least from the users view, if the user finds only a single vendor of the system and its components. An open architecture also supports open source software, object oriented software or open standards for hardware and software interfaces. So these two definitions of open architecture do not address all the aspects of openness.

All the aspects of open architecture represent sufficiently complex openness requirements that most modular system architectures currently are proprietary. That is, expansion or openness requires purchasing from the original vendors. This is acceptable to many users if the original vendors maintain their architecture for reasonable periods and offer acceptable migration strategies to new architectures.

Open Source describes a process of software development that makes use of multiple independent software developers. Due to copyright laws, this process requires specific arrangements of licensing that provide the legal framework to allow multiple independent software developers to change and extend the software while maintaining the software available to all (Rosen, 2005). Open Source does not require that software available to all is free.

Software development follows two major directions: proprietary and open source. Commercial software developers, the proponents of proprietary programs, state that they adapt their programs to changing user needs based on economic incentive. The more users who are willing to pay for a specific change, the more responsive the software developer will be. Open

source software developers may work for payment or work altruistically. [2] Free open source programs rely on the somewhat altruistic interest of programmers to continuously modify the program. Users then rely on the flow of changes to include the changes they desire. Many users find it more effective to pay for the changes they wish. With specific software programs (e.g. GNU Ada), free open source has been an efficient way to offer the program while commercial viability is achieved by charging for maintenance and integration needs (http://www.gnu.org/philosophy/categories.html).

Often open source systems make use of *open standards* for operating systems, interfaces or software development tools, but the purpose of open source is to support continuous software improvement (Raymond, 2002) while the purpose of open standards is to support common agreements that enable an interchange available to all. The Open Source Initiative uses the Open Source Definition (http://opensource.org/docs/definition_plain.php) to determine whether or not a software license can be considered open source. The definition was based on the Debian Free Software Guidelines, written and adapted primarily by Bruce Perens.

Under the Open Source Definition, licenses must meet ten conditions in order to be considered open source licenses:

1. Free Redistribution: the software can be freely given away or sold.

2. Source Code: the source code must either be included or freely obtainable.

3. Derived Works: redistribution of modifications must be allowed.

4. Integrity of the Author's Source Code: licenses may require that modifications are redistributed only as patches.

5. No Discrimination Against Persons or Groups: no one can be locked out.

6. No Discrimination Against Fields of Endeavor: commercial users cannot be excluded.

7. Distribution of License: The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

8. License Must Not Be Specific to a Product: the program cannot be licensed only as part of a larger distribution.

9. License Must Not Restrict Other Software: the license cannot insist that any other software it is distributed with must also be open source.

10. License Must Be Technology-Neutral: no click-wrap licenses or other medium-specific ways of accepting the license must be required (Wikipedia 2001-2005, http://en.wikipedia.org/wiki/Open_Source_Definition).

Open Standards. J. West (2004) defines "'open' for a standard as meaning rights to the standard are made available to economic actors other than the sponsor." This definition offers a succinct economic view of open standards. But economic rights cannot not be maintained without supporting political rights such as balance, consensus and due process. In order for the economic rights associated with compatibility standards to be available, some technical process (mutually agreed changes) and technical functionality (revision level negotiation) are also required. In order for specific economic rights associated with Intellectual Property (IP) to be available, specific licensing procedures must be defined.

B. Perens (no date) offers a software engineering perspective of open standards. He presents six criteria and related practices. The criteria proposed are: 1) availability, 2) maximize end-user choice, 3) no royalty, 4) no discrimination, 5) extension or subset and 6) no predatory practices. These six criteria are discussed further below.

Standards are a multi-disciplinary field. The criteria of open standards should address each of the related disciplines–economics, law, engineering, social and political sciences. From the legal perspective, each of the criteria that make up an open standard may be a legal right of a specific group. As J. West (2004) notes, each of these criteria has an economic cost and a benefit to specific stakeholders. From an engineering perspective, the criteria that create equal opportunity for manufacturers and developers directly impact communications equipment design. From a social science perspective, the dynamics of different stakeholders impacts the requirements that make up open standards. From a political science perspective, the political criteria are basic to any fair political process, including standardization.

Many see open source and open standards as complementary (Siegel & Soley, 2005). Standards are codified common agreements that enable information transfer, directly in the case of ICT standards, and indirectly in the case of all other standards. Open Source describes a process of on-going software development that often changes. Systems that combine Open Source and Open Standards suggest that the software programs evolve continuously while some interfaces remain fixed for spans of time to maintain compatibility.

The term open standards has also been associated with zero intellectual property costs (Rosen, 2005 chapter on Open Standards) by proponents of free open source software.

Other definitions: The term Standards Setting Organization (SSO) refers to any and all organizations that set, or attempt to set, what are perceived as standards. The term *SDO* refers to any SSO recognized directly or indirectly by a government. Consortium is the term used for any SSO that is not recognized directly or indirectly by a government.

The terms open interfaces and open standards are closely related. An interface is the description of the relationship between two (or more) compatible entities that are capable of communicating or interworking. Most commonly an open interface would be specified using open standards.

THREE VIEWS OF OPENNESS

The definitions given above address what these terms mean but do not indicate who views the definition. Different views of the meaning of openness affect the meaning of openness considerably. This section identifies the different criteria of openness as they relate to one or more of the interested parties–the stakeholders. Openness impacts system concepts and system development (design, development, meeting regulatory requirements and any required standardization). Openness also impacts system implementation and system use (by users).

Stakeholders may be grouped into three categories–external, implementers, and users. To identify all the requirements of openness, it is necessary to understand what these three

categories of stakeholders would consider the broadest reasonable criteria of openness. Each group of stakeholders is driven by specific desires:

- The external view of openness is driven from the government policy, legal and economic theory perspectives. From this view openness is a public good that should be monitored and protected, independent of commercial concerns.
- The implementation of openness is driven by production and distribution cost efficiencies.
- For the user, the use of standards and software that are deemed open is driven by an efficiency improvement associated with using a standardized product or system.

