06

N91-18618

RICIS RESEARCH

Computer Systems and Software Engineering

Charles W. McKay, Director, High Technology Laboratory and Software Engineering Research Center, Professor of Computer Science, UH-Clear Lake

Artificial Intelligence and Expert Systems

Terry Feagin, Professor of Computer Science, UH-Clear Lake

Information Systems

Peter C. Bishop, Director, Space Business Information Center, Associate Professor of Human Sciences, UH-Clear Lake

Mathematical and Statistical Analysis

Cecil R. Hallum, Associate Professor of Mathematics, UH-Clear Lake

Education and Training

Glenn B. Freedman, Director, Center for Cognition and Instruction, Associate Professor of Reading and Language Arts, UH-Clear Lake

An Overview of the Computer Systems and Software Engineering Component of RICIS

Charles W. McKay

The principal focus of this RICIS component is computer systems and software engineering in-the-large of the lifecycle of large, complex, distributed systems which:

* evolve incrementally over a long life,

* contain non-stop components, and

* must simultaneously satisfy a prioritized balance of mission and safety critical requirements for behavior at run time

This focus is believed to be extremely important at this time because of the contribution of the "scaling direction problem" to the current software crisis. That is, paradigms/models, techniques/methodologies and tools which often worked for yesterday's comparatively smaller, simpler, centralized systems have been shown to be an inadequate baseline to scale-up to meet the challenges of distributed systems. By contrast, models, methodologies, tools, and environments which are based on a sounder theoretical foundation to address these larger and more complex systems are capable of scaling-down to meet the needs of less demanding, centralized applications.

As shown in Figure 1, the Computer Systems and Software Engineering (CSSE) component addresses the lifecycle issues of three environments-host, integration and target. Solutions are proposed, specified, designed, developed, verified and sustained in the host environment. The solutions are deployed, monitored, interactively queried and operated int he target environment. Increasingly, components of both the host environment and the target environment are geographically as well as locally distributed. The solutions from the host environment are moved into the target environment under the control of the integration environment. The integration environment is responsible for monitoring and sustaining the current baseline of software, hardware and operational components in the target environment. The integration environment is also responsible for the test plans and for controlling the integration and evolution of advancing the target environment to the next baseline. Emergency interactions are also controlled through the integration environment.

As Shown in Figure 2, an integrated lifecycle support environment is becoming the common interface to four principal engineering activities: computer systems engineering, software engineering, hardware engineering, and the management of operations and logistics.

Currently there are more than 18 funded research activities in this technical area. Additionally, there is a larger number of CSSE coordinated research projects which are funded by companies working with the university. Several of these activities are deliberately structured interfaces to the other four components of RICIS.

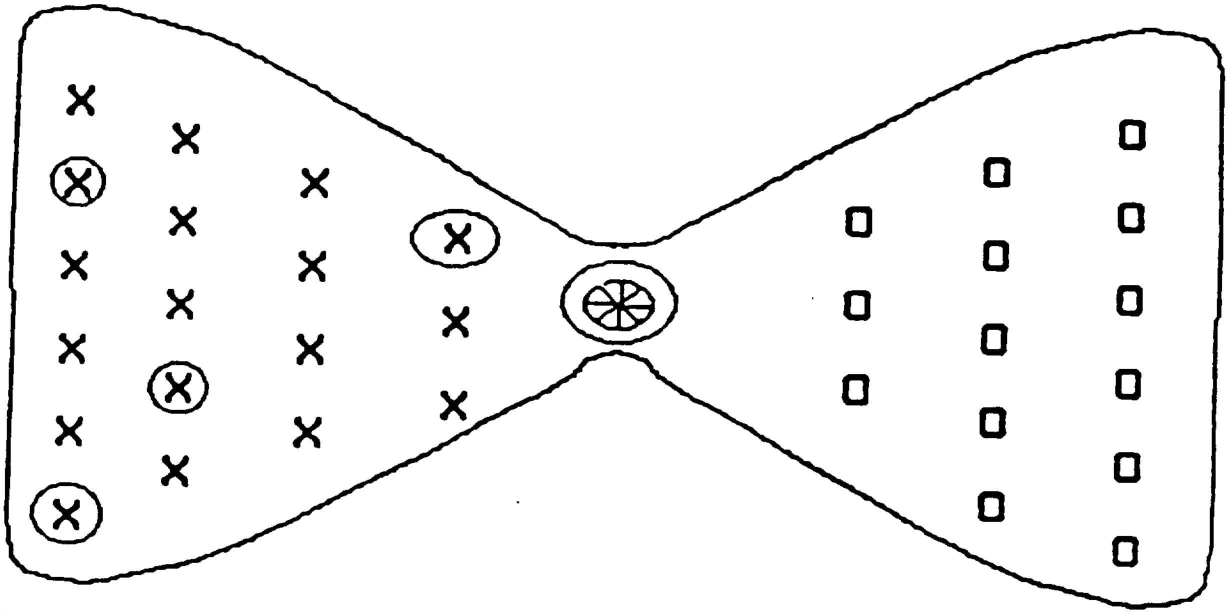
The goals for CSSE research during the next three years may be summarized as:

1. To develop a position of international leadership in the engineering of mission and safety critical components for the target and integration environments of large, complex non-stop, distributed systems.

