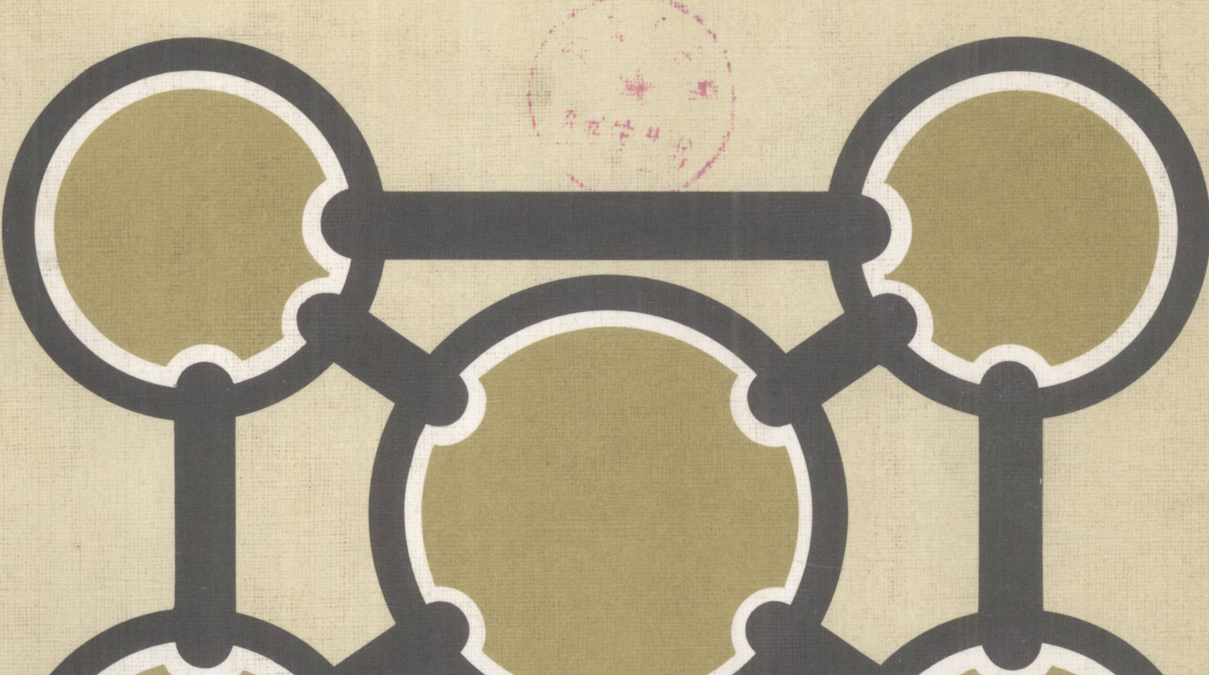
an introduction to DATA PROCESSING NETWORKS

By Adrian V. Stokes



INPUT TWO-NINE STUDENTLITTERATUR

Langton Information Systems Series

AN INTRODUCTION TO DATA PROCESSING NETWORKS

By Adrian V. Stokes

INPUT TWO-NINE STUDENTLITTERATUR

ISBN 0 905897 17 X Input Two-Nine Ltd.
91 44 15301 5 Studentlitteratur ab

Printed in Sweden

PREFACE

This book is intended to be an introduction to data processing networks. It covers the required background material in sufficient depth to give an understanding of the concepts of such networks and supplements this by a glossary and a bibliography.

The author wishes to thank the people who have helped in various ways with the production of this book, especially those who have read drafts and made helpful improvements. The book is based on some of the material used by the author in courses, particularly those for Langton Information Systems Ltd. in the UK, Holland and Belgium. He wishes to thank the participants on those seminars for their forbearance and constructive criticism. Finally, he wishes to thank Alan Simpson, the editor of "Data Communications : Spreading the Message" for his help when writing the original article on which the "Future Trends" chapter is based.

CONTENTS

INDEX TO FIGURES

CHAPTER 1
INTRODUCTION TO COMPUTER NETWORKS

Introduction to Computer Networks

1.1 Introduction

A decade ago, computer networks consisted of two computers, a few feet apart, linked so that one could be accessed from the control terminal of the other. Now, computer networks span the world.

