Software Engineering Technology Watch

Robert David Cowan, *Institute for Scientific Research*
Ali Mili, *New Jersey Institute of Technology*
Hany Ammar, Alan McKendall Jr., Lin Yang, Dapeng Chen, and Terry Spencer, *West Virginia University*

The evolution of software technology is fast paced and determined by many factors. Most cannot be identified, let alone predicted, with any significant advance notice. The authors discuss their first ventures in predicting the evolution of software engineering.

Predicting the evolution of software engineering technology is, at best, a dubious proposition. You need look no further than the quotes in the sidebar "Technology Predictions That Missed the Mark" on page 124 to get a sense of how evasive technology trends can be. The recent evolution of software technology is a prime example; it is affected by many factors, which are themselves driven by a wide range of sources.

We are at the early, and tentative, stages of a project to analyze technology trends and how they evolve. We are taking a two-part approach: first structuring the problem, then specifying the methods we will use in our analysis. When we examined the problem, we found many interrelated questions that beg for answers. To focus our effort on specific issues and to lend some structure to this inherently complex problem, we built a questionnaire (for our own use) on a hierarchy of the following questions:

- How do we watch software engineering trends?
- How do we predict software engineering trends?
- How do we adapt to software engineering trends?
- How do we affect software engineering trends?

For each question, we use a judicious combination of three research methods—analytical, empirical, and experimental—or some subset thereof:

- Analytical research lets us understand the phenomena that underlie observed behavior and build models that capture these phenomena.
- Empirical research makes no attempt to understand cause-and-effect relationships but merely to capture observed behaviors by empirical laws.
- Experimental research takes place after analytical and empirical research to validate the proposed models.

Breaking this into specific questions and possible methodologies to address them has

Technology Predictions That Missed the Mark

I think there is a world market for maybe five computers.—*Thomas J. Watson, IBM, 1943*

Computers in the future may weigh no more than 1.5 tons.—*Popular Mechanics, 1949*

I have traveled the length and breadth of this country, and talked with the best people, and I can assure you that data processing is a fad that won't last out the year.—*Business editor, Prentice Hall, 1957*

There is no reason for any individual to have a computer in his home. —*Kenneth H. Olson, Digital Equipment Corp., 1977*

640K ought to be enough for anybody.—*Bill Gates, 1981*

If the automobile followed the same development as the computer, a Rolls-Royce would today cost $100, get a million miles per gallon, and explode once a year killing everyone inside.—*Robert Cringely*

helped us better understand the problem. We do not offer any concrete answers in this article, only our insights and partial solutions, and perhaps the impression that the problem is not as intractable as it might seem. Eventually, we hope to build an expert system supported by a neural network that will help managers decide whether to implement a certain technology for their specific company. We formulated some of our questions to identify the data points we would need to collect to populate a neural network; others would be the actual questions placed to the neural network.

Watching trends

The goal of watching software engineering trends means to determine what information we must gather and maintain to gain a comprehensive view of the discipline and its evolution. This information must be sufficiently rich to support discipline-wide assessments and trend-specific analysis.

Watch research questions

We formulated these questions regarding watching software engineering trends:

- What is the relevant information that we must collect or monitor?
- Where do we find this information, or where do we infer it from? (Some information might be protected by corporate interests or classified by the government, or might be unavailable altogether.)
- How do we interpret this information?
- How often do we need to update this information? (This depends on its criticality, its variability over time, and the cost of collecting it.)

Research approach

We identified a number of software engineering-specific and technology-related indicators, which we divided into seven categories:

- *Classification standings* includes sectors of performance (distribution of activity sectors) and economic data (gross domestic product, population, and labor force).
- *Research and development* includes reports on science and technology activities (including intramural and extramural expenditures, with an emphasis on software engineering).
- *Science and technology output* includes patents (divided into national applications, resident applications, and nonresident applications), publications and awards, new products or processes introduced, and amount of expenditures on software engineering products.
- *Human resources* deals with the supply side, the demand side, and plans for future supply. The supply side tracks the number of degrees granted in software engineering related fields, the demand side tracks job openings in software engineering, and we monitor education and training trends for future plans.
- *Costs and funding* monitors the sources and recipients of R&D funding, federal expenditures, and acts of alliance and cooperation.
- *Standards and regulations* deals with relevant standards that are likely to affect technology evolution (ISO 9001, IEEE standards) and relevant regulations (National Privacy Act).
- *Best practices* keeps track of software engineering best practices and of relevant aspects of the state of the art.

