A systems engineering approach to servitisation system modelling and reliability assessment

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/8298

Publisher: © Kenneth Richard Astley

Please cite the published version.
This item was submitted to Loughborough’s Institutional Repository (https://dspace.lboro.ac.uk/) by the author and is made available under the following Creative Commons Licence conditions.

For the full text of this licence, please go to: http://creativecommons.org/licenses/by-nc-nd/2.5/
A SYSTEMS ENGINEERING APPROACH TO SERVITISATION SYSTEM MODELLING AND RELIABILITY ASSESSMENT

By

Kenneth Richard Astley

A DOCTORAL THESIS

Submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University

March 2011

© by Kenneth Richard Astley 2011
Abstract

Companies are changing their business model in order to improve their long term competitiveness. Once where they provided only products, they are now providing a service with that product resulting in a reduced cost of ownership. Such a business case benefits both customer and service supplier only if the availability of the product, and hence the service, is optimised. For highly integrated product and service offerings this means it is necessary to assess the reliability monitoring service which underpins service availability.

Reliability monitoring service assessment requires examination of not only product monitoring capability but also the effectiveness of the maintenance response prompted by the detection of fault conditions. In order to address these seemingly dissimilar aspects of the reliability monitoring service, a methodology is proposed which defines core aspects of both the product and service organisation. These core aspects provide a basis from which models of both the product and service organisation can be produced.

The models themselves though not functionally representative, portray the primary components of each type of system, the ownership of these system components and how they are interfaced. These system attributes are then examined to establish system risk to reliability by inspection, evaluation of the model or by reference to model source documentation.

The result is a methodology that can be applied to such large scale, highly integrated systems at either an early stage of development or in latter development stages. The methodology will identify weaknesses in each system type, indicating areas which should be considered for system redesign and will also help inform the analyst of whether or not the reliability monitoring service as a whole meets the requirements of the proposed business case.
Acknowledgements

I would like to thank Professor John Andrews for the guidance, insight, support and most of all, encouragement that he gave me in this work as my supervisor.

I would also like to thank my Mom, without whose love, support and understanding not only through these later years, but also throughout my life, I would have been unable to reach this goal.
# Table of Contents

Nomenclature ................................................................................................................ 1
Glossary............................................................................................................................ 2

## Chapter 1  Introduction .................................................................................................. 5

1.1 Research background ................................................................................................. 5
1.2 Current Product maintenance methodologies ............................................................... 7
1.2.1 Reliability Centred Maintenance ........................................................................ 8
1.2.2 Total Productive Maintenance ........................................................................... 8
1.2.3 Condition Based Monitoring .............................................................................. 8
1.3 Principal monitoring techniques used in Condition Based Monitoring .................... 9
1.3.1 Pressure wave/ acoustic .................................................................................. 11
1.3.2 Vibration ........................................................................................................ 11
1.3.3 Temperature ................................................................................................... 11
1.3.4 Fluid stream debris detection ......................................................................... 12
1.3.5 Electrical ....................................................................................................... 13
1.3.6 Software (real time, data fusion real time and data acquired after operation for trending and abnormality detection analysis) .................................... 13
1.4 Research motivation ................................................................................................. 13
1.4.1 Product availability enhancement .................................................................... 16
1.4.2 Consequences of enhanced availability through a monitoring service ................ 17
1.5 Risk and uncertainty in Servitisation business models ........................................... 18
1.5.1 Causes of risk and uncertainty in Servitisation businesses ......................... 20
1.5.2 Managing risk and uncertainty in complex systems .................................... 21
1.6 Differentiating between complex systems elements .............................................. 24
1.7 Soft System modelling and ‘reliability’ assessment techniques........................... 26
1.7.1 Soft System modelling techniques ................................................................ 29
1.7.1.1 Gantt chart .............................................................................................. 30
1.7.1.2 Process flowchart .................................................................................... 30
1.7.1.3 Role Activity Diagrams .......................................................................... 31
1.7.1.4 CIM-OSA ............................................................................................... 33
2.2 Key requirements for a methodology to assess Servitisation systems....
.........................................................................................................................50
2.2.1 Key requirement 1 – Identification of system ownership ..............52
2.2.2 Key requirement 2 – identification of system components ..........52
2.2.3 Key requirement 3 – identification of interfaces .........................53
2.2.4 Key requirement 4 – component context, scheduling and allocation....
.........................................................................................................................53
2.2.5 Key requirement 5 - system context and hierarchy ......................54
2.2.6 Key requirement 6 – ability to integrate legacy systems .............55
2.2.7 Key requirement 7 - assessment technique ..................................55
2.3 Discussion of methodology requirements ....................................56
2.4 The Role Activity Diagram (RAD) notation ....................................59
2.5 Reasons for modifying the Role Activity Diagram approach ..........61
2.6 The concept of extending the RAD notation from a Soft Systems
mapping notation to a Hard Systems mapping notation.......................64
2.6.1 Mapping Soft System attributes to Hard System attributes ........64
2.6.2 Modelling notation level of abstraction .......................................66
2.7 Considerations when applying the notation to Soft Systems ..........67
2.7.1 Further discussion on Gantt charts .............................................67
2.7.2 Further discussion on flowcharts ...............................................68
2.7.3 Discussion of the principal aspects of the Soft System modelling
approach ...........................................................................................................71
2.7.4 'Role' – System Boundary ..............................................................72
2.7.5 'Task' – tasks performed within the system boundary ..................74
2.7.6 'Interface' – interactions between the system boundaries ............74
2.7.7 Overview of process 'flow' ..............................................................76
2.8 Considerations when applying the notation to Hard Systems ...........77
2.8.1 Explanation of approach .................................................................77
2.8.2 'Role' - System boundary or platform ............................................78
2.8.3 'Task' – System component ...........................................................79
2.8.4 'Interface' – Component interfaces ..............................................80
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.8.4.1</td>
<td>Interface ‘flow’</td>
</tr>
<tr>
<td>2.8.4.2</td>
<td>Interface ‘source’</td>
</tr>
<tr>
<td>2.8.4.3</td>
<td>Interface ‘sink’</td>
</tr>
<tr>
<td>2.9</td>
<td>An introduction to the formal description of the methodology</td>
</tr>
<tr>
<td>2.10</td>
<td>A formal description of the methodology and how it is applied</td>
</tr>
<tr>
<td>2.10.1</td>
<td>Hard and Soft System example description</td>
</tr>
<tr>
<td>2.10.2</td>
<td>Step 1 – Determine the Hard System model</td>
</tr>
<tr>
<td>2.10.3</td>
<td>Step 1a - Establish the system platforms</td>
</tr>
<tr>
<td>2.10.4</td>
<td>Step 2 Defining Hard System platform components</td>
</tr>
<tr>
<td>2.10.5</td>
<td>Step 3 - Linking the components and adding interfaces</td>
</tr>
<tr>
<td>2.10.6</td>
<td>Step 4 - Using the model for analysis</td>
</tr>
<tr>
<td>2.10.7</td>
<td>Step 4a – Examine system for vulnerability to reliability issues</td>
</tr>
<tr>
<td>2.10.8</td>
<td>Step 4b – Description of notation used to highlight system reliability risk</td>
</tr>
<tr>
<td>2.10.8.1</td>
<td>Monitored components</td>
</tr>
<tr>
<td>2.10.8.2</td>
<td>Non-monitored components</td>
</tr>
<tr>
<td>2.10.8.3</td>
<td>Components that do not fall into either the monitoring or the non-monitoring categories</td>
</tr>
<tr>
<td>2.10.9</td>
<td>Step 4c - Determine System mitigation</td>
</tr>
<tr>
<td>2.10.10</td>
<td>Step 5 – Model the Soft System mitigation process</td>
</tr>
<tr>
<td>2.10.11</td>
<td>Step 5a - Establish the Soft System boundary</td>
</tr>
<tr>
<td>2.10.12</td>
<td>Step 5b Defining tasks of the Soft Systems boundaries</td>
</tr>
<tr>
<td>2.10.13</td>
<td>Step 5c Linking the tasks and adding the appropriate interfaces</td>
</tr>
<tr>
<td>2.10.14</td>
<td>Step 6 – Analyse the modelled Soft System mitigation process</td>
</tr>
<tr>
<td>2.10.15</td>
<td>Step 6a – description of notation used to highlight system reliability risk</td>
</tr>
<tr>
<td>2.10.16</td>
<td>Step 7 – Determine resultant risk exposure to failure of the system following analysis</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>2.11</td>
<td>Summary of the methodology proposed to model and analyse simple</td>
</tr>
<tr>
<td></td>
<td>systems</td>
</tr>
<tr>
<td>2.12</td>
<td>Chapter Summary</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
</tr>
<tr>
<td>3.2</td>
<td>Simple Hard System example description</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Assumptions regarding the overall system boundary of the example</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Assumptions regarding the control hierarchy of the example</td>
</tr>
<tr>
<td>3.3</td>
<td>Modelling the example Hard System</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Determining the main Hard System boundaries or platforms</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Allocating components to the system platforms</td>
</tr>
<tr>
<td>3.3.3</td>
<td>Arranging and connecting system platforms and their components to</td>
</tr>
<tr>
<td></td>
<td>produce the Hard System model</td>
</tr>
<tr>
<td>3.4</td>
<td>Developing the Hard System Reliability Risk View</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Monitored components</td>
</tr>
<tr>
<td>3.4.2</td>
<td>Non-monitored components</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Remaining components</td>
</tr>
<tr>
<td>3.4.4</td>
<td>Adding FMEA information regarding detection modes to the Hard System</td>
</tr>
<tr>
<td>3.5</td>
<td>Simple Soft System example description</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Assumptions regarding the overall system boundary of the example</td>
</tr>
<tr>
<td>3.5.2</td>
<td>Assumptions regarding the organisational structure of the example</td>
</tr>
<tr>
<td>3.6</td>
<td>Modelling the example Soft system</td>
</tr>
<tr>
<td>3.6.1</td>
<td>Determining the Soft System boundaries for the given example</td>
</tr>
<tr>
<td>3.6.2</td>
<td>Allocating tasks to the System boundary definitions</td>
</tr>
<tr>
<td>3.6.3</td>
<td>Arranging and connecting system boundaries and tasks to produce</td>
</tr>
<tr>
<td></td>
<td>the Soft System model</td>
</tr>
<tr>
<td>3.7</td>
<td>Developing the ‘Soft System Reliability Risk View’</td>
</tr>
</tbody>
</table>
3.7.1 Automatic interface ................................................................. 156
3.7.2 Manual interfaces................................................................... 158
3.8 Resultant risk exposure.............................................................. 160
3.9 Chapter Summary................................................................... 160
3.9.1 Hard System Reliability Risk Modelling review and general guidelines ................................................................. 161
3.9.2 Soft System Reliability Risk Modelling review and general guidelines ................................................................. 162

Chapter 4 - Critical appraisal of the application of the methodology to a simple system ......................................................... 165
4.1 Introduction ........................................................................... 165
4.2 Identified methodology constraints........................................... 165
4.3 Discussion in response to the identified constraints.................... 166
4.3.1 Lack of formality of notation ............................................... 166
4.3.2 Hard System redundant and standby systems...................... 168
4.3.3 Distributed systems............................................................. 169
4.3.4 Software reliability as part of the Hard System..................... 170
4.3.5 Repeatable and conditional Soft System steps....................... 170
4.3.6 Large system complexity..................................................... 173
4.3.7 Quantitative assessment techniques for Hard and Soft Systems... 173
4.4 Suggested extensions and enhancements to the methodology due to the constraints listed above.............................................. 174
4.4.1 Modelling redundant Hard Systems ...................................... 174
4.4.1.1 Applying the notation to the example of system redundancy ..................................................................................... 176
4.4.1.2 Summary guidelines for modelling redundant Hard Systems........ 181
4.4.2 Modelling multi lane systems (hot or cold standby Hard Systems) ...... ......................................................................................... 182
4.4.3 Modelling distributed Hard Systems.................................... 183
4.4.4 Modelling larger, more complex Hard and Soft Systems .......... 186
4.4.4.1 The Role Context Chart Concept......................................... 188
4.4.4.2 The Soft System Context Chart a redefinition of the Role Context Chart concept ................................................................. 193
4.4.4.3 Defining the Soft Systems boundary level limits of the model .............................................................................................. 196
4.4.5 Extending the Soft System Context Chart model notation to Hard Systems ........................................................................ 201
4.4.6 An example to demonstrate the extended modelling notation set used to produce the Soft Systems Context Chart ......................... 204
4.4.6.1 Soft System boundary definition ........................................... 205
4.4.6.2 Soft System boundary function definition ............................... 206
4.4.6.3 Interface - how the Soft System boundary functions interact ............................................................................................... 208
4.4.6.4 Soft System Context Chart for the complex Soft System example .......................................................................................... 210
4.4.6.5 Summary guidelines for modelling large complex Soft Systems ............................................................................................. 218
4.4.7 An example to demonstrate the extended modelling notation set used to produce the Hard Systems Context Chart ....................... 220
4.4.7.1 Hard System platform definition .......................................... 221
4.4.7.2 Hard System platform function definition ............................. 221
4.4.7.3 Interface - how the Hard System platform functions interact ............................................................................................... 223
4.4.7.4 Hard Systems Context Chart for the complex Hard System example ....................................................................................... 225
4.4.7.5 Summary guidelines for modelling large complex Hard Systems ............................................................................................. 228
4.4.8 Integration with existing or suggested new quantitative assessment techniques for Hard and Soft Systems ............................... 232
4.5 Chapter Summary ........................................................................... 234

Chapter 5 - Application to more complex Hard Systems ......................... 237
5.1 Introduction ..................................................................................... 237
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.7.8</td>
<td>Development of the detailed Soft System model with associated Soft System Reliability Risk Views for the example Soft System support mechanism</td>
<td>328</td>
</tr>
<tr>
<td>6.7.8.1</td>
<td>The data collection phase</td>
<td>328</td>
</tr>
<tr>
<td>6.7.8.2</td>
<td>The data analysis phase</td>
<td>332</td>
</tr>
<tr>
<td>6.7.8.3</td>
<td>The resolution phase</td>
<td>341</td>
</tr>
<tr>
<td>6.8</td>
<td>Soft System modelling technique summary</td>
<td>350</td>
</tr>
<tr>
<td>Chapter 7 - Proposed reliability assessment method</td>
<td>353</td>
<td></td>
</tr>
<tr>
<td>7.1</td>
<td>Introduction</td>
<td>353</td>
</tr>
<tr>
<td>7.2</td>
<td>Assessment aims for Hard Systems</td>
<td>354</td>
</tr>
<tr>
<td>7.3</td>
<td>Assessment aims for Soft Systems</td>
<td>355</td>
</tr>
<tr>
<td>7.4</td>
<td>The effect of the holistic nature of reliability monitoring system and also its ‘duality’ on assessment aims</td>
<td>356</td>
</tr>
<tr>
<td>7.5</td>
<td>Hard System assessment</td>
<td>357</td>
</tr>
<tr>
<td>7.5.1</td>
<td>System assessment for Hard Systems of a simple nature or those without a safety assessment made against them</td>
<td>357</td>
</tr>
<tr>
<td>7.5.2</td>
<td>System assessment for Hard Systems of a complex nature or those with a safety assessment made against them</td>
<td>360</td>
</tr>
<tr>
<td>7.6</td>
<td>Hard System reliability assessment summary</td>
<td>362</td>
</tr>
<tr>
<td>7.7</td>
<td>Soft System assessment</td>
<td>363</td>
</tr>
<tr>
<td>7.7.1</td>
<td>Developing Soft System models for reliability assessment</td>
<td>364</td>
</tr>
<tr>
<td>7.7.1.1</td>
<td>Relating timings and process flow conditions to the Soft System model</td>
<td>366</td>
</tr>
<tr>
<td>7.8</td>
<td>An example Soft System timing analysis</td>
<td>372</td>
</tr>
<tr>
<td>7.8.1</td>
<td>Data capture process shown in Figure 7.7.1</td>
<td>372</td>
</tr>
<tr>
<td>7.8.2</td>
<td>Known anomaly diagnostic process shown in Figure 7.8.2</td>
<td>376</td>
</tr>
<tr>
<td>7.8.3</td>
<td>Known event response process shown in Figure 7.8.4</td>
<td>379</td>
</tr>
<tr>
<td>7.8.4</td>
<td>Complete Soft System model response time</td>
<td>382</td>
</tr>
<tr>
<td>7.9</td>
<td>Soft System execution time reliability analysis</td>
<td>384</td>
</tr>
<tr>
<td>7.10</td>
<td>Soft System reliability assessment summary</td>
<td>384</td>
</tr>
</tbody>
</table>
Chapter 8 – Conclusions and future work .......................................................... 387
8.1 Summary of work ...................................................................................... 387
8.2 Conclusions ............................................................................................. 391
8.3 Further research ..................................................................................... 396
References ................................................................................................... 397
**List of Tables**