How has this growth come about? What is the state of the art in computer networks? What are the design criteria for such networks? What effect will standardization have on existing networks and what is the future of networks?

These are some of the questions that this book attempts to answer. It is intended as an introduction to the topic of computer networks and, for readers wishing to examine any aspects in greater depth, a comprehensive bibliography is included. In addition, a glossary of some common terms in computer networking is given, together with a list of acronyms.

The book starts with a brief introduction to the concepts and history of computer networks, especially networks provided by the telecommunication authorities (PTT's) in Europe. Chapter 2 covers the basic concepts of data transmission at a level sufficient for the remainder of the book and this leads to a discussion of the various services provided by the PTT's.

The book then examines the problems of implementing a network, including the hardware and software needed, security and the man/machine interface. Having examined the building of the network, its operation is discussed; finally, a look is taken into the future to see how current developments in technology will affect the development of computer networks.

1.2 What is a Computer Network?

In the terminology of computer networks, as with many other branches of computer science, there is considerable confusion. Indeed, the term "computer network" covers a vast range — from such large-scale public or private networks as the ARPA network, involving over a hundred computers and many thousand terminals, to a small, in-house network involving, say, a medium size main-frame and a few intelligent terminals. The term "distributed intelligence" is also used to describe computer networks but, in particular, those of the latter type.

In broad terms, as is obvious from the name, a computer network is a number of computers (where the term "computer" is taken to include intelligent terminals) interconnected in some way. A more formal definition of a computer network is that given by Roberts and Wessler (1970) where it is defined as "a set of autonomous computer systems, interconnected so as to permit interactive resource sharing between any pair of systems". In this book, the word "network" is used in such a way that it is compatible with either of these definitions.

1.3 Why do We Need Computer Networks?

For historical reasons, the modern computer has, in many cases, developed as a fairly general-purpose machine and, as such, either performs no functions especially well or performs some extremely well at the expense of others. This leads immediately to the first justification for networks, namely that of "resource sharing". It makes more sense, instead of choosing one large general-purpose machine, to choose a number of smaller machines, each well suited to its particular application.

In the case of large computer networks, this justification becomes even clearer. Consider an example in the ARPA network. There are countless applications for mass storage devices. However, when talking about terabit memories (of the order of 10^{12} bits), although the cost per bit is extremely low the total cost is nevertheless high. It is therefore impossible to install one such store at each site which has a need and so, for example, in the ARPANET, two stores have been installed, one on the East coast and one on the West and these are shared by many research groups. Furthermore, the store is accessed by a special computer, the "Datacomputer" which is dedicated to this purpose. This is almost certainly a far more efficient way of accessing secondary storage than many machines each accessing a medium sized store locally.

Although this example is a hardware one, similar considerations apply to software resources. Consider, for example, the question of a large database. Besides the obvious problem of providing suitable storage, there is also the problem of maintenance if such a database is duplicated. By only providing one database and allowing access via a network, many problems may be overcome.

The second reason for computer networks is "load sharing". That is, two (or more) computers are connected together so that when one gets heavily loaded, it may offload some work onto the other computer(s). Care must be taken, however, that the overheads involved are not so high as to make this offloading prohibitively expensive or, worse still, less efficient than waiting for the work to be done on the heavily loaded machine.

To give an example of a case where load sharing is not justified, a network consisted of a number of computers, the fastest of which was connected to the remainder of the network via a 2400 bits per second (bps) line. One application required six magnetic tapes to be analyzed each day; the output consisted of about the same amount of data. Although the suggestion of using the fastest computer is superficially attractive since it was approximately twice as fast as any other on the network, a simple calculation shows it to be impractical. The amount of data is about 200 Megabits. On the assumption that an effective throughput of 2 Kbps can be obtained (perhaps optimistic, depending on the quality of the line), to send this amount of data will require about the whole day! In this case, load sharing is not a valid justification.

