PROCEEDINGS
of
The Seventeenth Annual Conference on Information Sciences and Systems

Department of Electrical Engineering and Computer Science
The Johns Hopkins University
Baltimore, Maryland 21218

PROCEEDINGS

OF

THE SEVENTEENTH ANNUAL

CONFERENCE

ON

INFORMATION SCIENCES AND SYSTEMS

PAPERS PRESENTED

March 23, 24 and 25, 1983

PROGRAM DIRECTORS

Howard L. Weinert
Robert C. Melville

Department of Electrical Engineering and Computer Science
The Johns Hopkins University
Baltimore, Maryland 21218

DUALITY AND THE DESIGN OF CONCURRENT PROGRAMS

Michael A. Stanton
Departamento de Informática
Pontifícia Universidade Católica do Rio de Janeiro
22453 Rio de Janeiro, RJ
Brazil

In his book, "The Architecture of Concurrent Programs" [BH77, p.192], Brinch Hansen discussed the problem of designing concurrent programs, and proposed a methodology for determining an ideal program structure, in terms of simultaneous activities and data structures. More recently Gentleman [GEN81] stressed the importance of identifying limited and useful forms of structuring concurrent programs, such that these forms might be routinely used as building blocks in the assembly of larger structures. Gentleman went on to emphasize the important role played in the design of concurrent programs by an anthropomorphic point of view, which establishes analogies between sets of asynchronous processors and organizations of human beings. These analogies help the designer to choose which activities should be concurrent, and what is the nature of their interactions in a concurrent program.

Besides the discussions about the abstract structure of concurrent programs, there are also disagreements about how this structure should be implemented. There are two main models in current use, which combine the concepts of parallelism, synchronization and communication, and which can be said to define a programming style. The better known model employs monitors to control access to shared variables, whilst the other model involves the explicit transfer of data in the form of message passing [GEN81]. This latter model holds obvious advantages in a distributed memory environment, whereas either model is valid for implementing concurrency in a global memory environment. In an important paper [LAU79], Lauer and Needham showed that, subject to certain restrictions, there exists a duality between the two models, and that for every program implemented in accordance with one of the models, there is a directly corresponding program implemented in accordance with the other. Stroustrup [STR82] has recently confirmed experimentally the validity of the "Lauer-Needham Duality Hypothesis" for a large domain of applications.

The aim of this article is to show how the duality recognized by Lauer and Needham may be used as a tool in the design of concurrent programs, by permiting the determination of the appropriate program structure using one of the two styles mentioned, and its implementation using the dual style. It is shown how the graphical notation proposed by Brinch Hansen [BH77] for describing access rights for a monitor-based implementation has a direct interpretation in terms of an implementation based on message passing. Thus it can be shown that this graphical notation is at a higher level of abstraction than the two implementation models, and unifies their representation.

The structural design process then reduces to the determination of the appropriate structure graph, which may be derived either from viewing the program as using monitors, or as using message passing, for interprocess communication and synchronization. The resulting implementation is a direct consequence of the interpretation of the structure graph in terms of the model to be used in the implementation. In other words, the program may be designed in terms of one model and implemented using the other.

Another consequence of duality is the opportunity to build on the work of other authors who used the dual model. For example, Gentleman's administrator and workers which he defined in terms of message passing, have their direct equivalents in terms of monitors. The structure graphs presented in his article just need to be interpreted according to the monitor model.

REFERENCES

[BH77] Brinch Hansen, P., *The architecture of concurrent programs*, Prentice-Hall, Englewood Cliffs, New Jersey, 1977.

[GEN81] Gentleman,W.M., "Message passing between sequential processes: the reply primitive and the administrator concept", *Software-Practice and Experience, 11* (5), 435-466, 1981.

[LAU79] Lauer,H.E. and Needham,R.M., "On the duality of operating systems structures", *Second International Symposium on Operating Systems*, IRIA, Rocquencourt, France, October 1978, reprinted in *Operating Systems Review 13* (2), 3-19, 1979.

[STR82] Stroustrup,B., "An experiment with the interchangeability of processes and monitors, *Software-Practice and Experience, 12* (11), 1011-1025, 1982.