# Book review

## Japan's software factories: a challenge to US management

Michael A. Cusumano, Oxford University Press,
New York, 1991
ISBN 0195062167. 512 p. £27.50

This well researched book is of interest because it traces how the Japanese are able to produce some types of software more productively and to higher levels of quality than the West. Dr Cusumano is based at the Massachusetts Institute of Technology (MIT) Sloan School of Management. His knowledge of Japanese allows him access to material that is not available in English. The approach used is that of a business historian and this means that the book is an analysis of the development of software production strategies and practices.

The central theme is that as software systems increase in size and complexity, *ad hoc* approaches to managing product development seem to become inadequate. This creates opportunities for some developers to become more structured and systematic, in other words to become able to use the techniques of the factory. Cusumano sees the project-centred, loosely structured organization as a transitional phase before adopting a factory process.

The body of the book is five case studies or histories, of one early (1955–81) American attempt at a software factory – System Development Corporation, and four Japanese software producers – Hitachi (1969–89), Toshiba (1975–89), NEC (1955–88) and Fujitsu (1960–89). The data that forms these five studies shows that they all encountered the same problems. The Americans abandoned the factory approach of standardization and reuse within three years. The Japanese persisted, although in all four companies improvements took several years to materialize.

The evidence provided from the four Japanese companies demonstrates convincingly that their levels of defects, productivity and software task estimating have been improving steadily over 10–15 years. This patient and methodical approach with its constant measuring of performance and analysis of the production process, is in stark contrast to the chaotic fever found in many Western software organizations.

The power of the Japanese approach is shown with clarity in a three page appendix entitled 'Japanese and US Project Performance'. This reports research published in November 1990, on the comparative performance of 21 leading American and 15 Japanese software producers. It concludes that the Japanese are at least equal and probably ahead in the areas of productivity, failure rates and levels of reuse.

The Japanese use the 'waterfall' approach that means that the process of developing software proceeds through a sequence of analysis, specification, design, code and test. The 'waterfall' is often criticized in the West for being too slow and many 'rapid' alternatives are being advocated. The Japanese put their energies into optimizing the 'waterfall' process, realizing that a change of methodology would create a much larger learning curve. They also used standard languages, such as COBOL and are currently attempting to use the UNIX operating system widely. The picture painted is of a relatively low technology, conservative approach; with the efforts focused on refining an exceptionally disciplined and closely monitored construction process.

The differences in the market place between Japan and the West have driven their respective software industries in different directions. The American military and corporations prefer either large bespoke systems or cheap standard packages. Software of both types requires job shop or project management techniques for its construction. Conversely the Japanese corporations tend to seek standard systems but with tailoring to suit their particular needs. This pattern of market demand has encouraged a factory style production with standardization and the re-use of components.

The Japanese focus on the process of production, the West focuses on the product, this is the root of the differences in management. Because of this the Japanese will select markets and strategies that will give them time to perfect their production skills.

It is unfortunate that the American company Microsoft was not chosen for comparison with the Japanese. It does use the project-centred, loosely structured approach which Cusumano feels is not as powerful as the factory method. It also dominates the personal computer software industry.

In conclusion this is a comprehensively referenced and indexed book. It takcles a topic that has been under explored because of the difficulty of getting translations of much Japanese material. My only reservation with this professional piece of work, is that the writing style is slightly opaque. But it does clearly establish that software is just like any other industry, and that management is the key.

Peter Middleton
The Queen's University, Belfast