While there is some overlap among these desires, e.g., open meetings are more likely to create multiple procurement sources, each stakeholder group has a distinct motivation. Thus it is necessary to consider each stakeholder category independently.

The External View of Openness

The external view (e.g., a governmental view) of open systems, open architecture or open source is different from the implementer's or the user's view as it tends to focus on policy issues rather than product issues. However, the external criteria that define an open system may have significant impact on the implementer or user. An unfair political process or asymmetrical economic exchange in the creation of the systems or its interfaces may indicate some restraint of commerce. Much of the legal requirements surrounding the external dimensions are based on the need to protect the end users. In economic terms, the seller has a natural advantage by knowing more about their product than the buyer/user. The solution to this asymmetric advantage is for the user to examine multiple sources of a similar product and thereby learn enough about the product to make a fair choice. But multiple sources may be reduced when possible implementers are excluded by proprietary specifications, unfair political processes, excessive IP costs or trade barriers. Regulations that restrict the system, its uses or components may impact trade. Reduced markets may also increase user costs where trade restrictions appear. Implementers, not surprisingly, often have self-serving feelings about trade restrictions, favoring those trade restrictions that favor them, and not favoring the trade restriction that do not favor them.

It is easiest to identify the view of SDOs about open standards. Many SDOs' websites state what open or openness of standardization means to them:

The Institute of Electrical and Electronic Engineers (IEEE): "For over a century, the IEEE-SA has offered an established standards development program that features balance, openness, due process, and consensus" (http://standards.ieee.org/sa/sa-view.html).

The European Telecommunications Standardization Institute (ETSI): "The European model for telecom standardization allows for the creation of open standards..." (http://www.etsi.org/%40lis/background.htm).

The American National Standards Institute (ANSI) National Standards Strategy for the United States (2002): "The process to create these voluntary standards is guided by the Institute's cardinal principles of consensus, due process and openness ..." (http://www.ansi.org/standards_activities/overview/overview.aspx?menuid=3).

It is interesting to contrast these views with a view from the European Commission, which as a government represents the policy view (Interchange of Data Between Administrations, 2004):

The following are the minimal characteristics that a specification and its attendant documents must have in order to be considered an open standard:

- The standard is adopted and will be maintained by a not-for-profit organization, and its ongoing development occurs on the basis of an open decision-making procedure available to all interested parties (consensus or majority decision etc.).

- The standard has been published and the standard specification document is available either freely or at a nominal charge. It must be permissible to all to copy, distribute and use it for no fee or at a nominal fee.

- The intellectual property–i.e. patents possibly present–of (parts of) the standard is made irrevocably available on a royalty-free basis.

- There are no constraints on the re-use of the standard.

Most SSOs follow rules to ensure what they consider an open standards creation process by requiring open meetings, consensus and due process. Most SSOs do not suggest "the standard is made irrevocably available on a royalty-free basis" (the highest level of open IPR). Krechmer (2006) provides a more detailed view of the openness of different standards development organizations.

The Implementers View of Openness

The implementer, considered from an economic viewpoint, is within the production function of an organization that creates the open system or its components. Consider commercial developers of software, manufacturers of hardware or communications service providers as examples of implementers of open ICT systems. They want the ability to compete on an equal basis with their competitors. This concept has often been termed a *level playing field*. An implementer would describe a system as open when it is without cost to them, it serves the market they wish, does not necessarily obsolete their prior implementations, does not preclude further innovation, and does not favor a competitor. These five requirements ensure a level playing field.

As an example of the broad usage of the term open, Microsoft often refers to their implementation of Word as open, meaning that they make their implementations of Word widely available to users and the applications programming interface (API) of Word available to non-competitors (Gates, 1998). Certainly Microsoft does not plan to make the software of Microsoft Word open in a manner similar to Open Source. Commercial companies desire to have some controllable advantage over their competition. This reasonable commercial desire of an implementer to maintain a commercial advantage can become the basis for legal conflict when competitors feel that the playing field is tilted unfairly.

The purpose of the criteria open meeting, due process and consensus is to make possible a level playing field when defining an open system. But these criteria do not address two of the implementers' legitimate requirements: serves the market they wish and does not favor a

competitor. Many recognized SSOs are national or regional while many implementers' markets are international. Most recognized SSOs allow intellectual property to be included in their standards. This is not in the interests of the implementers who do not have similar intellectual property to trade.

Consortia have, in many cases, done a better job of addressing the needs of standards implementers. Many consortia offer international standardization and allow IP rights negotiation within their process (which recognized SSOs do not). Because consortia address more of the implementers' requirements, there has been an increase in standardization by consortia.

The Users View of Openness

The user is an office or factory of an organization that uses implementations of the open standard or open source software. A communications service provider may be both a user of hardware and software supplied by implementers as well as an implementer of communications services supplied to its users. The simple goal from the user's perspective is to achieve the maximum possible return on their investment. While many aspects of the users' return on investment are not related to openness (e.g., the implementation quality), four aspects are:

- The implementation operates and meets local legal requirements in all locations needed.
- New implementations desired by the user are compatible, as desired, with previously purchased implementations.
- Multiple interworking implementations from different sources are available.
- The implementation is supported over the user desired service life.

It is worth noting that users, with the exception of communications service providers, do not often participate in ICT standardization (Naemura, 1995). Perhaps that is because many of these requirements specific to users are not even considered in most SSOs.

Understanding the Requirements of Open Systems

Table 1 identifies eighteen individual criteria that relate to Open Systems. Open Systems, Open Standards and Open Interfaces potentially include all 18 criteria. Open Architecture and Open Source appear to relate to a subset of the 18 criteria. The columns indicating the criteria suggested for Open Architecture and Open Source are the author's opinion based on the definitions given above.