In general terms, load sharing may be performed in a homogeneous network where the tasks which are shared have little input/output compared with processor time.

However, consider, for example, a company with six sites, each of which needs the full services of a medium-sized computer. It may prove more sensible to provide one large mainframe at a central site with (intelligent) terminals at the others. This is, of course, an extremely common configuration and a few commercial users have set up a decentralized network. But a centralized configuration gives rise to security and reliability problems and a decentralized system may, in fact, be preferable.

Another reason is that a network allows distributed processing. That is, various steps of a job may be performed on different mainframes in the network. However, relatively little use has been made of this technique as yet.

Finally, there is the reason that, with a network, the data on each computer are readily accessible to all the others and this ease of exchanging information may be of great value although it gives rise, once again, to security problems.

In any particular environment, all these factors must be weighed to decide whether the (considerable) cost of implementing a network is justifiable. In some cases, it may be; in others it will not.

1.4 Basic Types of Computer Networks

There are many different types of computer network, differing in topology, method of interconnection, mode of data transfer etc. In this section, the network topology is examined. How the computers are actually connected is not defined as this is not of importance in this section. Therefore, the general term "node" is used. In most networks, the node is not the mainframe ("Host") but is a dedicated communications computer (front-end processor).

There are two basic topologies. The first is a star network (Fig. 1.1) where there is one central computer connected to a number of other computers, none of which is directly connected to any of the others. One notable feature about such a network is that all data must pass through the central node and therefore the reliability of this node is of paramount importance. Such a network requires (n—1) connections (usually telephone lines) where (n) is the total number of computers in the network.

The second type is a distributed network (Fig. 1.2) where there is no central node and each node is connected to a number of nodes. The network is "fully connected" (Fig. 1.3) if each node is connected to every other node (thus requiring n(n—1/2 lines) but this is an extremely unlikely configuration in practice due to the expense (although this may change in the near future because of broadcast satellite techniques which are discussed later). Usually such a network will have sufficient lines to ensure that each node has at least two connections to allow data to be sent by an alternative route in the case of faults.