We are gathering this information from a number of sources. The US National Science Foundation maintains science and technology indicators, and issues scientific and technological activity reports.[1] These indicators support science and technology policy. The World Bank maintains similar data on a worldwide scale. The European Union mandates regular surveys focused on scientific and engineering innovations and publishes the results in *Community Innovation Surveys*.[2–4] Other European initiatives include the European Innovation Monitoring Surveys[5] and the House of Lords Science and Technology Reports.[6]

Rebecca Zacks' study[7] provides a basis for quantifying the economic impact of academic research by licensing income (reflecting current income collected from past patents) and current patents (representing future income potential). Zacks uses these metrics to rank the top 50 universities.

Other sources have also influenced the questionnaire, including various annual reports; the Pocket Data Book (www.nsf.gov/sbe/srs/nsf00328/pdf/nsf00328.pdf); *Investment Risk Index* (Frost and Sullivan Data Series); the International Survey of Resources Devoted to R&D; the Organisation for Economic Co-Operation and Development; the UN Statistics Office's trade statistics; the *Elsevier Yearbook of World Electronics Data*; the Harbison-Myers Human Skills Index; the United Nations Educational, Scientific, and Cultural Organization; and the US Bureau of Labor Statistics.

We have built a Web site (www.serc.net/projects/TechWatch/techwatch.htm) to record and update this information as needed.

Predicting trends

Predicting trends is probably the most important goal of this study. It is both the most crucial, on which most other goals depend, and the most difficult to achieve. We are focusing on identifying the generic life cycle that trends follow (if indeed they do). We believe it goes through three cycles: research, technology transfer, and marketing. So, to study trends, we must consider three families of trends related to these three cycles.

Research trends are driven by perceptions of the state of the art and the state of the practice, researcher perceptions of practitioner needs, national funding programs that rally around specific strategic goals, and sheer technical interest (researchers flock to areas that have meaningful technical challenge). Research trends are a favorite topic of panel sessions[8–11] and surveys.[12,13]

Technology trends are driven by the maturation of applicable research ideas and the successful evolution of an idea into a useful, technologically viable product. Technology trends are the subject of many columns in scientific and professional publications.[14–16] It is also possible to infer technical trends from technical conferences and vendor exhibits.

Market trends are created either by the supply side (when a technologically viable product becomes economically viable) or the demand side (via the creation of new markets or the expansion of existing markets). Trade publications such as the *Wall Street Journal* and *Information Week* follow market trends.

Predict research questions

Once we identify this life cycle, we can then study questions such as

- What factors determine a trend's success or failure?
- How early can such factors be assessed?
- How early can the success or failure be predicted?
- Which success factors (if any) are controllable?
- What phases or transitions in the life cycle lend themselves to external intervention?