The political criteria provide a more detailed view of what constitutes a fair political process used to create an open system. The economic and legal criteria offer more of a governmental view of what constitutes an Open System. The implementer and user criteria are directly related to the interests of those groups (Cargill, 1997). It is interesting to see that the criteria suggested by Open Architecture and Open Source are more related to the users' and implementers' interests and less related to the process of developing the system.

Different viewpoints see the criteria of openness differently. From each viewpoint, specific criteria of openness emerge. The importance of specific criteria will even rise and fall based on the tenor of the times. Table 1 identifies five possible viewpoints and 18 related criteria. Of course the names of these 18 criteria are arbitrary and it is possible to imagine combining

different groups into one category, so there is no importance to the number 18. It appears that the legal viewpoint of openness may be expanded into far more criteria than shown. What is important is that these criteria, by whatever name, are the criteria that reasonably may be identified as part of the criteria that make up open systems.

Table 1. The Criteria of Openness for Open Systems and Open Standards.

	Criteria of openness:	Open architecture	Open source
1.	Political: A fair process:		
1.1	Open meeting		x
1.2	Consensus		
1.3	Due process		
2.	Economic: Open trade:		
2.1	No trade restrictions		
2.2	Symmetric exchanges		
3.	Legal: Fair IP costs and controls:		
3.1	Commercial licensing		
3.2	Licensing only to non competitors		
3.3	Non discriminatory licensing		x
3.4	Pass through licensing		x
3.5	Reverse engineering allowed		x
3.6	No IPR costs		x
4.	Implementer: Fair competition		
4.1	A level playing field for all competitors	x	x
4.2	Mutually agreed changes	x	x
4.3	Available documents	x	x
5.	User: Improve investment		
5.1	Multiple procurement sources	x	
5.2	Implementation assessment		
5.3	After market additions	x	x
5.4	Support over the service life		x

The first three view points (political, economic, legal) are external view points distinct from the implementer and user of an open system who are more focused on the related product or service. These three viewpoints relate most directly to the creation, design or development of an open system. The last two view points (implementer, user) are most relevant to the implementer (e.g., software developer or original equipment manufacturer) and the user of an

open system. There is some overlap of the effect of these viewpoints (e.g., a fair political process is more likely to result in multiple procurement sources).

There is considerable current interest from the legal and governmental spheres in open systems focusing on viewpoints 2 (economic) and 3 (legal). However, implementers and users are far more interested in viewpoints 4 and 5. Table 1 identifies the strong implementer and user interest in the terms open architecture and open source which more closely address their viewpoints.

UNDERSTANDING THE REQUIREMENTS OF OPENNESS

Table 1 shows that the criteria of the major stakeholder groups are sometimes similar and sometimes divergent. Users have little interest in how a standardization process is conducted. The concept that open meetings, consensus and due process support the development of multiple sources of implementations of a completed standard is recognized but rarely supported by users. Users are focused on being able to purchase compatible equipment from multiple sources. History does suggest that open meetings, consensus and due process facilitate the creation of multiple sources. In the case of the legal viewpoint, even though all the stakeholders have an interest, their interests are quite different. SSOs appear satisfied to support reasonable and non-discriminatory (RAND) IPR policies. Commercial implementers require a means to identify and control their IPR cost, which RAND does not offer. This pushes implementers to form consortia. Users are often unwilling to pay high prices caused by multiple IPR claims irrespective of which SSO (public or consortia) produced the standard. So users may be put off by the impact of RAND policies.

The first three criteria in Table 1 (1.1, 1.2, 1.3) are oriented to the stakeholders focused on open systems creation. The first four criteria (1.1, 1.2, 1.3, 2.1) are also at the heart of the World Trade Organization (WTO) Agreement on Technical Barriers to Trade, Code of Good Practice (http://www.wto.org/english/tratop_e/tbt_e/tbtagr_e.htm#Annex%203). The ANSI open standards concept requires the first three criteria for all ANSI accredited standards organizations (American National Standards Institute, 1998). The fourth criteria, no trade restrictions, is supported by ANSI but not required. Criteria 3.3, non-discriminatory licensing, has been formally added to the US standards development process by ANSI and many SSOs.

Currently the widest interest regarding open standards focuses on Fair Trade and Fair IPR. Fair Trade addresses standards as barriers to trade or enablers of trade. Fair IPR impacts the profitability of all communications equipment companies today. The additional seven criteria listed under 4 and 5 represent open standards requirements which are emerging, but are not yet supported by most SSOs. Table 1 identifies that these seven criteria are more oriented to the implementation and use of standards.

The following provides descriptions of the five viewpoints and discusses the 18 criteria identified as they relate to open standards and open source. Once these concepts are fully described, the definitions of open interface, open architecture and open systems may be refined by identifying which of the concepts apply.

1.0 Political Considerations

The political process associated with developing open standards appears quite different from the political process associated with developing open source software. Standards are developed in an SSO while open source is developed in an ad hoc group led by one or a few project leaders.[3] However, most standards development projects have just a few key contributors–similar to most open source software development projects, and consensus is usually sought among the participants of each. So underneath the different organizational veneers, the political operation is not as different, on average, as it first appears. The unencumbered licensing approach usually desired for open source software is very different from the reasonable and non-discriminatory (RAND) licensing often used by SDOs or the commercial licensing often used by consortia. Open source discussions are less fractured by specific economic goals. However, standardization meetings often have more formal rules to minimize polarization. Since open standards and open source are the concepts underlying open architecture and open systems, the political process that determines the generation of open source and open standards directly relates to how open, systems and architectures are.

