



TP274  
B629

8061749  
5

# *The Design of Real Time Applications*

**M. Blackman**

*Arthur Andersen & Co.*



A Wiley-Interscience Publication



E8051749

**JOHN WILEY & SONS**

London · New York · Sydney · Toronto

Copyright © 1975, by John Wiley & Sons Ltd.

Reprinted May 1976

All rights reserved.

No part of this book may be reproduced by any means,  
nor transmitted, nor translated into a machine language  
without the written permission of the publisher.

***Library of Congress Cataloging in Publication Data:***

Blackman, Maurice.

The design of real time applications.

'A Wiley-Interscience publication.'

Includes bibliographical references.

1. Real-time data processing. I. Title.

QA76.54.B55 1975 001.6'44'04 74-26960

ISBN 0 471 07770 4

Printed in Great Britain by J. W. Arrowsmith Ltd.,  
Winterstoke Road, Bristol

*The Design  
of  
Real Time Applications*

## *Preface*

My intention in writing this book is to add to the literature of computing, a guide for analysts, project leaders and managers who are designing and installing real time applications. In my experience, the literature available on real time concentrates on technical matters of software and equipment. In contrast, I have set out to indicate where to start, and how to proceed, to design an application which uses the real time mode of processing. My intention is to fill a gap in the literature concerning the activity of the application designer.

There are two components to the book. The first is a method for organizing the content of a transaction driven system into a hierarchy of system units. It starts with the entire application and ends with the program unit which processes one function for one type of entered message. The second is a scheme of documentation which formally describes each member of each set of system units at each level of subdivision. The attributes described at each level are chosen to ensure early and thorough consideration of the important issues in the real time mode of processing. These are identified early in the book from a study of the unique aspects of real time systems.

I do not wish to claim this book as a complete study of real time commercial computing. I have tried not to duplicate existing literature, particularly in the areas of sizing hardware and designing general software. However, I have had to consider the reader for whom this may be a first contact with real time. There are therefore introductory chapters on types of real time system and a description of one particular system. This system provides the source material for all the worked examples later in the book.

There are also chapters on the topics of planning system design and implementation since I could not assume a shared understanding of the meaning of such terms as preliminary design, detailed design, economic evaluation, etcetera. The really new material is therefore diluted with some recapitulation but still this is not an attempt to present a thorough study of all facets of real time.

The ideas presented have many sources. My first step was made when I needed to describe concisely the logic of the use of a series of terminal entries to accomplish the check in of a passenger during the processing of an aircraft departure. A flowchart showing the decisions governing the use of the available entries was prepared. On reflection it appeared that the design process would have been much better directed had such charts been prepared in advance to

identify the entries needed to suit each system function. Thus the concept of a system unit which linked a series of entries into one function was born. To this concept was added work habits of disciplined and formal documentation developed in batch systems design. My basic knowledge of real time and its unique characteristics was gained from experience of various manufacturers and from the writings of James Martin. The end result, my contribution, is to bring the formality of good batch design practice to real time systems in a way which recognizes and harnesses their unique features.

The methods presented have been used to assist in the design and installation of a number of systems with which I have been associated. The methods are therefore presented as a fully developed scheme of work and documentation. However I am sure that improvements and changes can and will be made. I would ask everyone using the book to think critically about its proposals, and tailor them to suit the scope of his own applications. I would welcome correspondence about successes, failures and adaptations of the methods.

In another respect, it would be naive of me to think that these proposals are the last word in systems methods. Real time use of computers is rapidly developing and advances in both hardware and software are making the work of the programmer less extensive. For all but high performance systems, packaged solutions will become more and more acceptable. The amount of design and implementation work may well reduce to the point where the more detailed levels of documentation will not be necessary. However, in the interim, I believe that the methods proposed are relevant and their use improves the quality of application design.

In recent months, I have noticed a tendency for batch systems development to change. Applications on which I have worked, which have involved processing many different kinds of batched transactions through common programs, have been usefully described by reference to their transactions rather than their programs. IBM have introduced 'structured programming' which includes among its innovations, transaction oriented descriptions of processing logic. I believe that the reader will find some affinities between the methods in this book and these new developments. It may be that the next few years will see the emergence of a method of design and documentation which will suit both batch and real time systems, by virtue of a common concern in both with the transaction as the impetus for processing. I hope that this book will therefore prove to be in the mainstream of general commercial computing.