Table 2.6.1  Mapping System attributes to Hard and Soft Systems.................65
Table 5.6.1  Table of vulnerable components by module..........................250
Table 5.6.2  Partitioning along the lines of interface..............................256
Table 5.6.3  Modules grouped by mechanical interface..............................263
Table 5.6.4  Modules grouped by pneumatic interface...............................264
Table 5.6.5  Modules grouped by hydraulic interface.................................264
Table 5.6.6  Modules grouped by fuel interface.....................................265
Table 5.6.7  Modules grouped by electrical interface.................................266
List of Figures

Figure 1.5.1 The ‘Chain’ of major subsystems of a technological system………18
Figure 1.5.2 The general decision support process………………………………22
Figure 1.7.1 An example Gantt Chart………………………………………………30
Figure 1.7.2 Example of a Flowchart………………………………………………31
Figure 1.7.3 An example of Role Activity Diagram notation……………………33
Figure 2.4.1 An example Role Activity Diagram showing an advice request 
following a failure alarm……………………………………………………….61
Figure 2.5.1 Modified RAD Notation – the reduced set………………………..63
Figure 2.7.1 An example Gantt chart representing a maintenance action………68
Figure 2.7.2 A flowchart showing response to a predicted failure event………70

Figure 2.10.1 A block diagram of the reliability monitoring of an engine tank 
(Hard System description)…………………………………………………..84
Figure 2.10.2 A flowchart of the analysis process of data received from the 
Hard System (Soft System description)……………………………………..85
Figure 2.10.3 Modelling Notation for a Hard System Platform………………..87
Figure 2.10.4 Hard System platforms with their respective components………..88
Figure 2.10.5 Hard System platforms showing interfaces and linked platform 
components…………………………………………………………………90
Figure 2.10.6 The Hard System to be analysed……………………………………97
Figure 2.10.7 The Hard System Reliability Risk View…………………………..97
Figure 2.10.8 Showing the main system boundaries in the Soft System mitigation 
process……………………………………………………………………….102
Figure 2.10.9 Showing the tasks that need to be carried out apportioned amongst 
the system boundaries that have responsibility for those tasks…104
Figure 2.10.10 Showing linkages of tasks, indicating task sequence..............105
Figure 2.10.11 Showing sequencing of tasks and interfaces between roles.....106
Figure 2.10.12 Soft System Reliability Risk View........................................113
Figure 3.2.1 A simple tank system.................................................................119
Figure 3.3.1 The principle system platforms of the tank system.................123
Figure 3.3.2 Assigning components to platforms created by the interface
definition......................................................................................................127
Figure 3.3.3 Showing fluid system flow transfers between system
platforms........................................................................................................130
Figure 3.3.4 The complete modelled view of the system described by Figure
3.2.1................................................................................................................132
Figure 3.4.1 Hard System Reliability Risk View – monitored components......136
Figure 3.4.2 Hard System Reliability Risk View - monitored and non – monitored
components....................................................................................................138
Figure 3.4.3 Hard System Reliability Risk View showing all component
categories......................................................................................................140
Figure 3.4.4 Hard System Reliability Risk View – adding FMECA
information......................................................................................................143
Figure 3.5.1 A Gantt chart representing a maintenance action...............145
Figure 3.6.1 System boundary definitions of the response to a maintenance alert
based on the Gantt Chart shown in Figure 3.5.1 for ‘predicted
fault’..............................................................................................................149
Figure 3.6.2 Model showing tasks assigned to Soft System boundaries for
‘predicted fault’..............................................................................................151
Figure 3.6.3 The completed model of the Soft system represented by the Gantt
chart in shown in Figure 3.5.1 for ‘predicted fault’.................................154
Figure 3.7.1 Showing automatic interactions of the ‘Soft System Reliability Risk View’ for ‘suspected fault’……………………………………………157
Figure 3.7.2 Showing manual and automatic interactions of the ‘Soft System Reliability Risk View’ for ‘suspected fault’……………………………………159
Figure 4.4.1 Multi input tank system…………………………………………………………….175
Figure 4.4.2 Modelled representation of the Multi input tank system………………180
Figure 4.4.3 An example of a SCADA system or typical distributed Hard System………………………………………………………………...185
Figure 4.4.4 Pugh’s Total Design Model…………………………………………………………190
Figure 4.4.5 Role Context Chart - the notation proposed by Murdoch et al……192
Figure 4.4.6 Soft System Context Chart – showing the revised notation concept of the Role Context Chart applied to the model shown in Figure 4.4.5………194
Figure 4.4.7 The V-Process model……………………………………………………………………198
Figure 4.4.8 Showing the Soft System Context Chart notation for the Airframer Role……………………………………………………………………………….206
Figure 4.4.9 Showing the Soft System Context Chart notation for the Airframer Soft System boundary function named ‘Airframer Product Support’……………………………………………………………………………….207
Figure 4.4.10 showing the Soft System Context Chart notation for Soft Systems boundary and Soft System boundary function ownership………207
Figure 4.4.11 Showing the Soft System Context Chart notation for an interaction between two Soft System boundary Functions………………209
Figure 4.4.12 Soft System Context Chart for the complex Soft System example………………………………………………………………………………211
Figure 4.4.13 Soft System Context Chart highlighting the interfaces appropriate for more detailed modelling of a response to a known failure event………………………………………………………………………………215
Figure 4.4.14 Detailed Soft System definition of the Soft System boundary functions highlighted in the Soft System Context Chart in Figure 4.4.13.................................................................217

Figure 4.4.15 Showing the Hard System Context Chart notation for the ‘aircraft’...............................................................221

Figure 4.4.16 Showing the Hard Systems Context Chart notation for the Aircraft Hard Systems platform function 'aircraft fuel system'........222

Figure 4.4.17 Showing the hard System Context Chart notation for Hard Systems platform and Hard System platform function ownership........222

Figure 4.4.18 Showing the Hard System Context Chart notation for an interaction between two Hard System platform functions.................224

Figure 4.4.19 Hard Systems Context Chart for the complex Hard System example.................................................................226

Figure 4.4.20 Hard System Context Chart highlighting the interfaces involved in monitoring the oil system...........................................229

Figure 4.4.21 Detailed Hard System mapping of the Hard System platform functions highlighted in the Hard System Context Chart shown in Figure 4.4.20.................................................................231

Figure 5.7.1 Example of notation used to show a module as a Hard System platform function......................................................271

Figure 5.7.2 Notation used for a component belonging to a Hard System platform function – no Performance Effect Criticality assigned........272

Figure 5.7.3. Notation used for a component belonging to a Hard System platform function – Performance Effect Criticality assigned.......273

Figure 5.7.4 A Hard System Reliability Risk View for the oil system interface..275

Figure 5.9.1 A diagnostic monitoring mitigation view covering engine bearing system failure..........................................................285
Figure 5.9.2 A predictive monitoring Hard System Reliability Risk Mitigation View used to predict high engine vibration ........................................288

Figure 5.9.3 A Summary Hard System Reliability Risk Mitigation View of all components involved in mitigation of reliability issues for the oil system interface type ......................................................292

Figure 6.7.1 An example process flow chart ..................................................307

Figure 6.7.2 Developing the flow chart into a modified Role Activity Diagram format ...............................................................................................312

Figure 6.7.3 A flowchart showing the response to operator feedback following a fault detection warning delivered by the service provider ........315

Figure 6.7.4 Soft System Context Chart showing derived Soft System boundaries .................................................................................................317

Figure 6.7.5 Soft Systems Context Chart for a fault detection support mechanism ........................................................................................323

Figure 6.7.6 Developing the Soft System Reliability Risk View using the example process given in Figure 6.7.2 ..........................................................326

Figure 6.7.7 Soft System Context Chart for data collection phase ............330

Figure 6.7.8 Data collection phase (detailed process) .................................331

Figure 6.7.9 Soft System Context Chart for known anomaly diagnostic phase 335

Figure 6.7.10 Soft System Reliability Risk View of the known anomaly Diagnostic Process .........................................................................................336

Figure 6.7.11 Soft System Context Chart for known anomaly predictive phase ........................................................................................................337

Figure 6.7.12 Soft System Reliability Risk View of the known anomaly predictive process .........................................................................................338

Figure 6.7.13 Soft System Context Chart for an unknown anomaly predictive phase .................................................................................................339
Figure 6.7.14 Soft System Reliability Risk View of the unknown anomaly predictive process……………………………………………………………….340

Figure 6.7.15 Soft System Context Chart for a known event response phase………………………………………………………………..344

Figure 6.7.16 Soft System Reliability Risk View for a known event response process……………………………………………………………345

Figure 6.7.17 Soft System Context Chart for an unknown event where the operator ignores maintenance advice………………………….346

Figure 6.7.18 Soft System Reliability Risk View for an unknown event operator ignores service provider advice……………………………347

Figure 6.7.19 Soft System Context Chart for an unknown event, operator requests event monitoring…………………………………………348

Figure 6.7.20 Soft System Reliability Risk View for an unknown event operator requests monitor progress of event…………………………349

Figure 7.7.1 Soft System Reliability Risk View of the data collection phase (detailed process)…………………………………………………367

Figure 7.7.2 Soft System Reliability Risk View of the unknown anomaly prognostic process……………………………………………………369

Figure 7.7.3 Soft System Reliability Risk View of the unknown event operator requests monitor progress of event…………………………371

Figure 7.8.1 Block diagram of the data acquisition phase defined by Figure 7.7.1………………………………………………………………..374

Figure 7.8.2 Soft System Reliability Risk View of the known anomaly Diagnostic Process……………………………………………………377

Figure 7.8.3 Block diagram of the Known anomaly Diagnostic Process defined by Figure 7.8.2…………………………………………………378

Figure 7.8.4 Soft System Reliability Risk View of the known event response process…………………………………………………………..380
Figure 7.8.5 Block diagram of the known event response process defined by Figure 7.8.4…………………………………………………………………………..381
## Nomenclature

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ARIS</td>
<td>Architecture of Integrated Information Systems</td>
</tr>
<tr>
<td>BIT</td>
<td>Built In Test</td>
</tr>
<tr>
<td>BPMN</td>
<td>Business Process Modelling Notation</td>
</tr>
<tr>
<td>CIM-OSA</td>
<td>Computer Integrated Manufacturing Open Systems Architecture</td>
</tr>
<tr>
<td>CMMI</td>
<td>Capability Maturity Model Integration</td>
</tr>
<tr>
<td>FADEC</td>
<td>Full Authority Digital Engine Control</td>
</tr>
<tr>
<td>FMEA</td>
<td>Failure Modes and Effect Analysis</td>
</tr>
<tr>
<td>FMECA</td>
<td>Failure Modes and Effects Criticality Analysis</td>
</tr>
<tr>
<td>HAZOP</td>
<td>Hazard and Operability</td>
</tr>
<tr>
<td>HCR</td>
<td>Human Cognitive Reliability</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organisation for Standardisation</td>
</tr>
<tr>
<td>IT</td>
<td>Information Technology</td>
</tr>
<tr>
<td>LRU</td>
<td>Line Replaceable Unit</td>
</tr>
<tr>
<td>MTTF</td>
<td>Mean Time To Failure</td>
</tr>
<tr>
<td>MTTR</td>
<td>Mean Time To Repair</td>
</tr>
<tr>
<td>PEC</td>
<td>Performance Effect Criticality</td>
</tr>
<tr>
<td>PSS</td>
<td>Product Service System</td>
</tr>
<tr>
<td>RAD</td>
<td>Role Activity Diagram</td>
</tr>
<tr>
<td>RCM</td>
<td>Reliability Centred Maintenance</td>
</tr>
<tr>
<td>SAE</td>
<td>Society of Automotive Engineers</td>
</tr>
<tr>
<td>SCADA</td>
<td>Supervisory Control And Data Acquisition</td>
</tr>
<tr>
<td>SCAMPI</td>
<td>Standard CMMI Appraisal Method for Process Improvement</td>
</tr>
<tr>
<td>THERP</td>
<td>Technique for Human Error Rate Prediction</td>
</tr>
<tr>
<td>TPM</td>
<td>Total Productive Maintenance</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modelling Language</td>
</tr>
</tbody>
</table>
Glossary

Hard system
A definition of a ‘Hard System’ is the type of system which is a physical system that can be represented as a designed product such as a car or an aircraft.

Interface
An interface is the connection between either a Platform in a Hard System or a System boundary in a Soft System.

The interfaces between each System boundary can be considered to be deliverables. That is, what one stakeholder has to produce and deliver to another stakeholder before the process is allowed to continue. They may also sometimes be an interaction between System boundaries where for example a message or control authority is passed from the one to the other.

A Hard System interface is defined as the connection between Platforms of identified components of the same interface type. For example, how the Hydraulic components in one Platform connect to the Hydraulic components in another Platform.

Platform
A Platform is considered to be primarily related to Hard Systems. It is considered to be an entity which is a major component of a Hard System that may describe a physical, contractual or legal boundary.

It will consist of its own subsystems and components which will interface with other platform subsystems and, possibly, the outside world.

Platforms may exist at different levels of abstraction. Examples would be an Aircraft or an aircraft engine. In this way levels of abstraction for analysis purposes are differentiated with the higher level of abstraction being the aircraft
as a platform, the lower level being a sub system component of an aircraft such as an aircraft engine as a separate platform.

**Reliability monitoring system**

In the context of this thesis a Reliability monitoring system is a system which monitors component failure and analyses and reports that failure in order to maximise product availability. This system therefore incorporates both Hard System components (for example monitoring sensors) and Soft System components (for example the maintenance response to a detected failure).

**System boundary**

System boundaries are considered to be the Soft System equivalent of a platform. As such they could refer to either a person or a team or departmental responsibility.

System boundaries are usually defined by deciding which are the principal departments (or specific individuals) in a company (or companies) that will be responsible for specific major functions in the Soft System.

An example one of these main system boundaries might be the person or department that orders withdrawal of equipment from service in order to effect a repair.

**Soft system**

Soft Systems are usually human activity systems which in particular have:

1) An ongoing purpose or mission. In the case of a Soft System this may be a continuing pursuit of something which can never be finally achieved – something such as ‘maintaining relationship’.

2) Has a measure of performance. This is the measure which signals progress or regress in pursuing purposes of trying to achieve objectives.
3) Contains a decision taking process – notionally a ‘decision taker’, as long as this is taken not to be a person but a role which many people in a given system may occupy. Via the decision – taking process the system may take regulatory actions in the light of 1) and 2).

4) Has components which are themselves systems having all the properties of a system.

5) Has components which interact, which show a degree of connectivity (which may be materials, information or influence) such that effects and actions can be transmitted through the system.

6) Exists in a wider system and/or environment with which it interacts.

7) Has a boundary which separates it from 6) which is formally defined by the area within which the decision taking process has power to cause action to be taken – as opposed to hopefully influencing the environment.