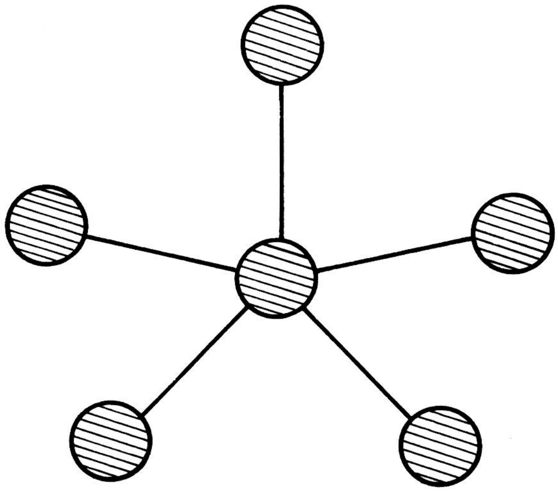


FIG. 1.1 A STAR NETWORK

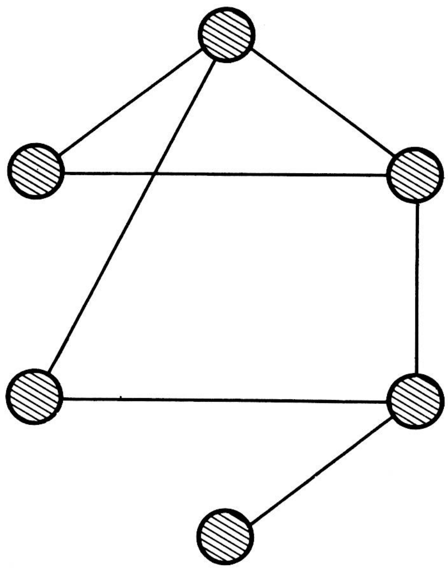


FIG. 1.2 A DISTRIBUTED NETWORK

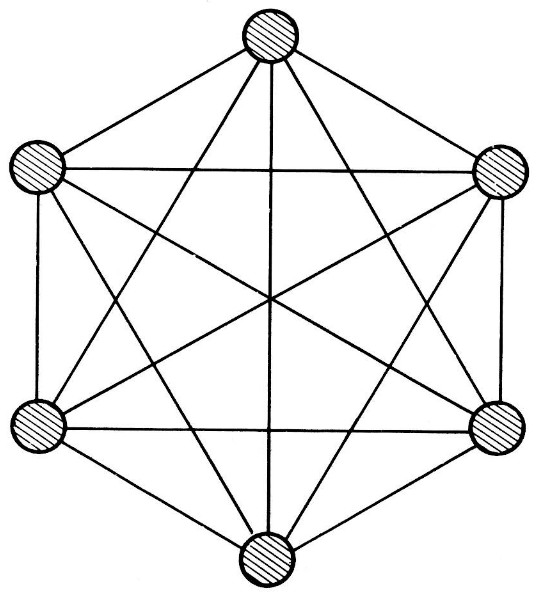


FIG. 1.3 A FULLY-CONNECTED DISTRIBUTED NETWORK

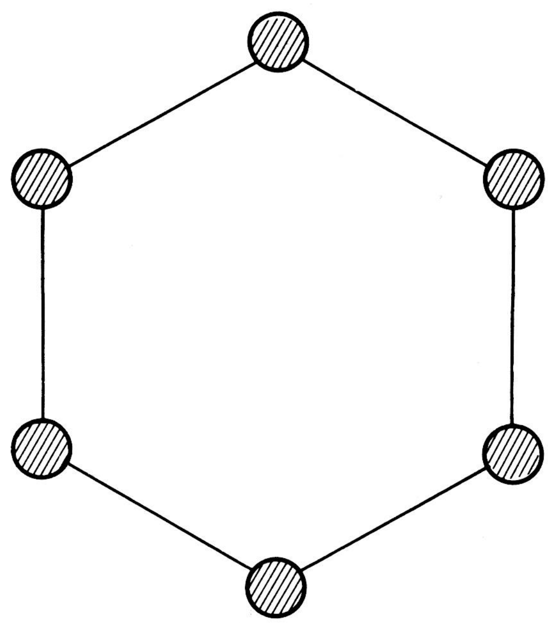


FIG. 1.4 A RING NETWORK

A special case of the distributed network is the "ring network" (Fig. 1.4) where, as the name implies, each node is connected to the two adjacent nodes (requiring (n) lines). Again the problem of reliability arises but there are two distinct paths between any two nodes compared with the one path in the case of a star network, although a special case of a ring network allows data in one direction only ("undirectional ring") and hence this advantage is lost.

Another special case is the hierarchical network in which the computers are interconnected in a tree structure (Fig. 1.5). Such a structure is not often used except that interconnection of complete networks is usually a special case of this.

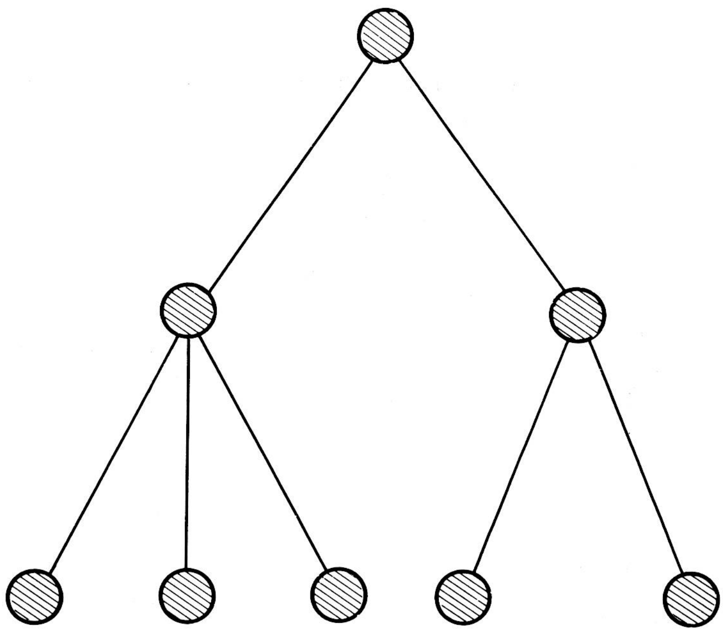


FIG. 1.5 A HIERARCHICAL NETWORK

1.5 The History of Computer Networks

Probably the first experiments in computer networks were those of Marill and Roberts (1966) at the Lincoln Laboratories connecting their TX–2 to the Q–32 at System Development Corporation.