Research approach

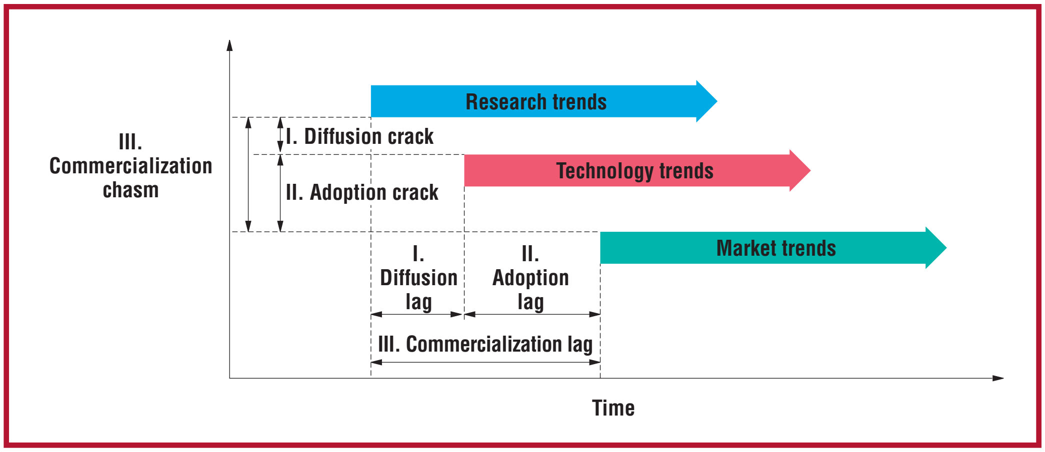
We have adopted three orthogonal research approaches to predicting technology trends: analytical, empirical, and experimental.

Using the analytical approach, we hypothesize that a software engineering trend proceeds through three phases—research, technology transfer, and marketing—and try to find models (existing models[4,5] as well as original models) for each one. We do not think a trend progresses through these phases sequentially, completing one phase before starting the next. Rather, although they start in sequence, the phases are likely to be active concurrently. We are especially interested to see what set of circumstances trigger each phase (see Figure 1).

Using the empirical approach, we observe sample trends' evolution over time, record this evolution by means of time series, then try to derive general laws for how these trends evolve. For example, if the variables X, Y, and Z capture a trend's evolution and X_H, Y_H, and Z_H represent the

Predicting trends is probably the most important goal of this study. It is both the most crucial, on which most other goals depend, and the most difficult to achieve.

Figure 1. Our model of a generic, evolutionary technology trend life cycle.



history of each trend variable, then we can write each variable as a function of the three history variables:

$$X = F_X(X_H, Y_H, Z_H)$$

The empirical approach derives functions F_X, F_Y, and F_Z using econometrics methods, mathematical curve-fitting methods, and approximation methods.

Using the experimental approach, we analyze the history of past trends and try to superimpose them on our proposed generic life cycle. The purpose of this exercise is to find some endorsement of our model (if the sample trend agrees with the proposed model structure) and to use the sample trends to understand the sequential structure of each family of trends.

The "Related Research" sidebar lists others' work on predicting trends.

Adapting to trends

Many stakeholders—corporate managers, financial planners, curriculum developers, government acquisition managers, and funding agency officers, for example—want to adapt their strategies to take advantage of evolving or emerging trends. Their questions, although varied, revolve around estimating the costs, risks, opportunities, and benefits that are attached to trends, and they want to quantify these factors to make the best decisions.

Adapt research questions

We considered the issue of adapting to a software trend from a corporate perspective: If a manager must make a decision on a given trend, what does he or she need to know about it? Several questions must be analyzed and quantified to support their decisions.

- What are the stakes of this trend for the organization? Does the trend affect the corporate business operations in a quantifiable way?
- What are the intrinsic technical merits of this trend? This issue is easy to overlook: the history of software engineering is replete with examples of good ideas that fail and incidental ideas that prosper.
- How much does it cost to adapt to the trend (upfront investment) and remain aligned with it (episodic costs)?
- What are the adoption risks of this trend? Is there internal (corporate) or external (market) resistance to the trend? Do the stakeholders (government, industry, academia, and so on) and the market support the trend?
- What are the adoption benefits of this trend? Benefits might stem from adherence to standards, access to markets, potential sales, corporate reputation (as an innovator), and so on.
- How long is the trend expected to have an impact? How long does the episodic benefit expect to accrue against the initial investment cost and the episodic cost?
- What is the optimal time to decide whether to adopt the trend? Generally, the more you wait, the more you know about the trend, but the more opportunity you miss.

We have formulated these questions so that we could quantify the answers to each by modeling the adoption decision as a return-on-investment decision. Even without

Related Research

Related work on predicting software technology trends falls into three categories, one for each family of trends. These represent the earliest sources of information used when we originally conceived this research project. These books were digested and synthesized into a rudimentary questionnaire that we refined and expanded as we started our quest for data points in earnest.

