Our objective is to provide a forum for sharing problems, experiences, and solutions that can help establish bridges among hardware engineers, software engineers, systems engineers, and project managers. To this end, we plan to present articles encompassing a broad spectrum of applications (both large and small) and application domains. We will seek to publicize innovative solutions and engineering methodologies that address

- complexity,
- model-based engineering (as exemplified by hardware/software codesign), and
- process management.

We believe these three areas are critical to advancing the successful, integrated engineering of computer-based systems.

MODEL-BASED ENGINEERING

Current practice dictates the separation of the hardware and software development paths early in the design cycle, with very little interaction between them until system integration. Because integration occurs late in the process, any problems encountered are costly. For example, the premature selection of unacceptably slow processors may require that the software attempt to correct the inadequacy. Conversely, poor software performance may necessitate the development of special-purpose hardware.
Hardware/software codesign attempts to integrate the hardware and software development paths, providing a more synergistic approach to system design. Using a codesign environment, engineers can determine the best mixture of hardware and software within a particular system.

Ideally, an integrated codesign environment supports hardware/software evaluation throughout the development process using models with various levels of detail. Appropriate abstractions are crucial. Abstractions let us focus on aspects of interest and thus manage complexity. More detailed descriptions of critical portions of the system let us assess risk. Using models, we can analyze the consequences of different design decisions within the system as a whole, gradually refining the system description into a hardware/software implementation—an idea called model continuity. Model continuity allows for the continuous and incremental evaluation of the system. Important issues include determining the amount of detail required within a model to obtain "reasonable" results and supporting hardware/software modeling at different levels of detail.

We advocate tools and techniques that foster the integration of the hardware and software perspectives. An important means of achieving this integration is through unified representations. Unified representations can be used to model a system independent of its implementation in hardware or software, or to model both hardware and software for combined evaluation. A number of modeling representations and formalisms have been proposed in the literature and applied to computer-based systems. They include dataflow diagrams, finite state machines, Petri nets, specialized algebras, and object-oriented representations.

Our position is that these formal specification techniques have limited utility without a systematic modeling methodology to guide their use. We will seek to present such modeling methodologies in this department, focusing on integrated design environments that use unified representations.

**PROCESS MANAGEMENT**

It is very important to capture the engineering process, its formalization, and its management. Well-structured domains such as chemical and civil engineering have established engineering foundations and principles. But even in these disciplines, methodological problems and errors can occur as processes and projects become more complex. They may stem from attempts to solve imprecisely stated problems or from improper or inadequate use of existing technologies. Most often, however, they result from the lack of an underlying engineering process methodology and personnel specifically trained to manage this process.

This problem is especially acute in the engineering of complex computer-based systems. Factors contributing to the complexity of the engineering process are the multitude of different principles, techniques, methods, and tools employed by designers and engineers, and the relentless introduction of new technologies that must be harnessed to support the management of such processes—for example, high-speed computer networks, standard data exchange formats, and multimedia.

However, these new technologies can facilitate a positive change in the work style and team approaches, as demonstrated by the new paradigms of collaborative, distributed design techniques. We will seek to show how new approaches can harness the available technology to handle the complexity of large-scale projects; to better support concurrent, collaborative, and distributed development of systems; and to automate or partly automate routine engineering activities. We will go beyond traditional approaches stemming from the operations research and computer science disciplines that represent, plan, and optimize project engineering. We will focus on a new paradigm, process modeling, and examine how it can be extended from the software domain to integrated engineering.

We feel that process modeling

- adequately captures multiperson design activities with a high degree of cooperation, distribution, and coordination of various tasks;
- reflects a computer-based systems development process that is evolutionary in nature; and
- provides a means to rapidly accommodate new design paradigms as new technologies emerge.

In the next five columns, we will be looking at these issues in greater detail. We will seek representation from researchers and practitioners from a broad spectrum who possess the expertise needed to build complex heterogeneous systems. We welcome your participation.

**For More Information**