8) Has resources, physical and through human participants, abstract which are at the disposal of the decision taking process.

9) Has some guarantee of continuity, is not ephemeral, has ‘long term stability’, will recover stability after some degree of disturbance. This might be helped from outside the system; it might derive internally from participant’s commitment to 1).
Chapter 1  Introduction

1.1 Research background

In recent years many industries have changed their business model from that of making a profit on manufactured products and high cost spares to one where the product manufacturer takes full responsibility not only for manufacture but also for maintenance and operation of its product in the field, delivering profits based as much on the aftermarket side of the business as on the manufacturing side.

This process has been described as ‘Servitisation’ a term defined by Professor Sandra Vandermerwe and Juan Rada [1].

Servitisation business models apply to a number of industries and can describe a number of different business models, recently for instance, Xerox changed their business model from supplying copiers to document management and other consultancy services and UK pump manufacturer Edwards Vacuum have turned their business model to refurbishment of their failed pumps which are refurbished to their latest standard and returned to the field when replacements are required and are now applying the same business model to refurbish their competitors pumps [2].

Such a business model is also prevalent in the aerospace industry and is typified by the Rolls Royce ‘power by the hour’ business model, where Total Care™ service management packages [3,60] are bought by the customers that effectively pay for engine operational use rather than for ownership of the engine.

Such business models are attractive for both the customer and the product supplier and are typified by the following principal characteristics [3):

- Customer definition of support level required;
  - The customer decides how little or how much support is provided by the supplier.
- Life cycle cost;
  - A secure mutually beneficial relationship between customer and product supplier is developed to optimise life cycle cost where the customer is provided with the latest technology and the supplier has a feedback path for product improvement.

- Sustainable product design;
  - An opportunity for used equipment to be upgraded and re-used.

- Availability;
  - Supplier commits to availability targets and customer has working equipment when required.

- Develop product knowledge in use;
  - Increases supplier’s knowledge of product behaviour in the field.

In the case of Rolls Royce, the cost of product ownership to the customer is effectively reduced and there is an option to completely outsource the maintenance function, allowing the customer to concentrate on its core business of flight planning and support of flight operations.

Benefits also include guaranteed long term business commitment from its customers along with the technical advantage of a greater understanding of how its products are used and behave ‘in the field’ which will aid product development and improvement.

Despite these benefits however, there are additional constraints on this operational model that neither party will have encountered in their previous working relationship. Indeed both parties will of necessity have to integrate or as a minimum, compromise on business activities that were previously defined by strict organisational and technical responsibility boundaries, as is the case with the Rolls Royce Total Care™ service management packages. This situation becomes more complicated as the number of parties involved in the Servitisation model increases.
It may also be the case that the product itself will need to be reconfigured in terms of: lower development costs; ease of maintenance and reliability of the product in order to meet business case demands on the cost of ownership to the product supplier in order to maximise their profit.

In the Rolls Royce application discussed above in particular, there are additional demands as the product:

- Has a long life cycle (20 flying years would not be unusual);
- Is peripatetic;
- Will only usually have one primary maintenance centre to cope with major reliability failures;
- Will cause a major loss of profit and customer reputation if their customers do not reach their destination on time;
- May in addition incur compensation and/or onwards travel transfer costs in the case of a major reliability failure.

These factors, when considered as a whole, illustrate the complex situation that arises when attempting to produce a business case for Servitisation. The most important aspect of which is the judgement of whether or not the business and its product has the capacity and, importantly, the technical capability to determine and resolve incipient reliability issues in time to deliver the expected equipment availability and consequent business benefits.

### 1.2 Current Product maintenance methodologies

There are a number of methodologies that have been developed to enhance reliability of products in the field and that could therefore be applied to a Servitised business. These methodologies are discussed in more detail in this section.
1.2.1 Reliability Centred Maintenance

Considered more as a methodology for maintenance, Reliability Centred Maintenance (RCM) is governed by SAE Standard JA1011. It is used to establish a scheduled maintenance programme and is used in many industries such as aerospace, electrical utilities and the chemical industry.

The aim of RCM is to ensure that the design is optimised in a reliability sense to ensure that the system availability is high. The design is examined functionally and potential failure effects are ranked using RCM logic in order to determine an appropriate maintenance strategy for that function [4].

1.2.2 Total Productive Maintenance

Total Productive Maintenance (TPM) is biased toward organisational awareness of maintenance issues in process plant operations. It is a technique of optimising the relationship between human and machine systems in a production environment.

Implementation of this philosophy often requires a cultural change to be fully implemented in that it recognises a co-dependency between Production, Operation and Maintenance and that any deterioration in production equipment is a shared responsibility which has to be resolved to the satisfaction of all and the benefit to the business.

It is above all a philosophy of continuous improvement of the production environment [5].

1.2.3 Condition Based Monitoring

This is an approach whereby equipment is monitored for its health in order to try and determine an appropriate period by which necessary maintenance is required. Such an approach is used in electrical utility companies and process plant operations where unavailability of specific equipment leads to loss of production.
The approach relies on being able to detect specific component failure trajectories, which in turn relies on Weibull techniques to estimate life and employs a Proportional Hazards Model (an extension to Weibull) in order to determine an optimised time in which to remove or repair equipment [6].

This approach is seen as the best approach for Servitised products as it is an approach which takes into account the possibility that in-service reliability is affected not only by initial design factors but also by other factors such as incorrect or poorly applied maintenance procedures. Whilst the other two methodologies do their best to prevent poorly applied product maintenance, this approach will detect if a problem has occurred during maintenance.

In addition, such an approach aims to provide warning of impending failure such that maintenance can be scheduled to maximise availability of the product.

There are other benefits in that design configuration changes to the product can also be monitored for differences in product operation in service.

1.3 Principal monitoring techniques used in Condition Based Monitoring

Condition Based Monitoring techniques are used to support Servitised products [56]. These techniques are used to monitor the equipment against known system characteristics in order to develop knowledge of system behaviour in the field and to determine when maintenance action is required. This ability to determine when a maintenance action is required by monitoring the failure of the system rather than requiring a scheduled maintenance period both increases product availability (as maintenance only occurs when necessary) and reduces cost by optimising the maintenance activity, involving maintenance only as and when required.

There is also a need to use these systems when a product becomes ‘Servitised’ not only for the above reason but also because the product supplier themselves will require some form of ‘insurance policy’ to account for unexpected arisings from the operation of the equipment in service. These arisings will stem from:
1. Incomplete information on the equipments environment;

2. Changes or modifications to the equipment in service (unsuccessful component integration);

3. Unpredicted methods of operation;


As the causes of these arisings are at the time of entry into service of the product ‘unknown’ and primarily will have an economic impact rather than an impact to product safety (as the safety case will ensure these arisings are mitigated), these techniques can provide corroborative evidence in order to develop a holistic understanding of the nature of the failure.

An understanding of these techniques and their respective capabilities is therefore central to developing technology appropriate to support the business case of Servitised products.

There are a number of techniques that can be adopted to predict and detect failure in a Condition Based Monitoring methodology. They fall primarily into the following categories and their use depends on application with pressure wave/acoustic, vibration and fluid stream detection applications biased toward mechanical systems and temperature based measurements important to both electrical and mechanical systems.

It should be remembered that Electrical systems can also be ‘self’ monitoring through open and short circuit checks.

Software is used not only in monitoring all types of system through sensor input as the system is performing its function in real time but also for trend monitoring away from the main system operation where days and even months of operational data are assessed. Software is expected to be fit for purpose and fail proof in production as it is not a physical entity and is usually extensively tested throughout the development programme.
1.3.1 Pressure wave/ acoustic
Pressure sensors or microphones (in the case of acoustic measurement sensors) are used to detect changes in pressure about the system of interest. Pressure sensors can be focused on a specific component or can determine more general changes in airflow.

Acoustic sensors have also been used to great effect on bearing failure [7]. In the very early stages of bearing failure the bearing will emit high frequency energy in the acoustic range. Although there is a sacrifice in the increased processor load and storage capacity required to detect these high frequencies, the early detection of incipient bearing failure sometimes outweighs these disadvantages providing warning of the need to change bearings long before the effects are noticed by vibration or evidence of debris in the oil stream.

1.3.2 Vibration
Vibration transducers have been used for many years to detect failures in rotating machinery. In particular they are highly effective in detecting bearing failure (in the same manner as acoustics) in determining bearing failure.

This is a mature technique and when combined with speed sensor information the resulting frequency spectrum can be used not only to detect bearing failure, but also which part of the bearing is failing. This is done by defining at what point in the frequency spectrum the greatest energy increase is and relating that to the rotational frequency of the bearing component part. As each component part of the bearing rotates at its own particular speed, this rise in energy can identify that part of the bearing which rotates at that speed as the part that is most under distress.

1.3.3 Temperature
Temperature transducers have been used for many years for both mechanical and electrical systems.
Mechanical systems heating above a pre determined level are often indications of localised or general system problems and electrical equipment has to be maintained at or below a certain temperature to perform as required.

1.3.4 Fluid stream debris detection

Magnetic chip detectors have been used for many years in both automotive and aerospace applications. The level of debris, the size and the morphology of that debris when examined can give accurate analysis of incipient transmission or engine failure (depending upon the application).

These techniques however, in recent years, have become more sophisticated in a move away from magnetic chip detectors to sensors which provide more diagnostic capability.

For example, in the aero engine field, the Gastops MetalSCAN sensor [8] is used and is able to detect ferro magnetic and non ferro magnetic debris particles in an oil stream and is non obtrusive to flow. The device is electro magnetic in nature and as debris passes the electromagnetic sensor, its size and nature is determined in order to build an aggregated view in time of how much debris is being generated by wear and whether the size of the particles is increasing. Larger size particles would indicate that the system being monitored is deteriorating more quickly.

A similar sensor, again used in the aero engine field, is developed by Tedeco, the Lubriclone®, [9] though this sensor is used as part of the magnetic chip detector arrangement whereby an electrical assessment of the material trapped on the plug is made at a predetermined point in operation. Larger deposits of ferro-magnetic material between successive points of operation would suggest incipient failure.

Note that there are a number of other techniques that are used to determine debris in the fluid stream, but these are mainly techniques that are developed for use away from the point of operation and cannot therefore form part of a real time system analysis and so are not considered here.
1.3.5 Electrical

There are a number of electrical system checks that can be applied, but they are usually limited to checks of electrical continuity and are more likely diagnostic (indicative of a failure condition) than prognostic (presaging a failure condition). These checks usually take the form of either open circuit checks or by the use of software to exercise electro motive components as part of a test routine.

1.3.6 Software (real time, data fusion real time and data acquired after operation for trending and abnormality detection analysis)

There are many systems and companies such as Science Applications International Corporation (SAIC) whose business it is to monitor and trend data acquired during system operation.

This type of activity is predominant in the process control and power industries supporting condition based monitoring discussed in section 1.2.3.

Other assets are increasingly being monitored however, such as transportation and trucking services, aeroplane fleets etc [55, 56]. Aircraft despite having built in real time monitoring systems also require trend analysis to determine slower trends of deterioration. A tool that performs such an operation is the COMPASS health monitoring application developed by Rolls Royce [56].

There is also the need to address the possibility of multi dimensional data. Here data fusion and neural network techniques are used to assess the data. The results may not necessarily provide a definitive solution to a particular system exhibiting abnormal behaviour, but the techniques can be used to identify abnormal behaviour in order for experts to apply more conventional techniques to understand that abnormal behaviour [57, 58].

1.4 Research motivation

Understanding how best to focus the development effort in the new Servitisation business model becomes of vital importance for both the Servitisation provider and customer. It is unlikely that either RCM or TPM will provide a complete
solution to all of the questions that Servitisation poses to the business as they are primarily techniques which are focussed on the improvement of the product.

Supportability Engineering offers guidance on many issues related to maintenance in terms of reliability design issues such as maintainability, which predicts the probability that a failed item can be repaired in a specified amount of time [61] and integration of the design with maintenance support activities through a testability engineering programme [62].

The main aim of Supportability Engineering is to reduce the cost of ownership of a product when it enters service and a number of techniques are deployed to support this aim in terms of not only maintainability, testability and availability but also the support systems and infrastructure that provide logistic support through the supply chain to the product in the field.

A key business driver however is availability as this is at the core of the Servitisation offering for customers. For example, in a contract between Vosper Thornycroft group (VT) and the Royal Navy, VT contracted to make its South Atlantic ship and three other patrol vessels leased to the Royal Navy available for use 92% of the time compared to the average time spent in service for other Royal Navy ships of 64% [10].

Availability is defined as: ‘The fraction of the total time that a device or system is able to perform its required function’ [11].

Availability (A) is represented by the expression

$$A = \frac{MTTF}{MTTF + MTTR}$$

Where MTTF is the mean time to failure

MTTR is the mean time to repair

The MTTF is the reciprocal of the (constant) failure rate and the MTTR is the average time taken from the failure of the system to its ability to start up following repair.
It is worth noting however, that in a Servitised business, availability of the product is also influenced by operational circumstances which are in turn dictated primarily by the customer. This aspect is not taken into account by the availability equation which relates only to the reliability of the product in engineering terms.

In addition, the design safety and reliability assessment of a system made as part of the certification process is a measure only of design. Although many reliability issues are dealt with through this kind of assessment, availability factors such as MTTR are not accounted for in this analysis. MTTR may have particular consequences for a Servitised business as the time to repair a component may be complicated by having to organise a co-ordinated maintenance response through customer and service provider organisations.

Determining product availability in a Servitised business is therefore not a simple matter and there are a number of circumstances which influence both technical and commercial issues.

For the customer a defined level of product availability from the service provider along with a reduced cost of ownership brought about by outsourcing the maintenance function provides sufficient motivation to enter into a Servitisation contract. For the Servitisation provider, there is only one way to reduce risk to their business and that is to concentrate on availability issues which are in their control and in terms of the product, this drives a product reliability focussed agenda. If reliability issues are not identified in their own right as factors which would not necessarily cause safety concerns but would cause operational disruption and are not mitigated as such the economic consequences would expose the company to sustained losses throughout the period of the Servitisation contract.

In addition, should the business case require the Servitisation provider to take over functions that were once part of the customer’s maintenance operation matters become further complicated as a full understanding of the customer’s processes is then required.
New processes will have to be developed in order to match or exceed the customer’s current operational support levels. Of particular interest to the service provider in this new model will be the need to develop the quality and timeliness of the flow back of information to the Servitisation provider of reliability issues.

It will also be of prime importance to the Servitisation provider that all aspects of the product failure and failure progression are accurately captured and disseminated along with the operational circumstances that brought about the failure.

In resolving these issues, business activity working relationships have to be redefined not only by organisational restructuring of both the customer and Servitisation provider but also by some form of product improvement strategy or strategies to ensure the product meets its operational criteria. It is highly likely that such a strategy would be delivered through enhanced availability measures.

In order to assess the proposed enhancement in availability both reliability critical components and reliability critical organisational response paths must be identified.

1.4.1 Product availability enhancement

Enhanced product availability measures can take either the form of:

- Increased system reliability (through better quality components or system redundancy);

- Improved Prediction/ detection of System failure using Condition Based Monitoring.

Both approaches have value and may also prove to be costly, but in recent years, prediction and detection of system failure has appeared to be the most cost effective approach as it supports development of a service provision business and is also useful both as an insurance policy to protect against reliability issues that emerge as the product goes in to service and as a monitor for the manner in which the equipment is operated in service.
1.4.2 Consequences of enhanced availability through a monitoring service

Developing a monitoring service to enhance product availability brings new challenges in terms of assessment of the monitoring service, as to perform a full assessment of the fitness of the solution to meet the desired business case will require an assessment technique to extend beyond that of the physical monitoring system, it must also include an evaluation of the ‘reliability’ of the organisational response to physical system failure or impending failure of the remodelled maintenance function.