Research Trends

Everett Rogers discusses a linear, sequential life cycle for innovations, ranging from research, through development and commercialization, to diffusion and adoption.[1] He stresses the events that define transitions from each evolutionary life-cycle phase to the next, ranging from knowledge, through persuasion and decision, to implementation and confirmation. Thomas Kuhn discusses a theory of scientific (r)evolution, in which he hypothesizes how ideas arise and evolve in scientific research, how ideas compete for dominance, and what selection processes come into play to promote one idea at the expense of others.[2]

Technology Trends

We distinguish between two kinds of studies: those that deal with the general discipline of evolution, and those that deal with the evolution of specific ideas or products.

Nancy Levenson's article deals with general evolution.[3] Pearl Brereton and her colleagues extend the debate on the future of software in two ways: by stepping back from a detailed technological focus, and integrating the views of experts from a wide range of disciplines.[4] These authors report on the process they followed to sketch future research and technological directions for their company, BT Labs, and discuss the influence that their work had on their corporate strategy. Brian R. Gaines discusses his BRETAM (*b*reakthrough, *r*eplicator, *e*mpiricism, *t*heory, *a*utomation, *m*aturity) learning model and uses it to forecast the evolution of information technology (see Figures A and B).[5] However, this model does not cover the market aspects that we discuss in this article.

Samuel Redwine Jr. and William Riddle's article discusses the evolution of specific trends.[6] They report on experimental investigations that attempt to recognize the life cycle of selected technologies, emphasizing chronological aspects of each phase. The life cycle they propose includes basic research, concept formulation, development and extension, enhancement and exploration (internal, then external), and popularization. Sridhar Raghavan and Donald Chand propose an alternative, less linear life cycle.[7] They attempt to specialize Roger's framework of the diffusion of innovations to software.[1] Geoffrey Moore's work[8] is an adequate model for capturing technology trends.

Market Trends

Moore's work on shareholder value appears to be an adequate model for capturing market trends.[9] Steve McConnell's work on the Gold Rush model[10] gives further insights into market trends. Redwine and Riddle's work[6] and Moore's work[8] also give insight into market forces affecting software technology, although they primarily center on technology aspects. We are investigating these models with respect to our model with the aim of synthesizing them.

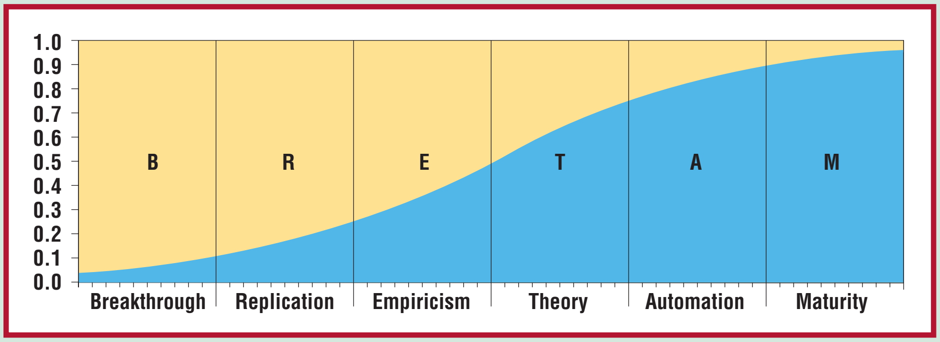


Figure A. The technological learning curve, according to Brian Gaines' BRETAM model. The y axis indicates the range from no knowledge to complete knowledge.