2. To sustain a position of international leadership in the research issues of the host environment for the above applications.

3. To augment the Computer Systems and Software Engineering research base and provide support as needed to the other four technical areas addressed in RICIS.

TWO SCENARIOS FOR SSP ENVIRONMENT IN 2000+ A.D.



HOST ENVIRONMENTS:

- DEVELOP
- SUSTAIN

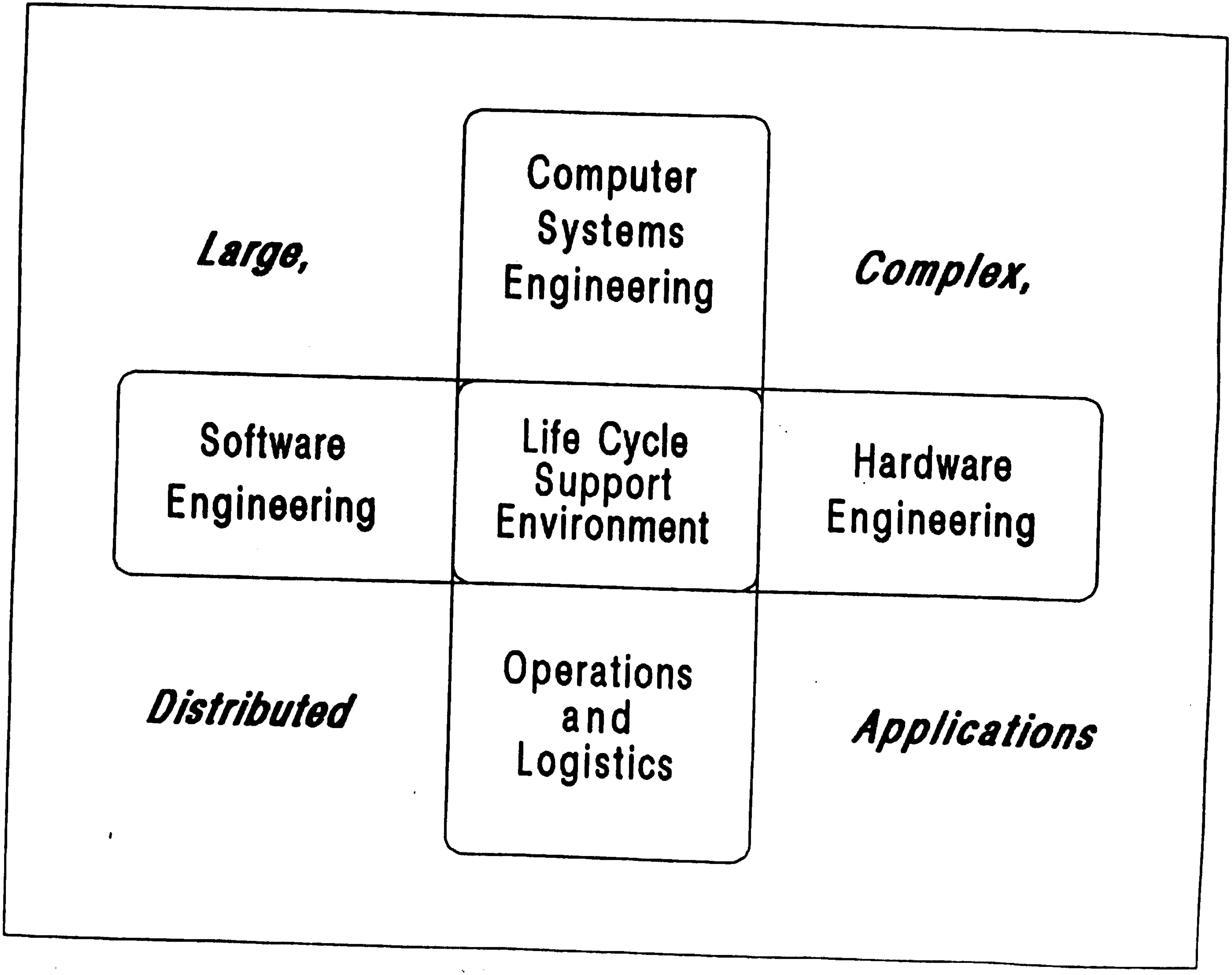
INTEGRATION ENVIRONMENT:

- CONTROL OF TGT. ENVIR. BASELINE
- INTEGRATION V&V FOR NEXT BASELINE AND TEST & INTEGRATION PLANS

TARGET ENVIRONMENTS:

- DEPLOY
- OPERATE

Figure 1



Large,

Complex,

Computer Systems Engineering

Software Engineering

Life Cycle Support Environment

Hardware Engineering

Operations and Logistics

Distributed

Applications