
Kathleen Foley Curley
Lotus Development Corporation

Marc H. Meyer
Northeastern University

Erl V. Sorensen
Bentley College

In 1990, Cusumano and Kemerer examined a statistically comparable sample of Japanese and American software development projects to assess whether significant differences could be identified in software development practices that would help to explain American dominance of the software market. This study extends the work of Cusumano and Kemerer by examining software development practices for knowledge-based systems and by including European projects as well as those from Japan, and the United States. The findings of our study confirm what Cusumano and Kemerer found and also confirm a study of quality assurance technology in Japan done by Kishida, Teramoto, Torii, Urano (1987). That is, there is no significant difference in Japanese, American and European software development practices, choice of software tools or level of project complexity. We will discuss U.S. dominance of the software market (U.S. suppliers hold more than 74% of world market share despite the fact that foreign demand now accounts for over 55% of all software revenue. U.S. Department of Commerce, 1994, pg. 27-5) and suggest that the concept of American dominance is more likely to be explained by economic and cultural factors than by differences in software development practices or access to the latest technology.

In 1990, Cusumano and Kemerer examined a statistically comparable sample of Japanese and American software development projects to assess whether significant differences could be identified in software development practices or processes and whether these differences could be used to explain American dominance of the software market. They also sought to determine if Japanese skill at process improvement could be detected and serve as a predictor of future dominance as the software market matured and process innovations became more important. (Abernathy and Utterback, 1978) Cusumano and Kemerer gathered data from 24 US and 16 Japanese software development projects across the dimensions of productivity, quality, sophistication and reuse of software code. While their findings did identify some interesting differences between the two groups, the overall conclusion of the study found no support for the belief that Japanese skills in software development were inferior to those in the US or that Japanese management of the development process was superior. Instead, the study identified several areas of remarkable similarity:

"(a) Japanese and US projects develop roughly similar products; (b) Japanese projects work with systems of at least equivalent size; (c) both use nearly identical language, tools, and hardware platforms; (d) both use personnel with comparable years of experience; and (e) at least equivalent levels of reuse, productivity, and quality in the Japanese projects suggest similar and possibly lower development costs over time for individual systems."
This study extends the work of Cusumano and Kemerer by examining software development practices for knowledge-based systems and by including European projects as well as those from Japan, the United States, and by updating analysis of trends in the worldwide software market. Our findings support the work of Cusumano and Kemerer and reveal that there are substantial similarities in project complexity, development costs and duration across all three of the national groups that we studied. While our study also reveals some interesting differences in specific areas such as choice of hardware platforms, our data show no significant differences in management technique or technical sophistication.

We will suggest that the concept of American dominance in certain segments of the software industry, notably packaged software, is not explained by management practices or access to technology. There is evidence that the technology transfer life cycle is short; with educated people gaining access to the latest information and sharing it quickly through global communications networks and air travel. Additionally, the definition of what is "American" is increasingly blurred and much harder to define as American companies hire foreign nationals to do contract programming and product development. (Computerworld, November 1, 1993) We will suggest that there are other influences such as access to venture capital, presence of local competition and finally national culture and values that should be examined in future research to gain a better understanding of the phenomenon of national dominance in certain industries.

The Sample

Over the past five years, two of the authors conducted field research with major American, European and Japanese companies to explore their selection and development of new, emerging information technologies. The majority of that work examined knowledge-based systems applications (see Meyer and Curley, 1991) and gathered data regarding the origins of systems ideas, development costs, project duration, management controls, and the composition of the software development teams.

We first sampled firms in the United States and England. Our structured questionnaire was translated into German by the Fraunhauser Institute, and administered to firms in Europe. The research instrument was translated into Japanese by the Nikkei Intelligent Systems Journal and administered to more than 70 Japanese firms. We then translated their responses into English with the help of Japanese speaking graduate students. A broad range of industries is represented: financial services, engineering, transportation, sales, energy, health services, construction, and government.

The study was based upon a sample of 134 "operational" information systems. The definition of an operational system is one that has passed the design, prototyping, and testing stages, and had been fielded as a working application and used within an organization (and possibly by its suppliers, distributors, or customers) for the system's intended purpose for at least one year.

Since there was strong support in the literature that effective management of software development was contingent upon the embodied complexity of the system, we developed measures to assess and categorize systems in terms of two dimensions of complexity: decision making or knowledge complexity and systems or technology complexity. Knowledge complexity is defined as the degree of depth and specialization of the internalized knowledge of human experts, the scope of the decision-making process (Clancy, 1985) and the level of expertise required (Dreyfus, 1986) including discipline-based knowledge, that is incorporated into the expert system application. Technological complexity, on the other hand is defined as the depth and scope of the programming effort, the user environment (Prerau, 1985) and related technical efforts involved in building such systems and in implementing them in production environments.

Conceptually, the combination of the knowledge and technological dimensions produces four generic types of expert systems applications. The groupings are based on the four combinations of "high" and "low" pairings of embodied complexity. Figure 1 shows these dimensions of complexity and our own generic labels for the types of systems they give rise to. Four actual cases from our research sample illustrate each quadrant in that figure.

- Personal Productivity Systems: a stand-alone PC-based application made by a large European airline company requires only a handful of "rules" to effectively assess the tax implications of employee travel from foreign countries back home to England.
- Knowledge Intensive Systems: a stand-alone PC-based application developed by a large engineering services firm to provide decision assistance for scheduling the operations of a large chemical plant. This system has logic that incorporates supplier management and logistics, manufacturing cost and quality, and production scheduling.
- Technology Intensive Systems: a sales force information system made by a computer manufacturer automatically downloads new product and component information to its global sales personnel.
- Strategic Systems: an underwriting system for a large insurer that combines medical, underwriting, and actuarial knowledge in a highly complex system integrated with mainframe administrative systems for the sharing of data and pricing of underwritten cases based on specific insurance products.

Our initial efforts to develop measures for classifying a particular system along these two dimensions of complexity were made through a pilot study that involved intensive discussion with software managers at ten large firms. The data gathering process involved the completion of interview forms by a number of key individuals associated with the respective systems: project managers, domain experts knowledge engineers, and key computer programmers. The depth of the interviewing helped to insure the reliability of the data collected. The majority of the data gathered were concerned with objective and verifiable information.