A failure to holistically evaluate these system aspects will develop a non optimised business model which in turn will lead to:

- Unidentified scenarios of system reliability flaws (where the ‘system’ is both the physical product and the organisational response to failure of that product);
- Unidentified potential threats to the product availability targets for the system as identified above.

Such a situation risks both the customer and the service provider’s business model.

For the customer, lack of product availability could lead to damage to the customer’s reputation and possible litigation or financial penalties in not being able to carry out their business. These risks for the customer are in the most part intangible and very much dependent upon not only their particular business operation but also the circumstances they find themselves in due to lack of product availability.

For the service provider, their risks will be well defined by contract as levels of availability they must maintain or pay a financial penalty. The service provider, however, will in addition be subject to the same risks that apply to the customer in terms of loss of reputation and possibly further risk of commercial penalties if
the customer feels that the service provider has been grossly negligent in their role.

Developing an understanding of the relationship of product reliability and the reliability of the organisational response required to respond to those specific threats to operation is therefore of extreme importance and, it is argued, is the foundation of any coherent Servitisation business model.

1.5 Risk and uncertainty in Servitisation business models

To illustrate further the holistic approach required in order to be able to assess the risk to the Servitisation business model, Deshotels’ concept of the ‘chain’ [12] is introduced in Figure 1.5.1 as a useful visual aid.

![Figure 1.5.1 The ‘Chain’ of major subsystems of a technological system](image)

Here each of the links in the chain supports output volume of production, but the chain will ‘break’ if any link fails. A link may fail because it is not robust enough
to cope with the demand put upon it. Deshotels’ further suggests that the majority of large scale technological systems failures have been caused by breakdowns in the weakest link of the chain which are the human or organisational subsystems.

It is recognised that this is a simplistic model of how businesses are constrained by their constituent parts but when it comes to Servitisation businesses it is argued that the same principles apply. If any one of those identified links in the proposed monitoring system fails, then the Servitisation offering is compromised in much the same way as the volume of output would be in Figure 1.5.1.

Recognising this limitation demonstrates the importance of understanding the true nature of the Servitisation model. The technology aspect of the model is as important in the service response as it is in the sensor systems that acquire and process the monitoring data from the product. Assuring that these aspects of the service are properly linked and optimised is of prime importance.

The organisational aspects of the model apply in the same way, collecting and distributing monitoring data to the correct place at the correct time is as much an issue for hard ware systems that support the monitoring effort as it is for managers ensuring that the data is handled and assessed correctly when it is received.

Finally the human aspects are important in that the monitoring system proposed but be easy to interface with and operate with optimum efficiency. In addition, there may be some failures which require interpretation by skilled operatives. Failure to provide the correct data in a timely fashion to the people who are most able to use it risks a failure of the system as a whole.

If these aspects are not fully understood, then the risks associated with the failure of that model that is being offered will not be fully understood. Developing an understanding of that offering for complex systems such as Servitisation models is a challenge therefore that has to be met.
1.5.1 Causes of risk and uncertainty in Servitisation businesses

It is argued that there will be a variety of causes of risk and uncertainty based on the product and these are highly dependent upon the product itself, its design and its operating environment. The primary causes of risk and uncertainty will depend on the level of knowledge available with regard to each of these aspects and there are a variety of techniques that are used to identify and reduce risk to the product and these are discussed in the following sections.

In terms of organisational response, causes of risk and uncertainty will be similar to those that are identified as causes of risk and uncertainty in project management as both activities involve some form of human interaction to produce their particular deliverables.

These causes are identified by Chapman and Ward [13] as:

- Variability associated with estimates;
- Uncertainty about the basis of estimates;
- Uncertainty about design and logistics;
- Uncertainty about objectives and priorities;
- Uncertainty about fundamental relationships between project parties.

Chapman and Ward identify as contributors to these causes not only variability of hard metrics like cost but also ambiguity in terms of lack of detail, structure, and assumptions regarding a project. These ambiguities are not as tangible as the hard metrics of the project but are equally as important.

This research has similar issues with regard to risk and uncertainty of organisational response. If the business moves toward a Servitisation basis there will be doubts in the early stages of business case development regarding a clear specification of what is required to be delivered in terms of a service and, importantly, how it should be delivered and by whom.

These issues are exacerbated if there is an existing product or organisational structure which has to be re-engineered to meet the new business model. Such
a situation implies defined and potentially entrenched positions on technical authority and responsibility and ownership of issues which may oppose an approach which attempts to develop a more integrated solution to a complex problem.

1.5.2 Managing risk and uncertainty in complex systems

In order to develop an outline solution to this complex problem it must be recognised that both the product reliability and the organisational response reliability are linked and can be considered to be a complex system as inferred by Deshotels’ model.

The model provides a simplistic view of what is a very complex arrangement which is of interest to both systems engineers and to those involved in management complexity research. Systems Engineers tend to try to interpret these situations in terms of systems which can be characterised and defined. This is the approach that will be tested here as some form of assessment is required and a basis for that assessment has to be determined.

Organisations and products alike however, may have emergent properties and as argued by Stacey et al [14] organisations are also capable of self – organisation at local interaction levels which produce globally coherent system behaviour which is aligned with the aim of the organisation as a whole, but is not necessarily designed by intent or consequently recorded as such.

Both emergent and self - organisation properties can therefore lead to distorted or limited views of the complex system where the aggregate activity is non linear and cannot simply be derived from summation of individual component behaviour.

To therefore refine the research goals some kind of process framework is required which enables definition and analysis of such complex systems in order to limit such distortion of the complex system model.

The approach proposed is loosely based on the general decision support process defined in Figure 1.5.2.
Figure 1.5.2 The general decision support process

This general decision support process is suggested by Chapman and Ward [15] as one that aids and informs decision making with regard to risk and uncertainty and is applied primarily to risk in project management.
It is suggested however that the steps provide a general basis upon which reliability assessments can be made on Servitisation business models. In addition, there is also the important recognition that as with project management there is a need for iteration about the decision support process (as indicated by the directional arrows). This iteration process is similar to that required when designing either products or modelling organisational responses and is therefore pertinent.

The ‘capture’ phase in terms of Servitisation businesses is especially important as there are liable to be existing products and organisational constructs that have to be registered.

The ‘focus’ phase would be used in the case of this research to provide understanding of those existing aspects of the business in reference to the new business model.

The ‘model’ phase would be used to provide a visual appreciation of the existing aspects of the business as they stand, providing the opportunity for design alteration and or mitigation of reliability concerns.

‘Test’ the process rigorously translates as far as this research is concerned to testing the model and the assumptions behind that model which, if they come from disparate sources may have contention.

‘Interpret’ the results creatively can be assumed to be the step at which the main uncertainties of the system can be identified for further consideration. Effectively, this equates to an assessment stage proposed by this research.

‘Implement’ the results would be the step at which system redesign is made appropriately and tested further until a satisfactory outcome is reached.

This decision support process is not at present used for reliability assessment purposes of Servitised businesses as it was developed by Chapman and Ward primarily for management of project risks and uncertainties [15]. This research will use this outline approach to develop its argument in order to provide a
framework for the assessment of complex systems which include both physical systems and organisational systems.

### 1.6 Differentiating between complex systems elements

Servitised businesses will consist of system components which, although they may have interacting agents, will be dissimilar in nature. These system components have been defined thus far as the physical system (in terms of product reliability) and organisational systems (organisational response). It is important to differentiate between these system types in order to understand the consequences regarding the reliability assessments that could be applied to each type.

With respect to this research the complex Servitised system can be characterised as having a ‘physical system’ or ‘Hard System’ component and an ‘organisational system’ or ‘Soft System’ component.

A definition of a ‘Hard System’ is one which is based in the systems thinking approaches defined in the period from 1950 to 1980 and is described by Checkland et al as follows [16] ‘Hard Systems’ thinking makes use of the kind of thinking which is natural to design engineers, whose role is to provide an efficient means of meeting a defined need…the design engineers problem is a structured one: there is a gap to be bridged between the desired future state and the present state; how to bridge it is the problem’. This type of system is in other words a physical system that can be represented as a designed product.

Checkland [17] also defines the notion of a ‘Soft System’ as ‘thinking applied to unstructured’ problems’ usually human activity systems which in particular have:

1) An ongoing purpose or mission. In the case of a Soft System this may be a continuing pursuit of something which can never be finally achieved – something such as ‘maintaining relationship’.

2) Has a measure of performance. This is the measure which signals progress or regress in pursuing purposes of trying to achieve objectives.
3) Contains a decision taking process – notionally a ‘decision taker’, as long as this is taken not to be a person but a role which many people in a given system may occupy. Via the decision – taking process the system may take regulatory actions in the light of 1) and 2).

4) Has components which are themselves systems having all the properties of a system.

5) Has components which interact, which show a degree of connectivity (which may be materials, information or influence) such that effects and actions can be transmitted through the system.

6) Exists in a wider system and/ or environment with which it interacts.

7) Has a boundary which separates it from 6) which is formally defined by the area within which the decision taking process has power to cause action to be taken – as opposed to hopefully influencing the environment.

8) Has resources, physical and through human participants, abstract which are at the disposal of the decision taking process.

9) Has some guarantee of continuity, is not ephemeral, has ‘long term stability’, will recover stability after some degree of disturbance. This might be helped from outside the system; it might derive internally from participants commitment to 1).

It can be seen that this description can also be related to Hard Systems and Checkland recognises this by stating that ‘…Soft methodology is seen to be the general case of which Hard methodologies are special cases’ [18].

Checkland also makes an important point regarding Soft System perspective which affects the modelling and assessment of Soft Systems ‘For a system of this kind there may well be as many descriptions of it as there are people who are not completely indifferent to it. This is the characteristic of the real world which forces the methodology to become a means of organising discussion, debate and argument rather than a means of engineering efficient solutions’ [18].
There are a number of techniques that are used to make particular reliability assessments of Hard Systems and Soft Systems, these are discussed in sections 1.7 and 1.8 below. It is suggested that any of these techniques are valid for assessment of complex systems as long as they are used as a component part of the decision support process outlined in Figure 1.5.2 and not purely in isolation.

1.7 Soft System modelling and ‘reliability’ assessment techniques

The most recent source that has been found that addresses the specific issue of Soft System reliability assessment related to service systems and that is by Gong Wang [19]. In that thesis a simulation model is developed of a Service Support System which is then tested for its reliability of operation using Monte Carlo techniques. Wang refers to Product Service System (PSS), as Servitised systems may sometimes be called. There is a variety of literature relating to PSS and PSS attributes as there are a number of varieties of PSS. Baines et al provide a review of the state of the art [59] and find that one of the principal issues is the lack of well developed tools and methodologies that can provide manufacturers with a business wide guide for the implementation of PSSs. This finding appears to support the notion that there is very little in the literature relating to this specific research area at present.

Outside of the PSS research area, the main body of work on Soft System reliability assessment falls broadly into the following category types:

1) Modelling and assessment of Human Machine Interface efficiency

2) Modelling and assessment of processes for quality or management monitoring purposes

In the first category, the aim of the assessment is to optimise the use of specific machinery by an operator. This is not something that is considered for this research as it would require a detailed knowledge of the components of the soft system. This research assumes little knowledge of a proposed Soft System in
the first instance, the aim being to assess the whole of the system rather than component parts of that system.

The second category is of more interest as these types of assessment are used to define the performance of the system as a whole.

It is recognised that there may be overlap in these categories of the techniques discussed below.

Furthermore, in order to perform effective ‘reliability’ assessments on Soft Systems some compromise has to be made as these systems are, as discussed, essentially unstructured. Nevertheless some agreed representation or model of the Soft System has to be developed for an assessment to take place.

Modelling of Soft systems is a well developed capability primarily because of the need to understand organisational relationships within businesses. Due to the nature of the systems being modelled however, and the disparate needs for this type of model, there are a wide range of modelling techniques.

In addition, regardless of organisational structure, there are varying Soft System functions, for example whilst there are simple and repetitive organisational responses which provide a support activity such as computing the payroll, ensuring that stock is kept to an appropriate level, there are other organisational responses that are purely project oriented and so are more abstract and therefore more difficult to model, such as research activity or response to an unexpected failure of the product in service. As the latter type is more difficult to model, it follows that it is also more difficult to assess.

Reliability assessment of Soft Systems is equally problematic as it involves attempting to evaluate the wide variety of human nature, ability and culture.

In the early part of the 20th century human performance assessment focused primarily on speeding up the production process. Since 1941 Human factors research has developed as an application of the principles of experimental psychology. Initially focused on the design of war equipment it has since
developed to become a distinct discipline encompassing a number of sub-areas [20].

Human factors research is based upon person-machine system relationships both are defined as follows 'A person-machine system is an arrangement of people and machines interacting within an environment in order to achieve a set of system goals. The human factors specialist tries to optimise the interaction between people and machine elements of the system, taking the environment into account' [21]. On occasion, the term ergonomics is used. This is defined by Meister (1989) as ‘…the study of how humans accomplish work-related tasks in the context of human machine system operation and how behavioural and non-behavioural variables affect that accomplishment [22].

Much of the focus of Human Factors research is based on human limits in terms of body functions (sensory limitations and physical limitations) and in terms of human reliability, the focus is on human error. Mathematical models of Human Operators have been investigated however in terms of understanding and improving performance, but success is limited due to the way in which Humans process information and their differing capabilities [23].

Another way in which Soft System performance aspects are measured relates specifically to project management and quality system performance improvement. Here Soft System performance is measured through a broad brush approach of business metrics or Key Performance Indicators (KPI’s) which can be based on production output or costs for example. For these organisational metrics to be meaningful however, they must relate in some part (even if is purely a boundary of responsibilities division) to a framework which reflects the organisation and how it behaves. These frameworks are usually, but not exclusively, organisational processes.

It should be noted that metrics can be specific to particular functions and viewpoints within an organisation and can therefore be many and varied and although very different in nature could be a function of the same organisational model (for instance level of resource available and staff turnover could be metrics
applied to the same business function but are not necessarily related). The understanding of the underlying impact of each metric would potentially require a different model of the business for analysis purposes.

This means that there can be no technique which will be capable of entirely capturing the behaviours of an organisation as any organisation is both dynamic in structure (constantly changing methods and procedures to cope with new threats to the business) and is also subject to the vagaries of human behaviour.

This view is supported by Robert Bailey [24] as long as we are dealing with a system level performance measure, human performance can only be inferred…it is so difficult to judge the adequacy of human performance in a large system’.

1.7.1 Soft System modelling techniques

Soft Systems modelling is problematic as not all aspects of the Soft System will be captured by a particular model as the Soft System itself has so many different attributes, which may or may not be causally linked.

The viewpoint of the modeller has also to be considered as the choice of modelling technique used will depend on the requirement for the model in the first instance. For example a Gantt chart may be all that is required to understand the relationship between a number of parties, their sequence or interaction and the associated resource they require to perform their tasks, but this type of model will ignore, for instance, resource capability.

Complex models which include a variety of attributes, some of which may not be obvious contributors to performance levels have a relevance to the Servitised business proposition as by crossing boundaries to deliver a service (in the case of this research a notional one of an Equipment Health Monitoring service) there are not only physical boundaries that need to be defined and understood in terms of hardware platform compatibility; data interface and format compatibility, the so called ‘Hard System’ but also the ‘Soft System’ boundary aspects of human machine interface behaviour and organisation culture.
Though by no means an exhaustive overview of techniques used to model Soft Systems, the techniques listed below vary from those typically used in business to those which are very new in the field and therefore not as widely used.

1.7.1.1 Gantt chart

The Gantt chart is primarily a planning tool and so is not readily identified as an organisational modelling tool. The purpose of the Gantt chart however is to describe not only activities or tasks that an organisation has to carry out to complete its purpose but also has resource attached to those tasks and therefore responsibilities.

The Gantt chart therefore describes organisational activity but not in such an easily understandable manner as the flowchart. An example of a Gantt chart is shown in Figure 1.7.1.