1.1. Open meeting

Open Meeting–all may participate in the standards or software development process. Currently openness of meetings is deemed to be met under many SSO requirements if all current stakeholders may participate in the standards creation process. But, as technology has become more complex, user participation in standards creation has declined significantly (Foray, 1995). When the largest number of stakeholders (users) no longer participate, such a definition of open meetings is no longer functional.

In open source software development, the users of specific software tools or processes are often the developers of the same open source software. This occurs because only those with some interest are likely to become developers.

"All stakeholders can participate," is a mantra of many recognized SSOs. But this mantra does not address all the barriers to open meetings. Recent social science research has identified 27 barriers to open standards meetings and grouped these into five categories: the stakeholders themselves, the rules of recognized standardization, the way the process is carried out, the role of the technical officers of the committee, and the culture of the committees (de Vries, Feilzer and Verheul, 2004).

A major barrier to standardization and software development participation is economic. Some recognized SSOs (e.g., International Telecommunications Union, ITU) and many consortia (e.g., World Wide Web Consortium, W3C) require membership before attendance. Paying to become a standardization committee member is a significant economic barrier when a potential standardization participant is not sure they are interested in attending a single meeting. Participation expenses, unless quite low, are part of the real barriers to standardization participation for students, many users and even start-up companies in the field. Currently only a

few SSOs such as the Internet Engineering Task Force (IETF), the standardization organization for the Internet, and the IEEE offer low cost per meeting participation.

The costs to become an open source software developer are quite low (a subscription to SourceForge is $39.00 per year). Of course making the time necessary to actively participate in open source software development or standards development is a very significant cost. While some pursue such development from their own resources, the development costs are often borne by the participant's employer.

1.2. Consensus

Consensus–all interests are discussed and agreement found, no domination. Different SSO define consensus differently. Ideally, consensus requires that no single stakeholder group constitutes a majority of the membership of an SSO. This concept is less practical for open source development as the open source software developers may have a similar background. Consensus may be identified by vote of the standardization committee or may mean without active or informed opposition, what the IETF describes as "rough consensus." Surprisingly the IETF, which many find to be an example of a more open SSO, does not support full consensus as the IETF Area Directors have a dictatorial level of control over the standardization decisions in their area (IETF, 1998).

In open source software development, the project leaders often have a similar dictatorial level of control.

1.3. Due process

Due Process–balloting and an appeals process may be used to find resolution. Different SSOs describe due process differently. In general it requires that prompt consideration be given to the written views and objections of any participants. A readily available appeals mechanism for the impartial handling of procedural complaints regarding any action or inaction is part of the due process requirement. As explained above, the three requirements: Open Meetings, Consensus and Due Process, are considered fundamental by recognized SSOs to the openness of their standardization process.

Due process is less likely to available in an open source development project. It is also less available in the IETF standardization process although the IETF is evolving its standardization system.

2. Economic Considerations

From an economic viewpoint, open systems, open architecture, open standards and open source are all means to support open trade. Open trade occurs when there are no trade restrictions and each transaction is symmetric (the buyer and seller are equally well informed). Open trade is more concerned with open standards than open source software. Open source software impacts on open trade can occur where licensing restrictions or government security concerns, usually over encryption capabilities in the software, impede open trade.

2.1 Trade restrictions

No trade restrictions requires the same standard for the same capability, world-wide. This criteria is supported by the WTO to prevent technical barriers to trade. The International Federation of Standards Users (IFAN) also supports uniform international standards (IFAN, 2000). However, politically this can be a very contentious area. There are national standards for food processing that are based on religious beliefs (e.g., halal and kosher). There are standards for the environment, health, medical care, and social welfare that cause an imbalance in cost between countries that implement them (often richer) and countries that don't (often poorer). To avoid these contentious issues, most recognized SSOs currently support, but do not require, coordination of their standards work with world-wide standards. This allows, but does not favor, divergent regional or national standards.

In the richer countries, the decline of publicly funded research and aggressive commercialism has made it more difficult to achieve a single standard for a single function world-wide. The five different incompatible wireless technologies of the 3G cellular standards (W-CDMA, cdma2000, UWC-136, TD-CDMA, FD-TDMA) are an example of these effects. Initially these five 3G versions will operate in different geographic areas but eventually users will demand world-wide compatible cell phone operation. It appears likely that standardization organizations will continue to create incompatible standards for similar capabilities. This may be viewed as an indication of the failings of recognized standardization (Cargill and Bolin, 2004), or as an indication of the need to increase aspects of the support of a level playing field (see 4.1, below).

The political criteria of open standards have been addressed and in large measure resolved in most SSOs. The requirement for open trade is supported by the three recognized world wide SSOs: ISO, IEC and ITU, but many nations cling to the view that giving up their national standardization prerogatives would be giving up an aspect of their nation's sovereignty. Consortia standardization, which is unimpeded by such political issues, usually creates world-wide standards.

2.2 Symmetric exchanges

Economic transactions tend to favor the seller, who has greater knowledge of the product or service for sale. To ensure a more symmetric exchange, the buyer relies on standards that define aspects of the product or service as well as an ability to purchase from multiple competing sources (see 5.1, below). In this way open standards are quite important to enabling more symmetric exchanges.

Open Source supporters suggest that software developed and maintained by multiple individual software developers will be better understood and more available to users, but this criteria is currently neither controlled nor experimentally shown.

3. Legal Considerations

Open source software and open standards require specific licensing mechanisms to ensure that implementers and users of the software and standards are not unreasonably constrained by others' IPR.

Most recognized SSOs and many consortia consider that holders of Intellectual Property Rights (IPR) must make available their IPR for implementation on Reasonable And Non-Discriminatory (RAND) terms. IPR associated with implementations (hardware or software) has generally proved to be useful to motivate innovation. IPR on interfaces, which define a relationship between two or more implementations, is of greater concern. Five major variations of IPR and IP control on interfaces are generally considered:

1.   Commercial licensing may be the most prevalent legal way to use IPR. It is also the least open. In this case the holder of IPR and the implementer of the IPR agree privately on commercial terms and conditions for the implementer to use the holder's IPR.