### ACKNOWLEDGEMENTS

For the development of my thoughts on real time I must thank my colleagues and friends at the then separate airlines of BOAC and BEA, in particular Alan Jacobs, the late Bill Devereaux and John Cutting (BOAC), Mark Haegele, John Morcom, Daphne Harris and the other members of the Transaction Utility Study Group (BEA) and Roger Townsend. Help and advice has also

been received from many other people with whom I have had the pleasure of working in the past few years.

Finally I am indebted to the devotion and care with which the typing and illustration work was done. Many people were involved over a period of time working under the direction of Mary Curley, Linda Jeffries and Terry Gannon. Without their help, the book would never have been fit for publication.

*October 1974*

MAURICE BLACKMAN



<b>PART I THE NATURE OF REAL TIME</b>	<b>1</b>
<b>1 An Invoice Payments System</b>	<b>3</b>
Introduction	3
Description of User Situation	3
The Firm and its Policy	3
Outline of Method	4
The First Day	4
First Entry	4
First Response	6
Second Entry	7
Second Response	7
Associated Response	8
The Batch Processing Phase	9
The Final Day	9
Third Entry	9
Third Response	9
Fourth Entry	11
Fourth Response	11
Fifth Entry	13
Summary	13
<b>2 Real Time Characteristics</b>	<b>15</b>
Purpose	15
Effects are Subject of Book	15
Properties of Real Time Projects	15
Direct Contact	15
Primary Purpose of Real Time	15
No Buffer between User and System	16
Security is Important in Design	16
Integrity, Fallback and Recovery	16
Conversion must be Planned	17
Topics are Considered Earlier	18
Testing is More Rigorous	18



Immediate Processing . . . . .	19
Analogies . . . . .	19
Relationship between Program and Transaction is Reversed . . . . .	19
Complex Interfaces . . . . .	19
Database Techniques are Applicable . . . . .	21
Unpredictable Combinations of Events will Occur . . . . .	21
Testing Needs Software . . . . .	22
Concurrent Execution . . . . .	22
Differs from Multi-programming . . . . .	22
Tasks Contend for Data and Programs . . . . .	23
Tasks must be Separated . . . . .	23
Tasks must be Supervised . . . . .	23
Program Modularity is Desirable . . . . .	24
Concurrency must be Tested . . . . .	24
Less Predictable Demand . . . . .	24
Predictability is Limited . . . . .	24
Task Identities are Unknown . . . . .	25
The System Contains many Queues . . . . .	25
Demand is Investigated Early . . . . .	26
Novel Technology . . . . .	26
Terminals and Communications . . . . .	26
Technology is Developing . . . . .	27
Protection by Logical Terminals . . . . .	27
Equipment Selection Considerations . . . . .	27
Summary . . . . .	28
Consequences in Systems Design . . . . .	28
 <b>3 Variety in Real Time Systems</b>	 31
Purpose . . . . .	31
Variety must be Acknowledged . . . . .	31
Variety in Many Characteristics . . . . .	31
Classification . . . . .	31
Enquiry System . . . . .	32
General Description . . . . .	32
Justification . . . . .	33
Key Design Decisions . . . . .	34
Message Switching . . . . .	34
General Description . . . . .	34
Justification . . . . .	36
Key Design Decisions . . . . .	37
Data Capture . . . . .	37
General Description . . . . .	37
Justification . . . . .	40
Key Design Decisions . . . . .	41

File Update . . . . .	42
General Description . . . . .	42
Justification . . . . .	45
Key Design Decisions . . . . .	46
Memo Update . . . . .	46
General Description . . . . .	46
Justification . . . . .	47
Process Control . . . . .	47
Summary . . . . .	48
<b>4 Conversations and Exchanges</b>	<b>49</b>
Computer Systems Design Tasks . . . . .	49
Batch Systems . . . . .	49
Real Time System Design Difference . . . . .	50
Exchanges and Conversations . . . . .	50
The Exchange Concept . . . . .	50
The Conversation Concept . . . . .	50
Procedures . . . . .	51
Stages of Real Time Design . . . . .	52
The Effect of the Conversation Concept . . . . .	52
Abbreviation of Messages . . . . .	52
Conversation Control Record . . . . .	52
Permitted Next Entries . . . . .	53
Entry Recognition . . . . .	54
The Context Hierarchy . . . . .	54
Relevance to Installation . . . . .	55
Effect on Design Effort . . . . .	56
Common Exchanges . . . . .	56
Common Programs . . . . .	56
Common Entries and Responses . . . . .	56
Multiple Relationships are Produced . . . . .	57
Defining Boundaries . . . . .	58
Separating Conversations . . . . .	58
Exchange Definition . . . . .	58
Summary . . . . .	59
Relationship to Other Chapters . . . . .	59
<b>PART II APPLICATION SYSTEMS DESIGN</b>	<b>61</b>
<b>5 Stages of System Definition</b>	<b>63</b>
Major Stages . . . . .	63
Continuity with Batch Practice . . . . .	63
Priorities set by Long Range Systems Planning . . . . .	63
Project Scope Identifies Boundaries . . . . .	64