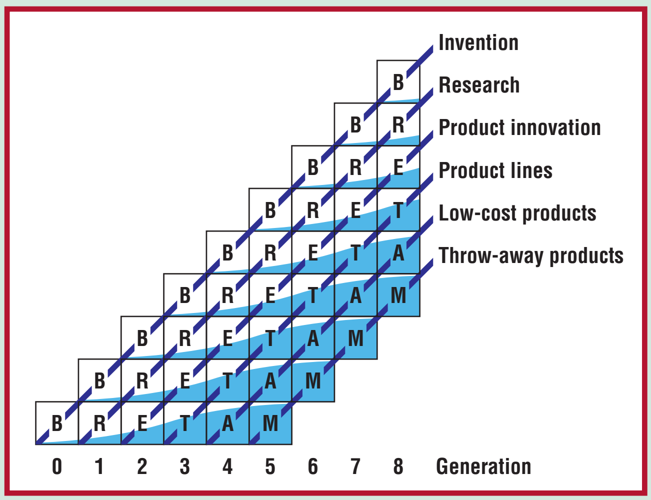


Figure B. Tiered infrastructure of learning curves based on the BRETAM model.

References

1. E.M. Rogers, *Diffusion of Innovations*, Free Press, New York, 1995.
2. T.S. Kuhn, *Structure of Scientific Revolution*, Univ. of Chicago Press, Chicago, 1996.
3. N.G. Leveson, "Software Engineering: Stretching the Limits of Complexity," *Comm. ACM*, vol. 40, no. 2, Feb. 1997, pp. 129–131.
4. P. Brereton et al., "The Future of Software," *Comm. ACM*, vol. 42, no. 12, Dec. 1999, pp. 78–84.
5. B.R. Gaines, *Modeling and Forecasting the Information Sciences*, tech. report, Univ. of Calgary, Calgary, Canada, Sept. 1995.
6. S.T. Redwine and W.E. Riddle, "Software Technology Maturation," *Proc. 8th Int'l Conf. Software Eng.*, IEEE CS Press, Los Alamitos, Calif., 1985, pp. 189–200.
7. S.A. Raghavan and D.R. Chand, "Diffusing Software Engineering Methods," *IEEE Software*, vol. 7, no. 4, July 1989, pp. 81–90.
8. G.A. Moore, *Crossing the Chasm*, Harper Business, New York, 1999.
9. G.A. Moore, *Living on the Fault Line*, Harper Business, New York, 2000.
10. S. McConnell, *After the Gold Rush*, Microsoft Press, Buffalo, N.Y., 1999.

We are interested in analyzing to what extent it is possible to affect or control technology trends in a premeditated and preplanned way.

accurate quantification, these questions are useful because they provide a comprehensive perspective on the adoption decision.

Research approach

We applied both an analytical and an empirical approach to this problem from the perspective of a corporate manager who must make an adoption decision regarding a rising technology.

The analytical approach views the adoption decision as a ROI decision. We can quantify this decision by considering the upfront investment costs, periodic costs, periodic benefits, and length of time that these periodic costs and benefits are expected to arise. Upfront investment costs include adoption costs, training costs, paradigm shifts costs, and so on. Periodic costs and quantifiable benefits make up the option balance sheet—adopting versus not adopting the technology. The decision of when to make an adoption decision can itself be modeled as a ROI decision by comparing the option of deciding immediately versus at a later date.

The empirical approach makes no effort to analyze or understand the precise economics of technology adoption, but attempts to derive relationships between relevant technology parameters and the outcome of an adoption decision. We submitted several examples (trend histories and adoption outcomes) to a machine-learning tool, and we let it discover relationships. The machine-learning tool reported on past or current trends, taken at various dates in the past—for example, the costs and benefits of adopting Ada as a company's development environment, assessed at years 1980, 1984, 1988, and 1992, and the costs and benefits of developing software for the Macintosh line of operating systems, assessed at years 1980, 1982, 1984, and 1986.

Affecting trends

We are interested in analyzing to what extent it is possible to affect or control technology trends in a premeditated and preplanned way. Launching a national research initiative in a strategic area qualifies as an attempt to control technology trends. Inventing a product (such as html) that, because of a special set of circumstances, revolutionizes the field does not.