![Figure 1.7.1 An example Gantt Chart](image)

1.7.1.2 Process flowchart

Process flowcharts are often used as a very basic organisation modelling tool. With such a flowchart, the symbols used are similar to those used in the field of software engineering.

The symbols show what actions are expected to be taken, who is responsible for the action to be carried out and sometimes the mechanism for carrying out the action. An example flowchart is shown in Figure 1.7.2.
The process flowchart is used as it is easily understandable and captures the purpose of the organisation at the simplest level.

The problem with such an approach however is that whilst the approach is useful for local processes and procedures, it is not easy to develop an understanding of more complex organisational structures at a higher level of abstraction.

1.7.1.3 Role Activity Diagrams

Role Activity Diagrams (RAD) are a more complex approach to modelling organisational systems.

They are similar to flowcharts with the process flow passing from task to task in the flowchart, once the criteria for continuance of the flow are met.

Role Activity Diagrams were developed by Martyn Ould [25]. They were principally developed in order to aid re-engineering of business processes.

An example of some of the formal notation constructs for RAD is shown in Figure 1.7.3. In this example, there are 2 ‘Roles’ in a process that must act together to produce an outcome. The process starts from the ‘Ready state’. The first task to
be performed is Task 1. This is denoted by the black box in the left hand Role in Figure 1.7.3.

Following the vertical line from the Task box, there is a statement that shows ‘Task 1 completed’ and then an interaction takes place between the two Roles, this could be passing information or control of the process between the two Roles. The interaction is denoted by a white box in each Role that is connected by a horizontal line, the arrowed head shows the direction in which the information or control is passed.

The interaction causes the process in the right hand Role to start as denoted by the hatched box. Following the interaction, there is a box with diagonal lines which shows that the Role will take precedence in an interaction with another organisation that is external to the company for which the process is designed and operated. This again is shown by an arrowed horizontal line which crosses an organisational boundary. Such an interaction however will only take place following an event (unspecified but denoted by an arrow) and a decision based on the event which is conditional (as shown by the triangles annotated ‘Y’ and ‘N’). If the decision is negative, then the process goes back to the state it was before the event occurred and if affirmative continues to an undefined state.

Meanwhile, the process in the left hand Role does not stop following the interaction, but continues to another conditional statement which has specific but unspecified tasks related to each condition.

Following this the process moves on to a ‘phase 2’ which has another conditional statement with unspecified tasks which relate to each specific condition.

Once this condition has been decided, the process moves on to a further, unspecified state.
1.7.1.4 CIM-OSA

The Computer Integrated Manufacturing Open Systems Architecture (CIM-OSA) [26] provides an architectural framework upon which the system life cycle can be defined using a number of different views of the enterprise in question. For example, one ‘view’ might show how the organisation should function, another modelled view for the same enterprise will show how the organisation is structured, etc.
A variety of modelling techniques such as Integration Definition for Function Modelling (IDEF$_0$) and data flow diagrams are used to populate the architectural framework defined by CIM-OSA to develop the complete view of the enterprise.

IDEF$_0$ is used to model business processes in order to effect process change and provides a modelling notation which is undemanding and therefore allows the necessary analysis and communication interaction between technical and non technical staff that is necessary to understand and build complex systems [63].

1.7.1.5 Unified Modelling Language (UML)

This was originally designed as a method for designing physical systems, particularly electronic systems, but has been used to define not only the organisational system but also to determine the requirements that drive that system design [27].

It could be considered deficient in terms of being too complicated a method to use to define these systems but it does by its nature enforce a rigour in developing new organisational models.

It is also problematic in that the systems it was originally designed to describe are deterministic and rigidly structured. As discussed earlier, organisational systems rarely lend themselves to be defined in such strict terms in practice.

In much the same way as CIM-OSA, UML develops a multi-dimensional view of the system process. It is part of the rigour of this language that these views must at least be explored and this can provide a greater depth of insight into the purposes behind the process model than other modelling techniques allow for.

1.7.1.6 Business Process Modelling Notation (BPMN)

The Business Process Modelling Notation (BPMN) is a flow chart based notation used to graphically illustrate business processes which can then be used as a mechanism to develop software code which represents the process. It is an agreement between multiple modelling tool vendors to use a single notation standard.
It provides similar flowchart structures to RADs such as ‘activities’ for tasks, ‘events’ and ‘connectors’ for interactions.

It also maintains the concept of ‘Roles’, these are defined as ‘pools’ by BPMN. A difference however is that these ‘pools’ can be divided into sub elements of responsibility or ‘swimlanes’.

The methodology also allows the modeller to create ‘sub processes’ which can be used to model certain parts of the overall process in greater detail.

BPMN aims to consolidate the best ideas of existing modelling notations into a single standard notation with a well supported tool set. The intention in using this standard is to reduce confusion amongst business and IT end-users [65]. This allows the process models developed by business people to be more easily translated into workable IT execution models.

BPMN also claims that it is methodology agnostic and can support other methodologies such as RADs and IDEF0.

1.7.1.7 Architecture of Integrated Information Systems (ARIS)

The Architecture of Integrated Information Systems (ARIS) methodology has many similarities with both the UML and IDEF0 notations in that a variety of different aspects of the business process are modelled in an attempt to provide a holistic view of the business model. These different aspects are modelled as separate views which are in turn collated together as the ARIS – house of business engineering concept [64].

The methodology is supported by a software toolset which enables models to be built and configured according to the rules of the methodology.

ARIS is, in very much the same way as other techniques reviewed in this section, aimed at describing the business process in software friendly terms. Describing the process in such a way opens up the possibility of being able to provide a software based support solution if required to improve the efficiency of the business process.
1.7.2 Soft System performance assessment techniques

There are a number of Soft System assessment techniques that are used to assess performance, efficiency and error rates. These broadly fall into the two categories of Human Factors performance assessment, basing assessment on the specific interaction of the machine and its operator or a task and its implementer or more broadly in a project management and quality system improvement sense which seeks to determine the measure of specific effects relating to business aims. What follows is by no means an exhaustive list of techniques used in each field.

1.7.2.1 Time and motion study

This Human Factors approach was pioneered in the early 20th Century by FW Taylor and F and L Gilbreth who studied factors which improved the speed at which people did their work by analysis of the way in which people worked. This bought about ‘scientific management’ or optimisation of the tasks they performed, the environment they worked in by timed analysis of those tasks [28]. Although a relatively simplistic approach, this structured form of analysis provides a structured basis for determination of what essential tasks have to be performed and how long they might individually take.

1.7.2.2 Technique for Human Error Rate Prediction

The Technique for Human Error Rate Prediction (THERP) [29] was developed in the 1960’s in order to gain some Human Factor insight into how human error is likely to affect systems in normal operation.

The technique is defined by a series of steps as follows:

1) Determine the system failures that could arise from human error;

2) The tasks performed by Humans in relation to the system functions are analysed and identified;

3) The relevant human probabilities are estimated;
4) The human reliability analysis is integrated with a system reliability analysis to determine effects on system performance;

5) Changes to the system which will improve reliability are then recommended and evaluated.

1.7.2.3 Human Cognitive Reliability

The Human Cognitive Reliability (HCR) model [29] is a Human Factors technique developed in the mid 1980’s in order to assess reliability of behaviour during accident sequences.

This method looks at behaviour based on a time dependent response and makes a judgement on response probability against time elapsed based on the following factors: knowledge of procedures that must be carried out in a certain instance; skill in performing activities in an automatic manner and behaviour in an unfamiliar situation.

A normalised time to respond is developed based on these factors and non response probabilities can be determined over a period of time after an emergency in the system develops.

1.7.2.4 Stochastic modelling technique

This is a method [30] based on the Monte Carlo method of human reliability analysis and was developed by Siegel and Wolf in 1969 and is intended to determine whether an average person can complete a task in an allotted time and the points at which an operator may be overloaded.

Effectively the technique estimates the efficiency of the operator within the wider system based on estimated times to complete each task which are derived by use of the Monte Carlo method.

The method, like the Monte Carlo method depends for its effectiveness on the accuracy of the model of the tasks which form the system.
1.7.2.5 Capability Maturity Model Integration

Work on the Capability Maturity Model Integration by the Software Engineering Institute (SEI) at Carnegie Mellon University began in 1998 and built on previous work regarding capability maturity. The intention was to enable a framework for businesses to effect process improvement.

It does this by providing a model that defines the characteristics of effective processes; this ‘model’ is constructed as a document which addresses different aspects of the process under examination.

The CMMI framework is the structure used to organise the material used to develop models and appraisal methods.

There is particular emphasis on 3 areas of business for CMMI methods, these areas of business (or in CMMI terms - constellations) are:

1) Services within organisations and to external customers;

2) Measuring and monitoring development programmes;

3) Acquisition leadership.

When the framework is populated an assessment takes place using a defined appraisal methodology, the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) methodology, which is used to assess the ‘maturity’ of the organisational processes against defined CMMI levels with specific attributes [31].

This baselines the level at which the organisation currently exists against that defined criteria. The objective is to use this baseline to further develop the organisation to achieve the highest defined CMMI level which means that the organisation processes are at their optimum level and are in a state of continual improvement (level 5). The previous level (level 4) requires that processes are measured and controlled [32].
CMMI therefore is important in as much as it not only provides a methodology for determining the fitness of the processes but also attempts to provide a measure of effectiveness such that they can be improved.

1.7.2.6 Balanced Scorecard

This technique was developed in 1992 by R Kaplan and D Norton [33] to enable organisations to implement their strategy and vision objectives working from 4 distinct perspectives, one of which is a Business Process Perspective. The objectives to be achieved; measures of progress; specific target values and any initiatives that are required to meet the objective are defined, monitored and reviewed in order to ensure that the company is performing as required. If not there is objective information in the metrics to show where performance needs to be improved.

The technique has been bought up to date in 2007 by Hennaburger, Buchman & Economy [34] which introduces a closer alignment of organisation with strategy and allowing a more structured methodology to enable underperformance to be identified.

1.7.2.7 Goal/ Question/ Metric

As with CMMI and Balanced Scorecard, the Goal/ Question/ Metric technique [35] is another technique derived from software engineering in order to help improve the software development process and the resulting product and at the same time align this activity with organisation business and technical goals.

It is used to develop and maintain a meaningful metrics programme for the process. It consists of 3 levels, the highest of which is the conceptual level, that of the measurement Goal, this could be a product, a process or a specific resource.

The second level is the operational level where a set of questions is used to characterise the way the assessment of a particular goal is to be made from a particular quality viewpoint. Several questions can be asked of the goal in order to help characterise fully the goal and each question is then further refined at the
next level, the quantitative level as a metric, there may be a number of metrics which relate to each question.

This results in a set of metrics which reflect the goals of the organisation.

1.8 Review of identified Soft Systems modelling and reliability assessment techniques

1.8.1 Review of Soft System modelling techniques

There are many different types of Soft System modelling technique ranging from the very simple such as the flowchart to the more complicated such as UML modelling.

Each technique has its own strengths and weaknesses but for this research there is a need to use a modelling technique that can:

- Convey as much information as possible about the process (in order that an assessment can take place).
- Extend to ensure that very large Soft System models (that may have to include business processes of other businesses than the Servitisation provider) can be produced.
- Be intuitive to use so that they can be developed by both technical and non technical people.
- Have the potential to be used to model both Hard and Soft Systems.

Clearly flowcharts and Gantt charts are not able to fulfil these criteria as they are limited in terms of the amount of information that they can convey about a process.

Conversely techniques such as ARIS, CIM-OSA and IDEF$_0$ are techniques that supply potentially too much information about the Soft System, some of which may be unnecessary to the assessment need.

BPMN forces an approach that is biased toward helping to provide IT support rather than toward describing human behaviour [66].
UML is a modelling technique that is applied to process modelling, but was originally developed to describe Hard Systems. Although this fills more of the criteria than other techniques it is also a disadvantage as the model has, by the nature of the notation, to be inflexible and therefore not easy to extend. It also has the disadvantage that it is not holistically intuitive as many ‘views’ of the process may be needed to gain a full knowledge of the process proposed.

Therefore the RAD technique is seen as the best potential solution to fulfil the main criteria listed due to its flexibility. The way in which it is able to fulfil these criteria is discussed in more detail in section 2.4 and onwards.

1.8.2 Review of Soft System reliability assessment techniques

The aim of the Soft System reliability assessment is to determine how quickly a process can be completed following initiation. The prime motivation is to understand the time taken and likelihood of achieving a maintenance response to a failure event. In the first instance a notional time is all that is required as there may be contractual consequences if a contractually agreed response time cannot be physically met. It must also be understood that the physical failure mode and detection mechanism will also have an effect on this time, but it may not be possible to account for these effects in early stages of the development of the Soft System.

The best approaches to this issue are therefore accepted to be a time and motion study approach where nominal times are developed or inferred for each task and potentially a stochastic modelling technique such as Monte Carlo analysis to provide confidence in task timings.

The other techniques offer opportunities to measure and improve more stabilised and mature Soft Systems and so they should not be entirely ignored for future improvement work.
1.9 Hard System modelling and reliability assessment techniques

There are a number of analysis and assessment techniques that have been developed in order to gain an understanding of component and physical system safety and reliability. These methods are well documented and are mature in use.

In a purely practical sense though these methods are primarily used to ensure a system is safe and reliable in terms of design (as it follows that if a system’s safety targets are met, then the system is also reliable for all practical considerations).

There are a variety of means available to perform these analyses, the most appropriate to use is dependent upon the nature of the intent of the analysis and the nature of the system under examination. Some of these techniques are discussed in section 1.9.2.

1.9.1 Hard System modelling techniques

Hard Systems differ from Soft Systems in that there is an intentional and structured design need and therefore Hard System design information is more readily available than that for Soft Systems. The reliability assessment techniques use Hard System models which are based predominantly on the design information available.

This design information is either available in drawings, for instance Pipe and Instrumentation Diagrams or in design documentation listing requirements and design criteria and sometimes block diagrams showing system functionality which can be reinterpreted using techniques such as reliability networks for assessment purposes [36].

All of the assessment techniques discussed in section 1.9.2 do not require further modelling effort for the Hard System beyond the information available in the design documentation with the possible exception of the Fault Tree technique
which, by their nature, in addition to performing an assessment role also define models of reliability failure.

1.9.2 Hard System reliability assessment techniques

1.9.2.1 Hazard And Operability studies
The Hazard and Operability (HAZOP) technique is primarily used in the process plant industry and is used to determine mechanisms by which a hazard to safety can occur.

The method involves the systematic and critical examination of the system design and plant operating procedures by a team of specialists.

The design examination is applied using a formal methodology which examines each of the design functions described by the Piping and Instrumentation Diagrams individually and applies defined key words to each component part of the function in order to develop an understanding of the effect of any deviation of the system function implied by the key words.

Any resulting consequences of the function deviation are then documented, evaluated and mitigated if they are deemed to be potential hazards.

This is a necessarily lengthy and rigorous process involving a team of multi-disciplinary specialists.

The technique however provides a means of developing a qualitative early stage reliability assessment in a systematic manner helping to identify and evaluate hazards and their likely frequency of occurrence which has gained wide acceptance by the process industries and regulatory authorities [37].

1.9.2.2 Failure Modes and Effects Analysis
The Failure Modes and Effect Analysis (FMEA) procedure is also a formal mechanism for recording the consequences of function or component failure. The approach is standardised and defined for aerospace applications by the Society of Automotive Engineers in [38].
In the component failure analysis technique, which tends to be a ‘bottom up’ approach, each component is evaluated individually to determine the effect of failure and hence the suitability of the design as a whole once the study is complete.

The evaluation each component of the design is usually combined with a failure rate for the component and is recorded formally on a work sheet which constitutes the FMEA [39].

The FMEA analysis itself is an analysis of the design which is performed to establish a number of ways in which a design might fail, which components are associated with those particular failure modes and the consequences or effect of those failures on the design.