Preliminary Systems Design defines the Project . . . . .	64
Installation is the Largest Stage . . . . .	65
Reconciliation Confirms Management Control . . . . .	68
Preliminary Systems Design . . . . .	68
PSD has Several Areas . . . . .	68
Systems Design is the Foundation . . . . .	68
Installation Stage must be Planned . . . . .	69
Equipment Selection depends on System Design . . . . .	69
Supplier Selection should be Objective . . . . .	70
Economics must be Evaluated . . . . .	70
Management Approval precedes Installation . . . . .	71
Scheme for Real Time Systems Design . . . . .	71
Real Time Project Entities . . . . .	71
Documents match Entities . . . . .	72
Documents Separately Defined . . . . .	72
Document Plan forms Work Programme . . . . .	73
Document Sections form a Filing Plan . . . . .	73
Reviews Enforce Quality . . . . .	74
Work Programme for Logic Design . . . . .	75
Work to be Done . . . . .	75
Existing Tasks must be Described . . . . .	75
Computer Assistance can be Identified . . . . .	76
New Procedures will be Needed . . . . .	76
Conversations can now be Designed . . . . .	77
Terminal Use can be Classified . . . . .	77
Exchanges are Selected to Minimize Installation . . . . .	77
Computer Elements are Identified in Exchange Specifications . . . . .	78
Transmission Network can be Designed . . . . .	78
The General Arrangement of Data is Designed . . . . .	78
Computer Element Development forms Installation Schedule . . . . .	79
<b>6 Documentation of Outline System . . . . .</b>	<b>82</b>
Introduction . . . . .	82
Chapter Structure is Formal . . . . .	82
Sequence of Preparation . . . . .	82
Success Depends upon Completeness . . . . .	83
Task Description . . . . .	83
Introduction . . . . .	83
Identifying Sections . . . . .	85
Mechanization Benefit . . . . .	85
References . . . . .	85
Conditions . . . . .	85
Usage . . . . .	85
Conversation Identification . . . . .	87
Introduction . . . . .	87

Inventory of Conversations . . . . .	87
Identity . . . . .	87
Class . . . . .	87
Justification . . . . .	87
Terminal Type . . . . .	88
Volume . . . . .	88
Information Worksheet . . . . .	88
Relationship to Procedures . . . . .	88
Linkage . . . . .	88
Procedure Specification . . . . .	94
Identification . . . . .	94
Description . . . . .	94
Conditions . . . . .	94
Manning . . . . .	94
Fallback . . . . .	97
Recovery . . . . .	97
Conversion . . . . .	97
Information Processed . . . . .	98
Controls . . . . .	98
Performance Monitoring . . . . .	98
Verification . . . . .	101
<b>7 System Design</b> . . . . .	<b>102</b>
Introduction . . . . .	102
Conversation Specification . . . . .	102
Conversation Diagram . . . . .	102
Processing Notes . . . . .	105
Testing Plan . . . . .	107
Verification of Design . . . . .	108
Message Planning . . . . .	108
Exchange Specification . . . . .	112
Inventory of Exchanges . . . . .	112
Automatic Exchanges . . . . .	112
Exchange Worksheet . . . . .	114
Exchange Diagram . . . . .	116
Processing Notes . . . . .	116
Testing Plan . . . . .	120
Exchange Linkage . . . . .	120
Verification of Design . . . . .	123
Database Specification . . . . .	123
Specification is similar to Batch . . . . .	123
Description . . . . .	123
Contents . . . . .	127
Access Method . . . . .	127
Record Use . . . . .	127