Affect research questions

We developed these questions:

- Is it possible to affect technology trends? We want to identify controllable factors that have an impact on a trend's evolution and investigate what factors or combinations of factors can have an impact.
- Who can affect technology trends? Government organizations, standards organizations, industrial organizations, academic institutions, and industrial consortia are possible candidates.
- How can technology trends be affected? Controlling funding for research and development, affecting standards, using market clout, affecting the supply side, affecting the demand side, lifting technical bottlenecks, and lifting legislative or regulatory bottlenecks are possible responses.
- At what phase of its evolutionary life cycle can a technology trend be affected? We are pursuing analytical means (based on tentative life cycles) and empirical means (based on empirical observations of crucial junctures in the evolution of past trends) to identify phases where a trend is most likely to be affected by an outside intervention.
- How can we quantify impact? If a funding agency invests some amount of resources into a funding initiative, how can it determine, postmortem, the benefit that it has reaped from the investment? Better yet, how can it predict the amount of benefit?

Research approach

We collected data that showed the success or failure of government software engineering research initiatives. The data we used came from a variety of sources, including governmental organizations and private and public studies. We also looked at the contribution of university-based research (where a large number of government-sponsored programs are based) toward innovation in general and industrial innovation in particular.

These success-or-failure measures included

- Trivial research metrics, such as the number of scientific publications in journals and conferences. We also related these to other measures, such as the number of patents. For example, between 1985 and 1994, the number of scientific and technical papers that were cited in US patent applications rose from 0.4 to 1.4; of those, about 75 percent were written by public-sector researchers in the US or abroad. This finding might support the notion that US funding bodies' interest in applicable research is increasing.

- The number of patents issued.
- The volume of spin-off economic activity, measured by the number of spin-off companies, their volume of business, their employment figures, and so on.
- The less tangible, but no less important, advancement of scientific knowledge, which we can quantify by its long-term technological and economic dividends.

Historical observations show that research funding has a tangible impact on technology evolution, which can be quantified in economic terms. In a recent study for the NSF, CHI Research tracked more than 45,000 references from US patents to the underlying research papers, tabulating both the institutional and financial origins of the cited work. It found that more than 70 percent of the scientific papers cited on the front page of US patents came from public science—performed at universities, government labs, and other public agencies. Furthermore, they found that the papers that are cited in patents come from mainstream US science (from quite basic, relatively recent, and highly influential journals) and are authored at prestigious universities and laboratories. Also, studies have shown that for every two million US dollars in government funding, 24 articles are written, one patent is issued, and total faculty salary per institution increases by $304,030.[17]

Our research questionnaire is nearing completion, and the neural network we are building from the data seems to return adequate responses. This has allowed us to derive candidate evolutionary models (or model aspects) for the complex evolution of software engineering trends, without emphasizing an analytical explanation of the models. It has also enabled us to collect the necessary data to fill in the parameters of our candidate models and to test them for adequacy. Although we are confident in our results, they represent a small portion of the entire data set required. Much like a research vessel about to hit an iceberg, we are certain about what we can see but not so sure about the mass of knowledge below the surface.

Much like a research vessel about to hit an iceberg, we are certain about what we can see but not so sure about the mass of knowledge below the surface.

Acknowledgments

This work is funded in part by the US National Science Foundation under the Small Grant for Exploratory Research program through grant number 0086226. Any opinions, findings, conclusions, or recommendations expressed in this article are those of the authors and do not necessarily reflect the views of the NSF.

About the Authors

Robert David Cowan is a senior research member at the Institute for Software Research. His research interests are in process engineering, CMMI, and Advance Distant Learning. Formerly he was assistant director for the Software Engineering Research Center and coprincipal investigator with Ali Mili on the TechWatch project. He received his MS in information systems from the University of Pittsburgh. Contact him at the Inst. for Software Research, 1000 Technology Dr., Fairmont, WV 26554; rcowan@isr.us.

Ali Mili teaches at the College of Computing Science of the New Jersey Institute of Technology and at Rutgers University, Newark. His research interests are in software engineering, ranging from technical to managerial aspects. He received a PhD from the University of Illinois at Urbana Champaign and a Doctorat es-Sciences d'Etat from the Université Joseph Fourier de Grenbole, France. Contact him at mili@cis.njit.edu.