If a system is under evaluation, then it is often the case that the information from a number of FMEA’s which represent sub-systems will be combined in order to assess the critical effect of each on the system as a whole. Such a technique uses an extension to the FMEA methodology and is called a Failure Modes and Effects Criticality Analysis (FMECA).

This extension uses the same format as an FMEA analysis but considers and ranks the reliability of the system as a whole based on the combinational effects of the data supplied in each sub-system FMEA.

Such a document therefore provides details on which components are the root cause of system failure and their respective criticality. This analysis is by its nature an exhaustive one and the number of failure modes for large systems can be numerous, resulting in a large document from which it is difficult to develop a holistic view of the system.

1.9.2.3 Fault Trees

Fault trees can be used in both a ‘top down’ and a ‘bottom up’ assessment mode. The ‘top down’ approach acknowledges a known or perceived hazard to operation and the bottom up sense is used when a particular design function is to be tested for failure consequence.
The fault tree is constructed from logical operators (AND, OR etc) which are applied to a stated condition and which then define another condition that results until either the final hazard condition is reached in the case of the ‘bottom up’ approach or the causal effects of the top event are reduced to the lowest component consequence level in the case of the ‘top down’ approach.

Many fault trees may need to be constructed in order to define a complete system and unlike the HAZOP and FMEA approach there is no guarantee that all parts of the system will necessarily be evaluated as there is no requirement to examine each component part in this methodology.

The coverage of the design and the efficacy of this approach are therefore highly dependent on the ability of the assessor. That said, ‘this approach is the most important and most frequently used of the methods to quantify system performance as it not only provides a means for system quantification but also a diagrammatic representation of the system failure causes, which is ideal for communicating the failure logic to other personnel’ [40].

1.10 Review of identified Hard Systems modelling and reliability assessment techniques

1.10.1 Review of identified Hard System reliability assessment techniques

The ideal Hard Systems models for this research would be the designs such as pipe and instrumentation drawings as they are developed for production. Developing a new Hard System model adds to designer workload.

It is however important that the design information held in drawings and design documentation is interpreted as a model for Hard System assessment. This is because there is a need for a focus on the monitoring sensors and their ‘duties and responsibilities’ in terms of monitoring components that could strongly affect the availability condition of the product. In some cases the sensors may have only a minor control role but have a major impact in determining or corroborating
component failure. Such a role has to be recognised by the design team in order to ensure that the monitoring component itself is sufficiently reliable and accurate for the dual role it is given.

No current Hard System modelling technique is able to fulfil this distinct role primarily because the focus is the design of the system as a whole rather than a focus on one particular aspect of it. It is felt that for the purposes of this research RADs may be able to fulfil such a role. This is discussed in more detail in section 2.4 and onwards.

1.10.2 Review of identified Hard System reliability assessment techniques

The Hard System reliability assessment techniques are mature techniques and as such should be used where possible to support the assessment technique. Fault tree assessment is not a technique that can be used in this research. This is because both the HAZOP and FMEA approaches by necessity include an analysis of the system as a whole. With fault trees, however, there is the possibility that some aspects of the system either may not be analysed or it may difficult to ascertain that the complete system has been assessed through the various fault trees that are produced.

In principle therefore any Hard System analysis should use information directly obtained from the FMEA. If in addition a HAZOP is performed on the Hard System under assessment then any further information obtained from that analysis should also be considered where possible.

The intention should be that this research should not require further analysis to be performed on the Hard System under assessment therefore requiring more work to be undertaken. Instead, the intention is that that the Hard System model will present the information from the system design assessment in a way which makes it easy to interpret with respect to the key issues of reliability monitoring system coverage and effectiveness. These issues determine the potential of the
Hard System aspect of the reliability monitoring system to deliver increased availability of the product.

1.11 Aims and objectives of this Thesis

The principle motivation for this thesis stems from the authors background in developing health management equipment to support the Servitisation of an industrial product.

Servitisation has the capacity to affect all aspects of the business from initial product design through to in service care of the product and all processes that support those efforts. This wholesale effect on the business stimulates many and varied business responses that have to be assessed and addressed in order to ensure alignment with the company’s Servitisation strategy.

There is a danger that commercial responses take precedence in this assessment phase as they include important business issues such as whether any investment should be made in Servitisation of the business and if made, what is the likely return on investment. Such judgements help to clarify and set the framework for the proposed change to a Servitised business model however, whilst such issues are important, they should not be made in isolation from the product that is being Servitised.

The essence of a Servitised business is the robustness and quality of the service that maximises product availability for the customer whilst at the same time minimising the service provider’s costs in maintenance effort. A key element which enables both product availability and ensures service quality is the reliability monitoring system which monitors the product’s health and informs the maintenance response to failure through monitoring of the product’s condition.

The reliability monitoring system is therefore a function both of the product (in terms of how likely it is to fail and whether that failure can be predicted) and of the organisation which responds to a detected failure condition (in terms of whether that organisation can respond in time to a failure or impending failure in order to mitigate any product availability issues). Capability shortfalls in either
the failure detection mechanism or the failure response mechanism will result in poor quality service and could potentially result in a financial loss if product availability cannot be maintained to contractually agreed limits.

The aim of this thesis is therefore to develop a methodology for a reliability assessment of both Hard and Soft System aspects (product and maintenance response system aspects respectively) of a reliability monitoring system. The purpose of this assessment is to identify and assess any likely capability shortfalls that will need to be addressed for a company to move towards delivering a Servitised business model.

A review of current reliability enhancement methodologies and the techniques used to support Condition Based Monitoring which requires reliability monitoring system technologies to support its implementation has been carried out.

Further to this, a review of research literature in the areas of Hard Systems and Soft Systems modelling and reliability assessment has been carried out.

The main objectives therefore are:

1) To investigate the development of a methodology of reliability assessment for both Hard and Soft Systems in order to inform the system design.

2) To include in the methodology a modelling technique which can be applied to both Hard Systems and Soft Systems elements of a reliability monitoring system developed for a Servitised business.

   a) The technique must also:

      i) recognise and accommodate the differences between diagnostic and prognostic reliability monitoring;

      ii) be capable of being used to inform design decisions in addition to reliability assessment;

      iii) recognise the nature of reliability monitoring systems in terms of their Hard and Soft Systems components;
iv) be capable of developing a model from a variety of system description source information.

3) To include in the methodology an assessment technique for Hard System and Soft System reliability issues based on the modelled systems.

4) To investigate the potential of integrating the developed modelling and assessment techniques with existing assessment techniques and procedures with the intention of reducing the amount of disturbance to conventional design and assessment activity.
Chapter 2 – Developing and defining the methodology

2.1 Developing requirements for a methodology to assess Servitisation systems

There are a number of issues to be defined and resolved in order to provide an effective methodology to assess Servitisation systems. These issues have to be defined in order to set the boundary of the assessment method. As has been seen from the previous chapter, there are a wide variety of methods available which could be and are used to evaluate component parts or aspects of Servitisation systems, be they Hard or Soft Systems, but not a method that could be used to evaluate the system holistically.

It may be that this is purely because Servitisation systems have such diverse attributes due to their nature of composition, comprising both Hard and Soft Systems, but an overall understanding of the system that is to be developed to implement the Servitisation business strategy is very desirable as failure to understand reliability aspects of these attributes will compromise the business strategy.

In order to make the judgement that the business strategy is not ‘at risk’, some level of understanding of the system capability as a whole is required.

2.2 Key requirements for a methodology to assess Servitisation systems

To address the aims of this thesis, outlined in section 1.9 a framework of key requirements is needed. These requirements will bound the issues that will have to be addressed in order to develop a framework methodology for assessment of Servitised systems.

These key requirements are notional ones derived from issues discussed in section 1.4 relating to Hard Systems and section 1.5 which relate mainly to Soft
System issues. The requirements are also developed to respect some principal aspects of current modelling approaches for Hard and Soft Systems which are listed in section 1.7 and 1.8.

For a business, the change to a Servitisation business model will affect how the business operates as evidenced by Vandermerwe and Rada [1] and Dwyer [2]. In order to understand the likely impact on the business an understanding of the product being offered and the system that currently supports that product in the field has to be made. This understanding has to be agreed before any operational changes which implement a Servitisation strategy can be contemplated. To do otherwise would go against the arguments put forward by Chapman and Ward [15] in reducing project uncertainty. Failure to either reduce uncertain issues or to clarify an understanding of the current business situation may in turn lead to long reaching commercial consequences for the business. For instance, are the current systems that operate the support service fit for purpose? If not, should they be replaced wholesale or can they be adapted or upgraded to support the proposed strategy as discussed in section 2.2.6.

Central to this understanding is the capability of the reliability monitoring system. As this capability has wide reaching effects throughout the business, there is the question of who is responsible for the system at any point in time. This is a vital aspect of the reliability monitoring system and is as important in the design stage of the system as it is in the operational stage and is discussed further in section 2.2.1.

As important are the key components of the reliability monitoring system; how they are connected; how they work together to deliver the desired outcome and how they relate to the responsibilities of those who are responsible for the reliability monitoring system. These aspects are discussed further in sections 2.2.2 to 2.2.5 inclusive.

Finally an appreciation of the capability of the reliability monitoring system is required; this is discussed further in section 2.2.7.
The requirements listed in sections 2.2.1 to 2.2.7 inclusive are developed to act purely as a guiding framework to develop a methodology aimed at building a modelled understanding of the reliability monitoring system in order that an assessment can be made as to its fitness for purpose.

Note that the methodology requirements discussed here are not intended to preclude use of other methods discussed in the previous chapter. The assessment method used should reflect the state of maturity of system design and the specific needs of the assessor.

2.2.1 Key requirement 1 – Identification of system ownership

A Servitisation system is developed and operated under a variety of constraints imposed by different aspects of the business which will develop it. These constraints may be for example, financial, physical, technological or a number of other issues which affect the manufacture and operation of a system.

Any of these constraints may seriously affect the performance of the system in operation if not recognised and mitigated in a coherent manner. In order to be able to perform this mitigation, ownership of each system aspect has to be determined.

It must be recognised that in some instances there may be system aspects that either are not recognised or will be developed as new entities when the system is at the conceptual stage or may involve a cross functional ownership issue.

The effect of each system owner’s contribution is therefore a key factor in assessing system operational reliability and must be understood as fully as possible in order for mitigation activity to take place.

The first key requirement therefore is that the system owners must be clearly recognised and represented by the methodology.

2.2.2 Key requirement 2 – Identification of system components

It is necessary when modelling a system to understand of what that system comprises.
In terms of Servitisation systems, it is likely that the system will comprise not only of physical system components but also physical tasks (such as monitoring system components for failure) that are required to be performed in order to ensure the availability of those physical system components. These physical tasks may require either human or computer interaction or a combination of both.

Some of the methods of physical systems component representation for the purpose of analysis and many ways of describing human task or computer interactions with the system were discussed in the previous chapter.

The second key requirement is therefore that the methodology must be capable of representing the principal attributes of the system, whether these are physical components or physical tasks.

2.2.3 Key requirement 3 – identification of interfaces

One of the most crucial areas of system design is that of interface management. Poor design and management of these interfaces and system dependencies can lead to system failure or poor overall system reliability and in addition, exposure to considerable commercial risk.

As with the section above (identification of system components) interfaces can be physical or could be through human interaction. The notation should therefore be able to describe both of these types of interface and any other such interface that becomes apparent.

The third key requirement therefore, is that the methodology must allow for interactions and dependencies between components and activities to be made clear.

2.2.4 Key requirement 4 – component context, scheduling and allocation

When modelling any system it is important to understand not only context but also if there is any defined order of operation of that system. Contextual information helps to provide a more in depth understanding of the system.
In many modern systems, components of systems may operate concurrently and may also use system redundancy to compensate for failure or unreliability.

If human action is required to facilitate the system operation then this may not be the case. Many systems do however require some aspect of human interaction at some point in their operation cycle and the ‘right’ number of people with the ‘right’ skills and authority acting at the ‘right’ time will be a factor in the reliability of the system as a whole.

The fourth key requirement therefore is that the methodology must be capable of representing context, timing or sequencing of system components where this is expected to affect reliability. It should also be capable of referencing resource allocation where that is seen as critical to the success of the operation of either type of system. Resource in this context refers either to a human interaction requirement such as the level of support available for the required level of activity or in physical system terms the availability of a motive source such as a fluid power reservoir or electrical generator.

2.2.5 Key requirement 5 - system context and hierarchy

When dealing with complex systems design, the procedure often adopted is one of compartmentalisation and/or development in a hierarchical manner. This approach is used in order to reduce the number of design issues to be resolved when designing large scale systems.

The system under development is, in effect, partitioned into functional ‘black boxes’. Resolving the characteristics of the ‘black boxes’ leads to an informed and hopefully, modular design.

Of course, overall system performance must still be maintained and this approach is optimised using Systems Engineering techniques such as case study trade-offs. Such trade off studies can be performed using system models. An example of such models developed using UML is discussed in [41].

It is recognised that in some cases, this approach could have either a positive benefit or a negative effect. For example, if the ‘black box’ functionality is set at
too high a level for accurate resolution of reliability assessment the technique will be at a disadvantage. By the same token, taking this approach may enable a reliability figure to be assigned to a black box based on ‘intuition’ with very little evidence, if the system has not been in operation before. This could be advantageous in constructing an overall reliability figure. Of course this approach would require revision and amendment once the system came into operation, but at least a reliability figure with an appropriate level of confidence could be provided.

This approach must be recognised as a ‘fact of life’ by the methodology and be sympathetic to it. The fifth key requirement therefore is that the methodology should be sympathetic to system partitioning and hierarchical constraints.

2.2.6 Key requirement 6 – ability to integrate legacy systems

It is expected that Servitisation systems will come into being through a change in direction of business strategy. This would mean that there would be existing physical systems and organisational support structures that would need to be assessed or integrated into the new business model.

The methodology should therefore be capable of representing these systems and their current role in the business, such that an assessment can be made for restructuring or improvement in order to develop a Servitisation model that meets the new needs of the business.

The sixth key requirement therefore is that the methodology should be capable of providing representation of current systems.

2.2.7 Key requirement 7 - assessment technique

The assessment technique may differ depending on the system attribute but it must be capable of identifying system weaknesses that affect the reliability of either that particular attribute or combination of attributes such that there is a either resulting unavailability or degraded system performance which infringes contractual agreements of the Servitisation system as a whole.
The seventh key requirement therefore is that the methodology should provide an assessment means which identifies system weakness in reliability and availability terms.

### 2.3 Discussion of methodology requirements

The methodology requirements indicate that before any assessment takes place an integrated representation of both the Hard and the Soft Systems which make up the Servitisation system is desirable to ensure that all key requirements are met and the key aspects of the Servitised system as a whole are understood. If this is not achieved then there is a consequent risk to the business model of the Servitised system as contractual guarantees guarding against unavailability of the Servitised systems are not underpinned by a formal and supporting analysis.

Therefore a modelling notation capable of representing both types of system and which is also capable of showing a method of integration of those representations has to be devised.

No such modelling notation currently exists as using a common modelling notation set to describe both Hard and Soft Systems has not been found to be necessary before the instigation of Servitisation systems as Hard and Soft Systems were considered as distinct design entities in terms of the operation of a product in the field.

A common modelling notation set is also difficult to achieve. This is because the Hard System is produced with definite constructs and functional components whereas the Soft System, however it is defined, can never be definitively modelled due to its nature. What is meant by this is that any Soft System that involves human interaction has the potential to deviate in its behaviour from time to time and may therefore compromise the model.

The diverse nature of Hard and Soft systems means that different modelling techniques are applied to each the choice of which technique is used relates not only to developing a representation of the system of interest but also to the particular aspects of the system that are of interest of the modeller. Some
modelling techniques are developed in particular to highlight specific system aspects, such as the nature of connections in a pipe and instrumentation drawing.