File Dimensions . . . . .	127
Control . . . . .	128
Recovery . . . . .	128
Program Identification . . . . .	128
Interfaces are Designed . . . . .	128
Purpose . . . . .	129
Parameters . . . . .	129
Links . . . . .	131
Document Format . . . . .	131
<b>8 Application of Design Data</b>	<b>132</b>
Introduction . . . . .	132
Network Design . . . . .	132
Functions . . . . .	132
Data Required . . . . .	133
Data Assembly . . . . .	133
The Peak Hour . . . . .	133
Peak Hour Analysis . . . . .	136
Computer Power . . . . .	136
Number of Terminals . . . . .	136
Number of Lines and their Connection . . . . .	137
Further Reading . . . . .	137
Response Time . . . . .	138
Control of Modifications . . . . .	138
Design Approval . . . . .	139
Installation Planning . . . . .	139
System Economics . . . . .	140
System Report . . . . .	140
<b>PART III APPLICATION SYSTEMS INSTALLATION</b>	<b>143</b>
<b>9 Detailed Logic Definition</b>	<b>145</b>
Introduction . . . . .	145
Installation Phase . . . . .	145
Some Steps are Common . . . . .	145
Systems Design is to be Completed . . . . .	148
Programming and Debugging . . . . .	148
Control of Modifications is Essential . . . . .	149
Physical and Conversion Preparation . . . . .	149
System Testing, Conversion and Reconciliation . . . . .	151
Chapter deals with Detailed Logic Design . . . . .	152
Software Environment . . . . .	152
Common Functions . . . . .	152
Entry Analysis . . . . .	156

Response Synthesis . . . . .	159
Communications Interface . . . . .	161
Programming Specification . . . . .	163
Interfaces . . . . .	163
Extent of Logic Definition . . . . .	163
Flowcharts . . . . .	164
System Logic . . . . .	164
Work Areas . . . . .	165
Testing . . . . .	165
Statistics . . . . .	166
References . . . . .	167
Service Routines User Specifications . . . . .	167
Specification Defines Interface . . . . .	167
Description is Limited . . . . .	168
Parameter Description is Main Section . . . . .	168
Error Conditions may Exclude Return . . . . .	168
Statistics Refers to Project Monitoring . . . . .	169
References give more Detail . . . . .	169
<b>10 Detailed Data Definition</b> . . . . .	<b>170</b>
Classes of Data . . . . .	170
File Definition . . . . .	170
Contents and Identification . . . . .	170
Multiple Records . . . . .	172
File Structure . . . . .	172
Access Method . . . . .	173
Control and Recovery . . . . .	173
Multiple Versions . . . . .	173
Filed Records . . . . .	174
Data has Further Subdivisions . . . . .	174
Standard Labels . . . . .	174
Repeated Fields or Groups . . . . .	177
Data Position . . . . .	177
Record Size . . . . .	177
Work Areas . . . . .	177
Message Definition . . . . .	178
Need for Further Detail . . . . .	178
Message Versions . . . . .	178
Entry Types . . . . .	179
Entry Design . . . . .	182
Error Entries . . . . .	184
Fixed Length Entries . . . . .	184
Responses . . . . .	185
Message Layouts . . . . .	185

Free Form Entry Definition . . . . .	185
Message Specification . . . . .	188
Field Definition . . . . .	190
Three Formats . . . . .	190
Physical Attributes . . . . .	192
Validation Rules . . . . .	192
Editing Rules . . . . .	192
Examples . . . . .	192
References . . . . .	192
Message Handling Software . . . . .	194
<b>11 Testing Stages</b> . . . . .	<b>195</b>
Nature of Testing . . . . .	195
Importance of Testing is Increased for Real Time . . . . .	195
System Design is a Communication Process . . . . .	196
Testing Corrects Distortion . . . . .	196
Real Time Testing has More Stages . . . . .	197
Simulation . . . . .	197
Functions of Testing Stages . . . . .	197
Program Stage Tests all Detailed Logic . . . . .	197
Link Tests Build Activities . . . . .	198
Exchange Tests Check External Appearance . . . . .	198
Multi-threading . . . . .	199
Conversation Tests Link Exchanges . . . . .	199
Cycle Tests Prove Interaction with Data Files . . . . .	199
Volume Tests Exercise Network . . . . .	200
Interface with Batch System . . . . .	200
Hardware Tests are Left as Late as Possible . . . . .	201
User will make Final Test of Acceptance . . . . .	201
Modifications May be Generated at any Level . . . . .	202
Testing of Software Environment . . . . .	202
Modularity is Even More Important . . . . .	202
Interface Defines Test Conditions . . . . .	203
Tests Employ Dummy Applications . . . . .	203
Traffic must be Simulated . . . . .	203
Software for Testing . . . . .	204
Policy of Simulation . . . . .	204
Simulated Environment . . . . .	205
Program Testing . . . . .	205
Exchange Testing . . . . .	207
Multi-threading . . . . .	207
Test Data Generation . . . . .	209
Test Assembly . . . . .	209
Variety in Applications . . . . .	210
Many Dimensions of Variety Exist . . . . .	210