Hany Ammar is an assistant professor of computer engineering at West Virginia University. He is a member of the IEEE and the ACM. Contact him at the CSEE Dept., West Virginia Univ., PO Box 6109, Morgantown, WV 26508-6109; ammar@vsee.wvu.edu.

Alan McKendall Jr. is an assistant professor in the Department of Industrial and Management Systems Engineering at West Virginia University. His research involves the application of operations research methods such as forecasting, simulation, optimization, and artificial intelligence to engineering problems. He received a PhD in industrial engineering and an MS in industrial engineering and applied mathematics from the University of Missouri-Columbia. Contact him at the SERC, IMSE, PO Box 6070, West Virginia Univ., Morgantown, WV 26506; armckendall@mail.wvu.edu.

Lin Yang is a doctoral candidate at West Virginia University. Contact him at the CSEE Dept., West Virginia Univ., PO Box 6109, Morgantown, WV 26508-6109; yang@vsee.wvu.edu.

Dapeng Chen is a senior software engineer at TMC Technologies. Contact him at joluum_00@yahoo.com.

Terry Spencer is a graduate student at West Virginia University. Contact her at CBE, West Virginia Univ., PO Box 6025, Morgantown, WV 26506; tspencer@westco.net.

References

1. *Science and Technology Pocket Data Book,* Nat'l Science Foundation, Arlington, Va., 2000; www.nsf.gov/sbe/srs/nsf00328/pdf/nsf00328.pdf.
2. *Second European Report on S&T Indicators (ERSTI)*, European Commission, Gare, Luxembourg, 1998.
3. "Diversity, Convergence, Cohesion," *2nd European Report on S&T Indicators (ERSTI)*, European Commission, Gare, Luxembourg, 1997.
4. B. Clarysse and U. Muldur, *Regional Cohesion in Europe? An Analysis of How EU Public RTD Support Influences the Techno-Economic Regional Landscape*, European Commission, Gare, Luxembourg, Jan. 1999; www.cordis.lu/rtd2002/indicators/publications.htm.
5. *Empirical Studies and the Community Innovation Survey*, Science and Technology Indicators for the European Research Area, European Commission, Gare, Luxembourg, Jan. 1999; www.cordis.lu/eims/src/stud-3.htm.
6. S.T. Redwine and W.E. Riddle, "Software Technology Maturation," *Proc. 8th Int'l Conf. Software Eng.*, IEEE CS Press, Los Alamitos, Calif., 1985, pp. 189–200.
7. R. Zacks, "The TR University Research Scoreboard," *MIT Tech. Rev.*, July/Aug. 2000.
8. E.M. Rogers, *Diffusion of Innovations*, Free Press, New York, 1995.
9. M. Sitaraman, "Panel Session: Issues and Non-issues in Software Reuse," *Proc. 1999 Symp. Software Reuse*, ACM Press, New York, 1999.
10. M. Zand, "Panel Session: Software Reuse Research—Are We on Track?" *Proc. 1997 Symp. Software Reuse*, ACM Press, New York, 1997.
11. M. Zand, "Panel Session: Closing the Gap between Research and Practice," *Proc. 1999 Symp. Software Reuse*, ACM Press, New York, 1999.
12. G.A. Moore, *Crossing the Chasm*, Harper Business, New York, 1999.
13. G.A. Moore, *Living on the Fault Line*, Harper Business, New York, 2000.
14. S. McConnell, "The Art, Science, and Engineering of Software Development," *IEEE Software*, vol. 14, no. 1, Jan./Feb. 1998, pp. 120, 118–119.
15. T.G. Lewis, "Ubinet: The Ubiquitous Internet Will Be Wireless," *Computer*, vol. 32, no. 10, Oct. 1999, pp. 128, 126–127.
16. R.L. Glass, "Examining the Effects of the Application Revolution," *J. Systems and Software*, vol. 46, no. 1, Apr. 1999.
17. A.A. Payne and A. Siow, "Estimating the Effects of Federal Research Funding on Universities Using Alumni Representation on Congressional Appropriations Committees," Dec. 1998.