Another barrier to defining a common modelling notation is the complicated nature of Servitisation systems. By their nature they are large and complex entities in the first instance and their response to events may therefore not be easy to describe and model. Such considerations impact on the boundary of interest of the Servitisation system.

The core of such a system and the subject of this research is the system that forms the basis of the business model of Servitisation and that is the reliability monitoring system.

This reliability monitoring system is an integral part of the Servitised system which is used to detect and warn both the customer and Servitisation system operator when there is likely to be unavailability of the Servitised system. The response to the detected unavailability event is initiated by the Servitisation system operator. A resolving action to the event would result from a negotiation between the Servitisation system operator and customer in order to ensure the minimum disruption to service.

When evaluating reliability monitoring systems it has to be recognised therefore that the Hard System aspect of the monitoring system does not form the boundary of the reliability monitoring system, this boundary has to be extended to include the Soft System aspects also as the actions of the Servitisation system operator and their corresponding functions have as much of an impact on the availability of the system as the failure that has been detected. That is if an appropriate repair or maintenance action is not delivered in a timely action system unavailability is the result.

From the previous chapter, it can be seen that the usual approach for Hard Systems safety and reliability evaluation is to use the currently available design documentation to perform the evaluation.
It should also be noted that at present, although Soft Systems are alluded to by current techniques as being a necessary part of the maintenance cycle, those techniques do not recognise the Soft System as a necessary part of any assessment of a reliability monitoring system and are therefore not modelled as such.

It has to be recognised therefore that any modelling notation that is developed for Servitisation systems will either not be the primary means of defining the system (in the case of Hard Systems) or will define a ‘new’ aspect of the system (as is the case for Soft Systems).

As the objective of this notation is to provide a platform for a holistic assessment of both Hard and Soft Systems, it follows that this approach implies notional extra cost and work above and beyond that of conventional system definition techniques. It is therefore important not only that the required notation is one that will provide an overview of both types of system which is detailed enough to provide a system assessment but also will enable information from a number of sources to be integrated into that view in order to reduce duplication of effort.

There is therefore the choice to either develop a modelling methodology for both Hard and Soft Systems or to sustain a conventional approach and develop an evaluation technique for the Soft Systems which complements that of the Hard System.

Evaluation techniques in general have the disadvantage in that their effective application is primarily dependent on both the skill of the person performing the evaluation and that person’s knowledge of the system under evaluation. Such an approach is therefore open to misinterpretation with very complex systems such as reliability monitoring systems.

Providing a notation that links both Hard and Soft Systems and will define such a diverse and complex system seems therefore to be to most appropriate approach.
It should be noted that although desirable, it is unlikely that a completely ‘seamless’ notation covering both types of system will be achievable as the ‘failure rates’ of Soft Systems and Hard Systems by intuition are likely to differ, potentially by orders of magnitude. In addition each systems attributes vary from each other by definition.

What is required is a scaleable system overview, which can show the interactions between Hard and Soft Systems in order that appropriate assessments can be made.

This system overview must be capable of either modelling or integrating representations of current systems in order that these systems can be re-engineered to suit the Servitisation business case.

From the review of techniques listed in the previous chapter, the modelling notation that provides the most flexibility in its format to accommodate these system overview requirements is considered to be that of the Role Activity Diagram (RAD).

This notation is seen as sufficiently flexible to interpret both types of system adequately.

In terms of assessment, the notation also provides an opportunity to examine important features of both Hard and Soft Systems in terms of component context and interfaces which (as evidenced by the techniques used for evaluation listed in the previous chapter) are prime aspects of system failure which have to be assessed.

2.4 The Role Activity Diagram (RAD) notation

The Role Activity Diagram (RAD) notation is one that is, by necessity, very formal as it was developed by Ould [42] in order to re-engineer businesses by describing the business processes in order that they could be either changed or developed.
It is a notation, which is in effect an enhanced flowchart. It has a formal method of operation, where ‘control’ of the operation of the flowchart is passed between elements in the flowchart by a notional ‘token’. The token can only be passed to a following element once the conditions for the previous element have been met.

This reflects the operation of Petri Nets [43]. Role Activity Diagram notation is essentially an enhanced and expanded form of Petri net modelling. The primary difference in approach is that the RAD notation also requires recognition of business process roles, which can be ‘created’ and ‘destroyed’ as a process life cycle is enacted. This is an aspect that is not normally associated with either flowcharts or with Petri nets.

Another important aspect of this notation is that it requires that process roles are identified which ‘own’ specific tasks. This feature removes ambiguity regarding responsibility ownership for a particular task. The role could be a person or could be a function within an organisation, for example, a chief engineer. That chief engineer may have many responsibilities and tasks to perform. Such a task may be ‘plan project’. It is unlikely such a task would be carried out purely by the chief engineer and so other roles would be assigned in order to help with that overall task and each would have specific tasks which they would have to perform in order to support the task. An example would be setting a project budget. The company accountant, the main board and the chief engineer would all have to take part in this process and would all have specific tasks relating to their role in order to achieve setting the budget. The Role Activity Diagram would be able to show specific Roles, their specific tasks and their interdependence and the order of sequence in which they must be carried out to achieve the objective.

In the notation Roles can be considered to work concurrently as timing is not a consideration, but sequencing is. For instance, no task may start until the task preceding it in the same Role has completed or, if ‘control’ of the sequence is restricted by input from another Role.

The notation has the further advantage that all parties involved in the business process being described have ready access to information regarding who they
interact with and how reliant their tasks and responsibilities are on those other Roles.

To illustrate, an example is given in Figure 2.4.1. In this example there are two Roles, one deals with Airframe data and is noted as such, the other with data collection. The first task that is performed is done so by the Airframe Data Role. Here the first task is to format the flight data. When the Format Flight Data task is completed, the data is then ‘Harvested’ by the second Role. Once the data has been harvested, the Data Collection Role performs failure prediction checks. If this task is completed and an alarm is raised, then an advice request is passed to an as yet undefined Role and the process completes.

![Role Activity Diagram](image)

**Figure 2.4.1 An example Role Activity Diagram showing an advice request following a failure alarm**

### 2.5 Reasons for modifying the Role Activity Diagram approach

In previous work [44] an attempt was made to assess the Role Activity Diagram notation in a practical context and to enhance the capability of the notation. In particular it was found that Role Activity Diagrams had the following drawbacks:
1) The diagrams became complex when describing processes in detail and level of process ‘granularity’ was an issue;

2) When modelling complex processes, for example when modelling processes across functional or departmental boundaries the number of diagrams required to reflect the process increased dramatically and did not help with an overall understanding of process flow;

3) The notation set although simplistic, may still not be accessible to all, as would be a flowchart, therefore rules which helped simplify the notation would be required.

This research intends to show that the reduction of the RAD notation set to a necessary subset will allow the notation to maintain its essential benefits and integrity but will reduce complexity and increase flexibility at the same time.

It was proposed by Murdoch et al [45] that the essential components of any process should be addressed by the notation. These essential components were judged to be ‘who’ was responsible for enacting a specific element of the process ‘what’ was required to be completed in that element and ‘how’ it was proposed that the required action should be completed. This was assumed to be the case as this matched with the prevailing standard of process definition of the company for which the research was carried out at the time.

Equating these essential components with RAD notation terms was accomplished as follows:

- ‘Who’ – the ‘role’ which equates to the named grey box in the example shown in Figure 2.5.1;

- ‘What’ – the ‘task’ to be performed which equates to the annotated black box in the example shown in Figure 2.5.1;

- ‘How’ – the ‘information exchange interface’ which equates to the annotated white box in the example shown in Figure 2.5.1.
A conclusion of this work was that often the ‘who’ and the ‘what’ aspects of process were adequately covered by existing company standards as responsibility for the completion of the task was often explicitly stated in the case of ‘who’ and a description of the task and guidelines relating to how it should be produced were often present for ‘what’ has to be produced but the ‘how’ aspect in terms of how the information (including what type of information) is passed between roles was often either neglected or poorly defined.

The ‘how’ aspect is often a crucial factor in the success or failure of a process as it represents interfaces between roles. If any interface is not clearly defined (as a minimum in terms of how do I pass on my task output and to whom) the process breaks down.

This is not a dissimilar position to that taken when developing a fault tree analysis for a Hard System where one of the legs of the fault tree is either unquantified or in the worse case, not identified.

Whilst this notation was seen to fulfil the requirements for Soft System modelling, a method for applying similar style notation to Hard Systems was needed in order to attempt a seamless mapping of Hard and Soft Systems which is the aim of this research. The RAD modelling notation appears to offer the both flexibility to model both system types in addition to providing the necessary clarity of system definition required in order that both types of system can be understood and assessed. In addition the notation is open to modification such that the definition
of the essential aspects of the system are not lost when the size of the system increases and the system can be modelled at a variety of levels of abstraction according to both the specific needs of the assessor and stage of the design process.

It is for these reasons that the RAD notation was chosen above any other present technique for this research.

2.6 The concept of extending the RAD notation from a Soft Systems mapping notation to a Hard Systems mapping notation

In order to support a holistic assessment technique, a seamless mapping of both Hard and Soft Systems attributes is desired. Therefore an understanding is required of the generalised common attributes that exist between both Hard and Soft Systems and in particular those attributes that will be pertinent to Hard and Soft System assessment.

2.6.1 Mapping Soft System attributes to Hard System attributes

Developing what is essentially a Soft System notation to one that could be used to represent a Hard System is done by maintaining the sense of the modifications to the reduced Soft System notation suggested above but allowing those modifications to represent different, but functionally similar, system attributes.

This may appear to be taking liberties with the notation, but in analysis terms, the same system aspects are under consideration for analysis in the Hard System as are with a Soft System. These similarities are demonstrated in Table 2.6.1.
The system attributes for Hard Systems are denoted in this way in order to ensure that they relate in the same way to the Soft System attributes (modified RAD notation primary Soft System attributes are shown bracketed in the table).

Thus a physical system or sub system plays a major ‘role’ in the Hard System as a whole in the same way as a system stakeholder is for the Soft System. In this way the principle of ‘who’ is responsible for specific areas of the design is established.

It follows that Hard System components exist within the boundaries established within those areas of design responsibility in order to perform defined actions or ‘What’ has to be accomplished within that design area of responsibility. In naming convention terms it is difficult to relate such a concept to a Hard System as the term ‘task’ is loaded, it could be taken to relate to a function, but the aim of the notation is to determine physical design features (components) rather than more abstract functional concepts.

All components within the Hard System have interfaces, both within the ‘Who’ boundary and external to that boundary. It is these external interfaces that are of most interest as they have reliability implications if they are either poorly defined

<table>
<thead>
<tr>
<th>System attributes</th>
<th>Soft System</th>
<th>Hard System</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘Who’ (role)</td>
<td>A system stakeholder</td>
<td>A physical system or sub system</td>
</tr>
<tr>
<td>‘What’ (task)</td>
<td>A task that stakeholder must perform to ‘enable’ the system</td>
<td>A component of that system or subsystem</td>
</tr>
<tr>
<td>‘How’ (interface)</td>
<td>How the stakeholders interface</td>
<td>The physical interfaces between components of the system or subsystem</td>
</tr>
</tbody>
</table>

Table 2.6.1 Mapping System attributes to Hard and Soft Systems
or not well understood. It is ‘How’ these particular interfaces and the particular components in each sub system to which they relate, connect the system to ensure correct operation of the system that is of importance when assessing the system as a whole.

2.6.2 Modelling notation level of abstraction

The purpose of this aspect of the research is to arrive at a common notation that reflects the important facets of the system and can lend itself to application of current techniques for reliability assessment.

One of the most important aspects of any model is the level of abstraction of the model and what techniques are available for that level of abstraction.

This aspect affects not only what assessment techniques can be applied but also determines the boundary of the system assessment.

The notation therefore must be capable of handling different levels of model abstraction. That is it must be as capable of modelling detailed, complex systems as well as simple systems. It should therefore be as capable of modelling an aircraft and its systems as a single Line Replaceable Unit (LRU) of that aircraft system as it would be for Soft Systems in modelling an organisational response to receiving a purchase order to fetching a component from a warehouse.

For Hard Systems modelling the approach adopted is to concentrate on a modelling level of abstraction that relates to distinct systems and subsystems.

The exact approach of course is very much dependent on the system being modelled. One of the main aims of the research though is to apply this technique to complex systems. The definition of a ‘complex’ system in this particular instance is that of a system that has many subsystems that are required to act in unison to deliver the system purpose.
2.7 Considerations when applying the notation to Soft Systems

Soft Systems are by their nature more abstract than Hard Systems regarding perception and purpose as there is the potential for these systems to have a number of potential uses and dimensions (for example such a system could provide information regarding resource costs, resource required to do work as well as actually defining the tasks to be performed) and so there is the potential for there to be disagreement about purpose, boundary etc.

In addition to the obvious difficulties in modelling this type of system there are also many ways in which such systems can be represented. As discussed in the previous chapter, there are methods of direct modelling such as flowcharts and methods of indirect modelling of these systems using planning tools like Gantt charts. These methods are discussed in more detail below.

2.7.1 Further discussion on Gantt charts

Essentially, for the purposes of this research, planning applications such as Gantt charts can be used as source material for Soft System modelling. Although Gantt charts do not account for aspects of culture (where it is possible that political sensitivities are reflected regarding the contribution of certain departments), they do provide information on the actual process and process metrics of the business and show how the organisation actually behaves rather than how it is ‘expected’ to behave as specified by departmental procedures.

The simple Gantt chart shown in Figure 2.7.1 describes how an organisation may respond to a maintenance event indicating a fault that has been produced by a monitoring system.

The chart is read from left to right and shows tasks that must be performed (represented by the blue bars in Figure 2.7.1), in the sequence in which they are performed to bring the response to the maintenance alert which indicates and confirms a known failure before scheduling and making a repair to that equipment. The diamond shapes on the Gantt chart are the deliverables of the process; that is the process outputs.
In this case, a new task is not started before its previous task is completed. This is not always the case, but is shown as such here to provide an understanding of the technique adopted in this research. On occasion, a task may not be started before a deliverable is either generated or received.

<table>
<thead>
<tr>
<th>ID</th>
<th>Task Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Failure detected</td>
</tr>
<tr>
<td>2</td>
<td>Failure alert issued</td>
</tr>
<tr>
<td>3</td>
<td>Failure investigation</td>
</tr>
<tr>
<td>4</td>
<td>Provide maintenance schedule</td>
</tr>
<tr>
<td>5</td>
<td>Undertake repair</td>
</tr>
<tr>
<td>6</td>
<td>Confirm repair complete</td>
</tr>
</tbody>
</table>

**Figure 2.7.1 An example Gantt chart representing a maintenance action**

Such a chart provides a great deal of information on the tasks in hand. These charts can however be limited in terms of providing an understanding of the process that lies behind the actions that need to be taken and do not necessarily indicate who has the responsibility to complete, validate the completion of, or manage the task described.

For complex programmes the charts become large and often difficult to assimilate especially if there are a number of dependencies between tasks or organisational boundaries.

**2.7.2 Further discussion on flowcharts**

A popular method of defining organisational behaviour is the flowchart which is often used in process quality management applications.

In this case, the flowchart is used to show how typically someone should perform their tasks which are written as process steps (represented by the rectangles in Figure 2.7.2). The direction of ‘flow’ between and sequencing of these process steps is shown by arrowed lines. The process stops when it encounters a process end box (represented by the rectangular boxes with semicircular ends).

This type of representation does not easily lend itself to a planning function because it includes decision gates (represented as diamonds in Figure 2.7.2)
which offer alternative routes for the process to follow. Such a process step is difficult to show on a Gantt chart.

It is feasible and even desirable however for various aspects of the planning process to be derived from such a flowchart.

In the example in Figure 2.7.2 the flowchart describes the action required and the sequence in which those actions are performed when dealing with a predicted failure alert.