A quantum jump in the development of computer networks occurred when the Advanced Research Projects Agency (ARPA) of the US Department of Defense decided to interconnect various sites performing research under contract to ARPA by means of a computer network. Various options were examined for ARPANET and these are described in detail in Roberts and Wessler (1970). A number of criteria were proposed. These were that there should be at least two physically independent paths between any pair of computers, that the network end to end delay should be less than 0.5 second, that the communications costs should be less than 25% of the computing costs of the connected systems, that the software development cost at each site be minimized and that the network be exceptionally reliable.

In order to achieve these aims, it was decided to use a form of "message-switching" system (see Chapter 2.6) based on concepts attributed to Paul Baran in a paper "On Distributed Communications" published in the Rand Series Reports in August 1964.

These design aims were well achieved. There were two major design decisions which were basic to the success of the network. First, the interconnected computers (Hosts) were isolated from the communications sub-system which consisted of a number of mini-computers (Interface Message Processors – IMP). These IMPs were ruggedized Honeywell–516 (later 316) machines and had no backing storage. In fact, the only moving parts were fans giving a mean time between failure (MTBF) of over a year. Secondly, a distributed topology was chosen with at least two separate paths available to every node, thus giving a down-time between any pair of nodes of about 30 seconds per year. Also, in order to guard against data corruption, a 24 bit cyclic redundancy check was used giving an undetected error rate of 1 in 10^{12}, that is, less than one per year.

The initial contract was given to Bolt, Beranek and Newman (BBN) in 1969 and, by the end of the year, the initial network (Fig. 1.6) was in operation. It expanded rapidly and, although the original design envisaged about twenty sites, this figure was soon surpassed – in late 1971 (Fig. 1.7) and the current configuration is as shown in Fig. 1.8. This includes two point-to-point satellite links, one to Hawaii and one to Norway (and thence to London).

It is perhaps worth pointing out that the original design of ARPANET provided for a maximum of 64 nodes (well in excess of the twenty nodes originally envisaged). This figure has almost been reached and the network is being redesigned at a fundamental level to allow it to expand beyond 64 sites – MORAL – Infinity may be larger than originally thought.

Recently, ARPA has sponsored an experiment using broadcast satellite techniques (SATNET) and this is interconnected with ARPANET via a number of "gateway" computers. The technology of this network is examined later.

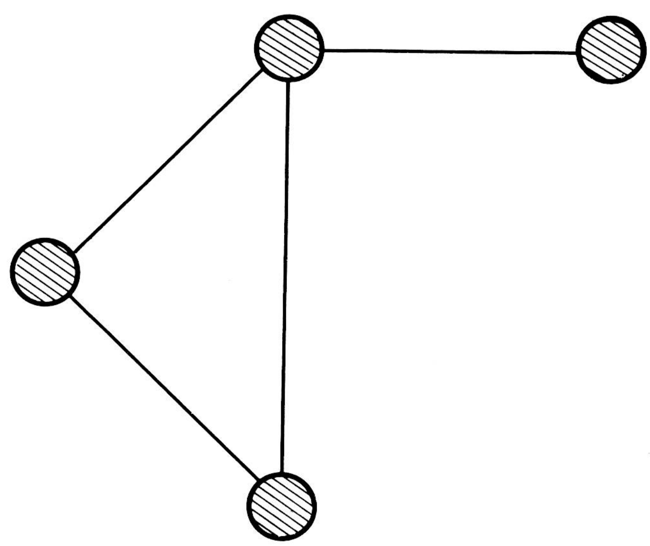


FIG. 1.6 ARPA NETWORK INITIAL TOPOLOGY