J. Band (1995) described four additional levels of increasing openness relating to IPR associated with interfaces:

2. Microsoft believes that interface specifications should be proprietary, but will permit openness by licensing the specifications to firms developing attaching (but not competing) products.

3. The Computer Systems Policy Project (CSPP) also believes that interface specifications can be proprietary, but will permit openness by licensing the specifications on RAND terms for the development of products on either side of the interface.

4. The American Committee for Interoperable Systems (ACIS) believes that software interface specifications are not protectable under copyright, and that therefore reverse engineering (including disassembly) to discern those specifications does not infringe the author's copyright.

5. Sun Microsystems believes that critical National Information Infrastructure (NII) software and hardware interface specifications should receive neither copyright nor patent protection. This variation precludes commercial advantage. Using the concepts discussed under 4.1, below, some commercial advantage could be maintained. (Band, 1995)

The range of possible refinements in this contentious area is probably limitless. Some refinements include:

An additional variation of licensing, that can impact all five variations above, is pass through licensing, which allows a licensee, who is an implementer, to pass through to other implementers the licensing terms negotiated without additional actions.

The CSPP variation (3, the manner of operation of most SDOs currently) might be more acceptable to implementers if an IPR arbitration function existed when IPR is identified during the creation/modification of a standard (Shapiro, 2001).

Sun Microsystems variation (5) might be more acceptable to implementers if claims on basic interfaces were precluded but IPR on proprietary extensions was allowed. This could be technically feasible using the concepts identified in 4.1, below.

Summary of the External Criteria

The current practice of many SDOs, to require RAND licensing, appears to increase the number of consortia as it does not allow implementers to determine the impact of standards-based IPR on their costs. The semiconductor industry, where manufacturing cost drops rapidly with volume, exacerbates IPR problems for implementers. In the case of semiconductors, IPR costs may be based on fixed unit charges. Such charges can be the largest single cost component of a semiconductor. Semiconductor implementers must control their IPR costs. It seems only fair that any implementer has a right to determine the exact cost of IPR to them before they accept its inclusion in a new standard. This issue has led many implementers to join consortia as consortia often require joint licensing of related IPR. This practice defines the cost of the IPR to the implementer. While commercial licensing may seem the least open process, it may not be more costly than the RAND approach to specific implementers.

For emerging countries, RAND policies also appear to be causing an undesirable situation. China is rapidly developing into the major supplier of communications systems and equipment but Chinese companies do not have a portfolio of IPR that can be used to trade with commercial organizations in more developed countries who have IPR in communications standards. This may cause the Chinese to consider developing non-standard technologies for existing communication systems so that they do not have to pay for previous IPR standardization decisions that they did not participate in (Updegrove, 2005).

The web site Cover Pages (2005) maintains a section on open standards and collects many different descriptions of open standards. The view of open standards from the SSOs quoted on this site follows the political and legal criteria identified here. The view of other organizations is more divergent and includes criteria discussed below.

4. Implementer Considerations

The implementer of the products or services that are based on standards and the developer of software products or services that utilize open source software are affected by a number of criteria that impact the openness of the standards or open source.

4.1. Level playing field

Manufacturers or service providers must compete in order to offer multiple sources of their products and services. As was noted above, without competition the seller becomes more dominate as asymmetrical transactions become more likely. Standards represent a means to help balance the buyers' and sellers' information, but when everything about a transaction is standardized there is no longer any grounds for product competition (only price competition). In similarity standards a balance is achieved by standardizing some aspects of a product or service

but not others. Interface (compatibility) standards, in order to support a level playing field, should offer means to support proprietary advantage as well as compatibility.

Interfaces that are open, not hidden or controlled and support migration, can also support proprietary advantage. Such interfaces which exhibit both proprietary and public advantages, termed Open Interfaces, are an emerging technical concept applicable to interface standards used between programmable systems. Programmable systems with changeable memory make possible multi-mode interfaces that can be changed to support backward and forward compatibility. The idea that Open Interfaces should embody both public and private advantage is relatively new. But interest is increasing due to the considerable success of open interfaces in facsimile (T.30), telephone modems (V.8 and V.32 auto baud procedures) and Digital Subscriber Line transceivers (G.994.1 handshaking).

One way of achieving open interfaces is to implement a fairly new technique called an *etiquette* (Krechmer, 2000). Etiquettes are a mechanism to negotiate protocols. While a protocol terminates an X.200 (OSI) layer, an etiquette, which may negotiate multiple OSI layer protocols, does not terminate (replace) any protocol layer function. An etiquette is used only for negotiating which protocol, options or features to employ. The purpose of etiquettes is connectivity and expandability. Proper etiquettes provide:

- Connectivity, negotiating between two or more devices in different spatial locations to determine compatible protocols.
- Means to allow both proprietary and public enhancements to the interface that do not impact backward or forward compatibility.
- Adaptability, so that one communications system can become compatible with a different communications system.
- Easier system troubleshooting by identifying specific incompatibilities.

As long as the etiquette is common between the equipment at both ends, it is possible to receive the code identifying each protocol supported by the equipment at a remote site. Checking this code against a data base of such codes on the web or in a manual, the user can determine what change is necessary in their system or the remote system to enable compatibility.

One of the earliest etiquettes is ITU Recommendation T.30 which is used in all Group 3 facsimile machines. Part of its function includes mechanisms to interoperate with previous Group 2 facsimile machines while allowing new features (public as well as proprietary) to be added to the system without the possibility of losing backward compatibility. Another etiquette is the ITU standard V.8 which is used to select among the V.34 and higher modem modulations. More recently ITU G.994.1 provides a similar function in Digital Subscriber Line (DSL) equipment.