Simple Activities Need No Links . . . . .	210
Multi-threading is Introduced for Volume . . . . .	210
Interaction is Not Always Necessary . . . . .	210
A Profile can be Drawn for System Classes . . . . .	211
Costs can be High . . . . .	212
<b>12 Testing Methods</b>	<b>213</b>
Test Specification . . . . .	213
Identification of Cases is Foundation . . . . .	213
Specification Ensures Complete Test . . . . .	213
Test Preparation is Reduced by Common Data Files . . . . .	214
One Submission can Include Several Attempts . . . . .	215
Program Test Specification . . . . .	215
Specification has Pre-defined Sections . . . . .	215
Case is Embodied in the Opening Conditions . . . . .	216
Job Control Functions are Listed . . . . .	217
Results are Predicted in Advance . . . . .	217
Program Testing Applies to Parameter Programs . . . . .	218
Link Test Specification . . . . .	218
Links have Various Patterns . . . . .	218
Strings are Built Up . . . . .	219
Tests can use Data from Program Test Stage . . . . .	221
Linkage Table Identifies Cases . . . . .	221
Tests with Messages . . . . .	222
Simulation is Recommended . . . . .	222
Test Cases are Identified from Tables . . . . .	223
Terminals are Used to Prove Network . . . . .	223
Administration of Testing . . . . .	224
Assignments must Avoid Loss of Interest . . . . .	224
Reviews Provide Quality Control . . . . .	225
Annotation of Results is First Review of Tests . . . . .	226
Common Programs must have One Keeper . . . . .	226
System Tests Need New Organization . . . . .	227
 <b>PART IV CONVERSION AND CONCLUSION</b>	 <b>229</b>
<b>13 Conversion and its Documentation</b>	<b>231</b>
The Application Users . . . . .	231
Real Time Introduces New Operators . . . . .	231
Terminal Users Need Instructions . . . . .	231
Computer Operation Includes a New Role . . . . .	232
Chapter Does Not Specify Standards . . . . .	232
Terminal Users Documentation . . . . .	233
Construction Reflects the Organization . . . . .	233



Training and Reference Material Needed . . . . .	233
Reference Manual Reflects Procedures Identified . . . . .	233
Describe Each Procedure in Script Style . . . . .	234
Appendices are Required for Formats and Errors . . . . .	235
System Modification Procedure Affects User Documentation . . . . .	236
System Management Manual . . . . .	236
Operator Functions Change . . . . .	236
Instructions must be Separate from Descriptions . . . . .	237
Some Instructions will be System Wide . . . . .	237
Network and System Patching must be Possible . . . . .	238
Corruption of Data . . . . .	238
Procedures are Needed for Logging Incidents . . . . .	239
Housekeeping Activities are Important . . . . .	239
System Management Facilities should be Integrated with Appli- cations . . . . .	240
Conversations in Conversion . . . . .	240
Conversion Strategies . . . . .	240
System Recognition of Conversations . . . . .	241
Entry Analysis Control Tables . . . . .	241
Continuous Conversion . . . . .	242
<b>14 Conclusion</b> . . . . .	243
Summary of Concepts . . . . .	243
System Subdivision . . . . .	243
Computer Realization . . . . .	243
Testing Structure . . . . .	244
System Characteristics and Types . . . . .	244
Real Time Characteristics and Consequences . . . . .	244
Satisfaction of Requirements . . . . .	244
System Types . . . . .	245
Conversation Types . . . . .	246
Applicability of Concepts . . . . .	246
<b>Glossary</b> . . . . .	248