As an example of the usefulness of Open Interfaces, consider Microsoft APIs. Assume that a standard based upon the Microsoft Windows API is created. Then any vendor could create an operating system (OS) to work with Microsoft's applications or create applications to work with Microsoft's OS. If any vendor (including Microsoft) identified a new function such as music delivery service or IPTV that was not supported across the basic API, that vendor could then offer the new function, as an identified proprietary feature across the API, to users that purchase that vendor's OS and appropriate applications. Since an Open Interface supports proprietary

extensions (Krechmer, 2000), each vendor controls the way the new function is accessed across the API, but does not change the basic compatibility of the API. In this manner a vendor is able to maintain control and add value, based on the desirability of the new function.

Some aspects of the issue of open interfaces were explored in technical detail in 1995 (Clark, 1995). Since then, seven technical aspects of open interfaces have been identified (Krechmer, 2000).

4.2 Mutually agreed changes

To maintain openness all changes to existing standards need to be presented and agreed in a forum supporting the political criteria identified above. Controlling changes is a powerful tool to control interfaces when system updates are distributed over the internet and stored in computer memory. Even with the most liberal of IPR policies identified (3.6), Microsoft would still be able to control its Windows Application Programming Interfaces (APIs) by distributing updates (changes) to users that updated both sides of the API interface. But without a similar distribution at the same time, competing vendors' products on one side of the same API could be rendered incompatible by such a Microsoft online update.

The only way that interfaces can remain open is when all changes are presented, evaluated and approved in a committee that supports the political criteria identified above. Considering today's environment of computers connected over the internet, identifying and requiring mutually agree changes is vital to the concept of open standards. Surprisingly this is not widely understood. The original US judicial order to break-up the Microsoft PC-OS and application software monopoly did not address this key issue (United States District Court). On March 24, 2004, the European Commission (EC) announced its decision to require Microsoft to provide their browser (Explorer) independently of the Windows operating system and make the related Windows APIs available to others (European Union, 2004). This decision did not address the necessity for mutually agreed change. The EC announced on June 6, 2005 receipt of new proposals from Microsoft addressing Microsoft's support of interoperability (European Union, 2005). Unfortunately these proposals still do not directly address mutually agreed change.

Mutually agreed change is also a significant issue for the open source software development community, as the concept of open source allows independent change based on any programmer's desire. While this maximizes the freedom the programmer has to develop new applications, it is also responsible for the splintering of some open source software systems into multiple similar but not fully compatible software systems (e.g., UNIX).

4.3 Available documents

Committee documents, completed standards and software documentation should be readily available. This criteria allows any stakeholder to be able to see any documents that relate to an open standard or open source program. The openness of a standardization meeting or software development project to outsiders is closely related to the availability of the documents from the meeting. All technical documentation falls into two classes: work-in-progress documents (e.g., individual technical proposals, meeting reports), and completed documents (e.g., standards,

source code, test procedures). Different stakeholders need to access these different classes of documents. Standards implementers and software developers need access to work-in-progress documents, to understand specific technical decisions, as well as access to completed standards or source code. Implementation testers (users and their surrogates) also need access to completed documents.

The Internet Society (ISOC) supports a non-government recognized standards making organization, the IETF, which has pioneered new standards development and distribution procedures based on the internet. While the IETF does not meet the criteria for Consensus and Due Process, the IETF is perhaps the most transparent standardization organization. Using the internet, the IETF makes available on the web both its standards, termed RFCs, and the drafts of such standards at no charge. Using the facilities of the internet, IETF committee discussion and individual technical proposals related to the development of standards can be monitored by anyone and response offered. This transparent development of IETF standards has been successful enough that some other SSOs are doing something similar. In July, 1998, ETSI announced that their technical committee TIPHON (Telecommunications and Internet Protocol Harmonization Over Networks) would make available at no charge all committee documents and standards drafts.

SourceForge also uses the Internet to provide software developers a forum to access open source development projects. Currently the Open Source concept does not define the level of software documentation required: how well the source code is commented, how changes are documented and how the specific software developers are identified. Some open source development projects provide excellent documentation and others do not.

Ultimately, as technology use expands, everyone becomes a stakeholder in technology and the technical documents that describe it. Using the internet, access to documents and discussion may be opened to all. In this way, informed choices may be made about being involved in a specific committee or project, and potential new participants could evaluate their desires to participate.

5. User Considerations

The end users view of openness is critical to the success of any product or service that claims openness. However this perspective has been overlooked by many (e.g., governments) who are more concerned with how openness impacts the political and economic processes. Companies that are strongly focused on serving their markets address the users view. However such companies may be seen as not open because they may not support open political and economic processes.

5.1 Multiple procurement sources

From the users' perspective, the most significant aspect of openness is the availability of multiple sources for products and services the user desires. The importance of this from the user's perception of openness cannot be overstated. If a product or service is only available from a single supplier, the product or service is obviously proprietary. This is true irrespective of the

political process or economic criteria. From the users' view, proprietary products or services are acceptable if the cost is acceptable for the function performed. But many users recognize that their desire for after market additions (5.3 below) and support over the life of the product (5.4, below) may be compromised when products or services are only available from a single source.

5.2 Implementation assessment

A user requires the means to determine if a product or service will be suitable. Implementation assessment identifies the importance to users of reliable implementations. A user may trust an implementation based on previous performance, its brand or simply familiarity with the requirements. When such trust is not reliable, compliance, conformance and/or certification mechanisms for implementation testing, user evaluation and identification may be necessary. This more exact form of implementation assessment is termed conformity assessment. Implementation assessment is supported by ANEC (no date), a European organization that focuses on consumer issues associated with standardization. ISO/IEC 17000 defines conformity assessment as the "demonstration that specific requirements relating to a product, process, system, person or body are fulfilled" (ISO/IEC 17000). Conformity assessment procedures, such as testing, inspection and certification, offer assurance that products fulfill the requirements specified in the appropriate regulations or standards.

Implementation assessment covers all possible parameters that may need to be identified as conforming for accurate, safe and/or proper use. Such parameters could include physical access (e.g., access by people with disabilities), safety (e.g., CE or UL mark, the European and US indications that equipment is designed safely) and correct weights and measures (e.g., certification of scales and gasoline pumps). Achieving implementation assessment may be as simple as identifying a known brand or requiring testing by implementers, regulators, users or their testing agencies as well as displaying known and controlled identification marks (e.g., UL, CE) to indicate conformity to certain requirements.

Implementation assessment may represent requirements on the standardization process or software development process as well as requirements on implementations that use the standard to identify and assure compliance and, if necessary, conformance. For a manufacturer of a scale to measure weight, a self certification process traceable to national standards may be required. For a communications equipment or communications software manufacturer, an interoperability event may be needed (often termed a plug-fest) to test that different implementations interoperate. For the user, a simpler mark of conformity is often desirable. As example, in the European Union (EU), the CE marking is the manufacturer's indication that the product meets the essential (mostly safety) requirements of all relevant EU Directives. This specific marking indicating compliance reduces the user's safety concerns. Many consortia support plug-fests and compliance testing as part of their members' desire to promote associated products. Open source software is just beginning to develop implementation assessment procedures, e.g., Linux is a more user trusted UNIX version.

5.3 After Market Additions

One way the end users maximize their capital investment in products and services is to acquire extensions of such products or services that increase the service life or functionality. For the user, one advantage of products and services that dominate in their respective markets is that such products or services often attract after market additions. Smaller suppliers often complain about users being unwilling to purchase from any but the dominate supplier. But users often turn to the dominate supplier because of the expected availability of after market additions. Some suppliers (e.g., Microsoft) provide proprietary interfaces and yet support large after markets. Sometimes an interface seen as open attracts after market suppliers. But open standards and open source cannot guarantee after market additions. Successful open source software may attract continued software development, but open source software itself is no guarantee of after market additions.

5.4. Support over the service life

Products and the related software should be supported until user interest ceases rather than when implementer interest declines. On-going support of hardware, software and services and associated standards is of specific interest to end users as it may increase the life of their capital investment in equipment or software. The user's desire for implementer independent on-going support is noted by B. Perens (1999) as one of the desirable aspects of open source software. While this is true, the open source community does not provide a mechanism to ensure users that open source software will be maintained. The support of an existing standard, which directly impacts any products that utilize the standard, consists of four distinct phases after the standard is created (Table 2).

Table 2. The Phases of Support During a Standard's Lifetime.

Phase	Activity	Description	Major Interest Group
0.	Create standard	The initial task of SSOs	Creators
1.	Fixes (changes)	Rectify problems identified in initial implementations	Implementers
2.	Maintenance (changes)	Add new features and keep the standard up to date with related standards work	Users
3.	Availability (no changes)	Continue to publish, without continuing maintenance	Users
4.	Rescission	Removal of the published standard from distribution	Users

It is difficult to interest users in the first phase of standards development (creation) shown in Table 2 (Naemura, 1995). Even the second phase, fixes, may be of more interest to the developers and implementers than the users. The next three phases, however, are where users

have an interest in maintaining their investment. Currently few SSOs actively address maintaining their standards based on user desires. Possibly greater user involvement in the on-going support of standards would be practical by taking advantage of the internet to distribute standards and allowing users to track potential changes in specific standards. Increasing the users' involvement with the maintenance phases of the standardization process may also represent new economic opportunities for SSOs. The ITU-T Telecommunications Standardization Bureau Director's Ad Hoc IPR Group report, released in May 2005, includes "On-going support–maintained and supported over a long period of time" as one element of its Open Standards definition (http://www.itu.int/ITU-T/othergroups/ipr-adhoc/openstandards.html).

CONCLUSIONS

By looking at a full list of criteria that define open, it is clear that the terms open systems, open architecture, open standards, open interfaces and open source have been misapplied in the past. This accounts for much of the confusion that they engender. What a government deems to be open has only a little relation to what a user deems to be open. Commercial implementers are often caught between these two forces. In the author's opinion no SSO or open source project meets all of the criteria described. In fact, different SSOs and open source projects differ significantly in which criteria they do support.

Digital convergence offers many desirable opportunities, but it also may increase the opportunity for monopoly behavior on the part of information pathway suppliers. The increased compatibility required for digital convergence reduces the types of information pathways, creating the possibility of single vendor systems. Open systems can increase the number of information pathways by allowing competition. But achieving open systems is not straight forward and has not been well defined previously.

The 18 criteria developed here are a broad view of the meaning of openness. Are fewer criteria sufficient? That question can only be answered when each stakeholder understands the consequences of what they may be giving up. Users are advised to be alert to the pros and cons associated with digital convergence.

REFERENCES

American National Standards Institute (1998). Procedures for the Development and Coordination
    of American National Standards. Washington, DC: American National Standards Institute.

ANEC, the European consumer voice in standardization. Retrieved from http://www.anec.org/

Band, J. (1995). Competing Definitions of 'Openness' on the NII. In B. Kahin and J. Abbate
    (Eds.), *Standards Policy for Information Infrastructure*. Cambridge, MA: The MIT Press.

Cargill, C (1997). *Open Systems Standardization*. Upper Saddle River, NJ: Prentice Hall.

Cargill, C. and Bolin, S. (2004). Standardization: a Failing Paradigm, paper presented at the
    Standards and Public Policy Conference, Federal Reserve Bank of Chicago, May 13-14,
    2004.

Clark, D. C. (1995). Interoperation, Open Interfaces and Protocol Architecture. Retrieved from
    http://www.csd.uch.gr/~hy490-05/lectures/Clark_interoperation.htm

Cover Pages (2005). Technology Reports: Open Standards, October 6, 2005. Retrieved from
    http://xml.coverpages.org/openStandards.html

Critchley, T. A. and Batty, K. C. (1993). *Open Systems: The Reality*. Hertfordshire, UK: Prentice
    Hall.

de Vries, H., Feilzer, A. & Verheul, H. (2004). Removing Barriers for Participation in Formal
    Standardization. In: F. Bousquet, Y. Buntzly, H. Coenen & K. Jakobs (Eds), *EURAS
    Proceedings 2004*, 171-176. Aachen, Germany: Aachener Beiträge zur Informatik, Band
    36, Wissenschaftsverlag Mainz in Aachen.

European Union (2004). EU Commission Concludes Microsoft Investigation, Imposes Conduct
    Remedies and a Fine. March 24, 2004, 45/04. Delegation of the European Commission to
    the USA. Retrieved from http://www.eurunion.org/news/press/2004/20040045.htm

European Union (2005). Delegation of the European Commission to the USA. Retrieved from
    http://europa.eu.int/rapid/pressReleasesAction.do?reference=IP/05/673&format=HTML
    &aged=0&language=EN&guiLanguage=en

Foray, D. (1995). Coalitions and committees: how users get involved in information technology
    (IT) standardization. In R. Hawkins, R. Mansell and J. Skea (Eds.), *Standards,
    Innovation and Competitiveness*. Hants, England: Edward Elgar Publishing Limited.

Grey, P. (1991). *Open Systems a Business Strategy for the 1990s*. Blacklick, OH: McGraw-Hill.

Hugo, I. (1991). *Practical Open Systems a Guide for Managers*. Manchester, Oxford:
    NCC/Blackwell.

IETF Working Group Guidelines and Procedures, RFC 2418, September, 1998. Retrieved from
    http://www.ietf.org/rfc/rfc2418.txt

IFAN strategies and policies for 2000-2005. Retrieved from http://www.ifan-online.org/

Interchange of Data Between Administrations (IDA; 2004). European Interoperability
    Framework section 1.3 "Underlying principles," derived from the eEurope Action Plan
    2005 as well as the Decisions of the European Parliament, November, 2004. Retrieved
    from http://xml.coverpages.org/IDA-EIF-Final10.pdf

ISO/IEC 17000: 2004. *Conformity assessment–Vocabulary and general principles*. Geneva: International Organization for Standardization (ISO).

Krechmer, K. (1998). The Principles of Open Standards, *Standards Engineering*, 50(6).

Krechmer, K. (2000). The Fundamental Nature of Standards: Technical Perspective, *IEEE Communications Magazine*, 38(6), page 70.

Krechmer, K. (2006). Open Standards Requirements, *Journal of Information Technology Standards Research* (JITSR), 4(1).

Naemura, K. (1995). User involvement in the life cycles of information technology and telecommunications standards. In R. Hawkins, R. Mansell and J. Skea (Eds.), *Standards, Innovation and Competitiveness*. Hants, England: Edward Elgar Publishing Limited.

NRENAISSANCE Committee (1994). Computer and Telecommunications Board, National Research Council, *Realizing the Information Future*. Washington, DC: National Academy Press.

Perens, B. (1999). The Open Source Definition. In C. DiBona, S. Ockman, & M. Stone (Eds.), *OpenSources voices from the Open Source revolution*, 171-189. Sebastopol, CA: O'Reilly & Associates.

Perens, B. (no date). Open Standards Principles and Practice, summarized at http://xml.coverpages.org/openStandards.html

Raymond, E. S. (2000). Homesteading the Noosphere, section 2, August 25, 2000. Retrieved from http://www.csaszar.org/index.php/csaszar/interesting/the_open_source_reader

Rosen, L., (2005). *Open Source Licensing*. Upper Saddle River, NJ: Prentice Hall.

Shapiro, S. (2001). Setting Compatibility Standards Cooperation or Collusion. In R. C. Dreyfuss, D. L. Zimmerman and H. First (Eds.), *Expanding the Boundaries of Intellectual Property*. Oxford, England: Oxford University Press.

Siegel, J. & Soley, R. M. (2005). Open Source and Open Standards–Working Together for Effective Software Development and Distribution. In S. Bolin (Ed.), *The Standards Edge: Open Season*, p. 131-141. Ann Arbor, MI: Sheridan Books.

United States District Court for the District of Columbia Civil Action No. 98-1232 (TPJ).

Updegrove, A. (2004). Best Practices and Standard Setting (How the 'Pros' Do It). In S. Bolin (Ed.), *The Standards Edge, Dynamic Tension*. Ann Arbor, MI: Bolin Communications.

Updegrove, G. (2005). China, the United States and Standards, *Consortium Standards Bulletin*, IV(4). Retrieved from http://www.consortiuminfo.org/bulletins/apr05.php#editorsnote

West, J. (2004). What are Open Standards? Implications for Adoption, Competition and Policy, paper presented at the *Standards and Public Policy Conference*, Federal Reserve Bank of Chicago, May 13-14, 2004.

Webopedia (2001-2005) Webopedia, An online dictionary and search engine for computer and Internet technology definitions. Available at http://webopedia.com/

Wikipedia (2001-2005). Wikipedia, the Free Encyclopedia. Available at http://en.wikipedia.org .

ENDNOTES

---

[1] This chapter is an expansion of an earlier paper submitted to the Journal of Information Technology Standards Research (JITSR Vol. 4 No. 1). In turn the JITSR paper is a significant revision of a paper published in the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS), January, 2005. In turn, the HICSS paper is a major revision of Krechmer (1998).

[2] See http://www.slackbook.org/html/introduction-opensource.html for a Linux view.

[3] SourceForge is the world's largest development and download repository of Open Source code and applications. http://sourceforge.net/