In this case, there is a need to warn the service scheduler of a potential failure and maintain a ‘watch’ on the system being monitored to confirm that the expected failure progresses in the manner that has been identified and catalogued for this type of failure.

It should be noted that the event here is defined as ‘abnormal behaviour’ for the system which effectively refers to an identified precursor event and not an event which is indicative of an actual failure.
Figure 2.7.2 A flowchart showing response to a predicted failure event
Both of the above mentioned techniques are equally valid techniques for representing organisational behaviour, but there are differences in the way these representations are made. Both also have their limitations, in terms of representing the type of information that is required for a reliability analysis and also for reflecting organisational structure and hence structural deficiencies.

A method is therefore required which can model organisations using the type of information contained in the Gantt and flowchart representations and which also addresses the other factors mentioned above.

The following approach is suggested. Note that this approach is similar to that of the Hard System modelling approach. This is no coincidence as it is an aim of this research to develop a position from which both Hard and Soft System artefacts can be combined in the same representation. Of course such a representation is only valid in a reliability sense if the ‘reliability’ attributes of both systems are numerically of the same order. For instance, safety critical Hard System reliability could guard against serious events occurring only once in a number of million hours of operation where the probability of a serious event occurrence in a human centric system could be considerably less.

2.7.3 Discussion of the principal aspects of the Soft System modelling approach

As discussed earlier in the key requirements, the most important aspects to be taken into consideration when modelling business processes or Soft Systems are:

1) Who performs the system roles and what is the boundary to these roles;
2) What tasks must each system role perform;
3) How do system roles interact to perform these tasks and how are they organised?

These considerations help form the view point from which modelling can be initiated.
The approach taken here is exactly the same as would be taken if the analyst followed the standard Role Activity Diagram methodology except a ‘reduced set’ notation as defined in Figure 2.5.1 is used. The assumptions made when applying this approach to the principal aspects of the model are outlined below.

2.7.4 ‘Role’ – System Boundary

The ‘role’ in a business process is enacted by a person or department accountable for a series of tasks that must be performed to ensure the process is completed.

It is assumed that the person or department enacting the role in question has the requisite skills to enable them to perform the task – and therefore this aspect does not impact on the analysis.

The Role holder themselves may or may not perform or complete these tasks (others in their employ may be asked to perform the tasks). The important distinction is that process ‘flow’ will not progress until any task that they are accountable for has received their authority to allow it to progress.

In practice, this ‘mechanistic’ approach has to be tempered by the system role holder’s judgement. For instance, there may be many occasions where a task is not fully complete, but there is enough information to allow progress of the process. In this case the role holder may allow progress ‘at risk’ as long as that risk is defined and the stakeholder(s) that make use of that information are made aware of the deficiency, there is also the possibility of operation about the same task steps if the conditions to progress are not met, this is discussed further in section 2.10.13.

Defining a system boundary for modelling purposes will depend in the case of a Soft System on both internal technical and commercial responsibility (for technical expertise and apportionment of budget and resource) and external technical legal and commercial responsibilities that are shared with the customer and with any partners in the service provision that forms the Servitisation business.
For the type of business model reflected by this research, where a former product supplier decides to offer a leasing service for their product, it is essential that the Role or in terms of this methodology, system boundary is defined, as a lack of organisational response to maintenance alerts has a direct and immediate effect on the profitability of the Servitisation business case.

It is also likely given this scenario that a number of parties, each having their own system boundary, will be involved in the actual response to the alert and that not all of them will be under the direct control of the party leasing the equipment.

Developing a modelled understanding of weaknesses in response reporting is essential therefore not only to improve the response of the party providing the lease in order to improve their own profitability but also in defining contractual boundaries (or as with the physical system - interfaces) with other parties in terms of their responsiveness and what is required of them.

Soft Systems are heavily dependent on establishing a modelling perspective. To model any system effectively reference should be made to the incumbent organisational structure. This issue is complicated if the organisation is a matrix or matrix variant organisation type [54].

By and large though there are a string of events or tasks which make up the organisational response to an alert and these are usually bounded by either departmental (in a hierarchical organisational structure) or management role responsibilities (in a matrix organisational).

The start point and end point of this string of events when transposed on the organisational structure form the system boundary. The interface points within the system boundary will be determined by the organisational structure of the company. If some of the organisational response tasks are carried out by a third party, then there is also a contractual interface with a separate business organisation to be considered.

Organisational structure is always an important consideration not only because it gives an important guide to the way operations are delivered but also because
the change in operations as reflected by the research (moving from product to service delivery) may require business process re-engineering such that either department or managerial responsibilities are amalgamated, partitioned, reduced or extended. In other words it is important to understand what operations are not currently performed by the organisation and what capability therefore needs to be generated in order to be able to respond to the new business need.

The method proposed can be extended to deal with such questions.

2.7.5 ‘Task’ – tasks performed within the system boundary

The tasks are those operations within the system boundary that the role holder must perform in order to fulfil their ‘role’ in the process as a whole. It should not be assumed that the role holder will perform a series of tasks in isolation. Often tasks are carried out concurrently and there may be tasks which are shared between system boundaries.

It should also be noted that the tasks for which the role is responsible are generally those that form their accountability statement.

2.7.6 ‘Interface’ – interactions between the system boundaries

The interfaces between each system boundary can be looked on as deliverables. That is, what one stakeholder has to produce and deliver to another stakeholder before the process is allowed to continue.

This may appear simplistic and indeed the level of abstraction imposed by the analyst is an important aspect in delivering an accurate view of the process and therefore the resulting analysis.

It should also be noted that not all deliverables in the defined interface need be met – process can proceed at risk in this case. It is expected that risk values can be laid against partial delivery, and this may be used as part of the analysis.

It should also be noted that the system interfaces for a Soft System will be defined by the system boundaries. This means that note should be made of
boundaries which develop or change through the modelling process in order that the completeness and correctness of the interfaces are maintained.

Taking such an approach acknowledges the type of organisation that must be reflected by the model whilst not allowing the organisational structure to drive model development. System boundary definitions can be applied equally to both managerial responsibilities and departmental responsibilities (usually listed as job and department descriptions).

In addition, organisations compliant with the international standard for quality management systems ISO 9001:2008 [46] also have a prescribed set of tasks which describe the operation of the department (or system boundary) and the way in which the department should behave and respond in performance of its everyday functions.

These tasks therefore form the basis of the model of the Soft System and all that is required is to link them appropriately in order to understand the full response model.

This linkage is only performed where necessary under ISO 9001:2008 and is best achieved graphically.

It should also be noted that unlike the Hard System interfaces (which are largely defined by component attributes and connections) these interfaces are:

- Not always well defined;
- May require some form of human action to stimulate the connection to enact;
- Are not necessarily maintained continuously (if the action is for example a permission to proceed delivered by agreement from a manager);
- Are temporal in nature.
  - May not always occur in the same time frame over the same time period every time they are enacted;
The nature of a system boundary will change throughout the action of the response, for example, the system boundary may have a number of tasks at the initiation of the response but be progressively less involved during the solution phase and required to be heavily involved again at the end of the response cycle).

2.7.7 Overview of process ‘flow’

There are strict rules imposed on process flow through using the standard Role Activity Diagram methodology.

These rules are imposed primarily because the methodology was formed in a software engineering domain where rigidity of process is essential. The main points are as follows:

1) Process flow moves generally from left to right across the diagram and starts at the role at the extreme left of the diagram;

2) The process flow begins at the top of the role and ‘falls through’ each task in that role until an interface ‘branch’ is reached;

3) Process flows from one task to another task when the previous task is complete;

4) A role interface passes process control to the role which it interfaces with;

5) Process flow will continue in a role until either all tasks are completed or the process is waiting for an input via an interface from another role.

In the formal RAD notation, the state of the process (that is the point at which the process has progressed to) is indicated by a ‘token’. This is not shown on any of the following diagrams but it is an attribute that RADs share with Petri Nets where the net updates to a new state (or a token is passed to the next task in the process chain) only when input conditions for that state are met.

This approach ignores the everyday practicalities of business and therefore process operation as it is very much an idealised view of the world. Effectively it requires that each task has to be completed before the flow of the process can
resume. It does allow for different process routes to be undertaken (analogous to ‘if-then’ loops), but not necessarily for interrupts or iterations of that process. These factors are routine and everyday occurrences which upset the flow of normal business operation.

The approach taken by the proposed modelling notation, does not take account of process interrupts and neither does it allow for ‘if - then’ statements. This can, rightly, be seen as a slackening of the rigidity of the methodology, but by the very fact that these types of decision are not specified, it forces the analyst to think at a higher level of abstraction. The primary aim is to answer the question “what is really required for the process to continue?” This approach, it is felt, will also appeal to the non-specialist who is asked to map a process. Minimal training is required to acquire the full benefit of the notation.

Individual decisions usually specified in the standard RAD methodology are considered to be at too low a level of abstraction to be assessed. Effectively, it is the results of these decisions that are crucial to effective process progress and performance.

In this ‘truncated set’ methodology, it has to be assumed that any particular set of tasks by any particular stakeholder can be iterated at any time. This aspect will need to be accounted for in the analysis methodology by time taken to complete a step or particular set of steps. Repeat analysis of the time taken to carry out certain steps will eventually lead to either an acknowledgement that the time taken to carry out the series of steps is extremely variable by nature (and can therefore have a probability limit imposed on it) or redesign of the process is required.

2.8 Considerations when applying the notation to Hard Systems

2.8.1 Explanation of approach

The most important system attributes to be modelled in a Hard System are:

1) The physical system or subsystem;
2) A component of that system or subsystem;

3) The physical interfaces between components of the system or subsystem.

There are issues that need to be resolved before a representation of the system can be made even with just these few attributes. The main point to be resolved is ‘what constitutes a system and what constitutes a subsystem?’

2.8.2 ‘Role’ - System boundary or platform

The answer to the question of what constitutes a system or subsystem is subjective. The practical answer is that the ‘system boundary’ depends on the analyser’s perspective and as long as that perspective can be justified, there should not be a problem.

In fact, these issues may be quite simple to resolve. It may be that a ‘system’ can have its boundary defined practically by its functionality, its interface type (for example all hydraulic components) by those parts that are provided by a particular supplier.

In order to simplify this issue for demonstration purposes though, we use the concept of a system ‘platform’. It may be that an Aircraft is a platform or an engine is a platform or an engine controller is a platform. In this way levels of abstraction for analysis purposes are differentiated with the higher level of abstraction being the aircraft as a platform, the lower level being a sub system component of an aircraft such as an engine as a platform.

Each platform will consist of its own subsystems and components which will interface with other platform subsystems and, possibly, the outside world.

This will provide a clear view of system to system interface and dependency which enhances the analysis process when compared to trying to analyse a system either based along functional lines or in terms of trying to establish important interfaces from a schematic diagram. In other words the discipline of recognising interface dependency is established early on in the process.
2.8.3 ‘Task’ – System component

Each platform ultimately consists of components. Again there is a distinction to be drawn between certain types of components. For example, a bolt is a component and is as worthy of consideration in the analysis as an electric motor used to power a system.

Again this is a subjective argument as above, and the boundary set around the definition of ‘component’ is largely to be decided and justified by the analyst and will depend largely on the nature of the analysis to be carried out.

As a nominal guide, we can say that a system component can exhibit some or all of the following properties:

- The smallest easily replaceable item in the system;
- Provided by a single supplier, who is wholly responsible for that equipment;
- An item which has had a Failure Mode and Effects Analysis (FMEA) performed on it by its supplier.

These attributes are only provided as guidelines and the final decision as to what constitutes a system component should be made by the analyst, along with the assumptions made in deciding that approach.

It is acknowledged that product supportability issues also have an effect on these decisions. The magnitude of that effect depends upon the Servitisation business case proposed, the overall design and design hazards. If the product to be Servitised is already in service then such issues can be dealt with in the design of the reliability monitoring system. If this is not the case then there is a balance to be struck in terms of design benefits and disadvantages when designing both Hard System components and the monitoring system which monitors their reliability.
2.8.4 ‘Interface’ – Component interfaces

This modelling technique aligns with the HAZOP analysis approach in so much as the interfaces are defined in order to be assessed.

In the notation for physical component system modelling, interface is defined as the connection between identified components in each platform.

Most components will share some kind of common interface with another component in the same platform and some will share a common interface with components in other platforms.

The primary perspective of analysis will be concentrated on interfaces of a particular type, for example, analysis of all components that have a fluid system connection between components on different platforms.

It is noted however, that some components on any particular platform will have interfaces with more than one control mechanism. An example would be that of a solenoid control valve which is operated electrically, but on demand will allow flow of a fluid to pass through the valve. In this case, all perspectives would be modelled individually according to interface type meaning that the component would appear in the model as many times as it has interface type.

2.8.4.1 Interface ‘flow’

When modelling Soft Systems the process interface ‘flow’ is usually sequential from system boundary to system boundary and ends with the final task in the process.

Hard System interface ‘flow’ is different as it depends upon the type of interface and whether or not the interface flows in a closed circuit (such as an oil lubrication system). Some interface types may be open circuit (such as fuel flow from a tank to a burner). Here the interface may start and end on the same platform but be passed to other platforms in the interim. For example a fuel tank (interface ‘source’) and fuel burner (interface ‘sink’) may be on one platform and the components that control the fuel system and fuel system distribution may be on other platforms.
2.8.4.2 Interface ‘source’

An issue which impacts on interface dependency is that of interface ‘source’. This is defined as the source of the interface. In most cases, the interface source will be defined by the action on that source of a component. For example, if a solenoid valve is opened, a ‘source’ of hydraulic fluid is produced for all components connected to that interface.

The interface source must always be defined at the highest level of platform abstraction for open circuit systems. This must be done to ensure that all issues concerning the continuity of interface flow are highlighted and considered. Failure to do so may affect system availability.

2.8.4.3 Interface ‘sink’

In the same way that an interface is ‘sourced’ it must also end appropriately in a ‘sink’. In many cases this will be a component and will not be of consequence, but in some cases it may be that a sink is of interest to the analyst in terms of safe and effective removal of a particular substance such that it will cause no hazard to either the platform or system as a whole. For instance hot air may be exhausted from a component and it would be of interest to the assessor that the hot air did not impinge on another component to that component’s detriment.

2.9 An introduction to the formal description of the methodology

The aim of this methodology is to assess the effectiveness of reliability monitoring systems in their entirety in order to support a Servitisation business case.

The intention is for the assessment to be performed in the early stages of system development in order to test the system concept and provide proposals for any necessary system redesign based on the assessment of effectiveness of that system concept as it relates to the Servitisation business case.
It is not an intention of this methodology to perform a full scale and detailed evaluation of a system as other methodologies and assessment techniques are in existence which can test in detail, specific aspects of the overall system.

For any level of assessment to take place, first the system as a whole must be modelled. The formalised description of that modelling notation, based on the key requirements and principal components for modelling discussed above is listed below.

Once the modelling is complete the evaluation is made. The evaluation suggested is qualitative rather than quantitative in the first instance, primarily because the intention is to inform a view regarding reliability monitoring system capability and weaknesses at the whole system concept stage of design.

In summary the intention of this methodology is to provide:

1) An understanding of both Hard and Soft Systems and where and how they integrate;

2) An appreciation of which system ‘components’ in both Hard and Soft Systems are most vulnerable;

3) Identification of which Hard System components are monitored for control purposes (to avoid duplication of sensors and potentially optimise the use of these sensors);

4) A recognition of which Hard System components are not/ cannot be monitored;

5) A view of the state of the infrastructure and the integration of that infrastructure that is required to provide data for decisions to be made on the ‘health’ of the Hard System;

6) An insight into how the decision making data is to be captured and processed;
7) An awareness of the time required to perform the data review process in order to determine response time for system failure identification. Such an appreciation informs the Servitisation contract negotiation process regarding failure identification and subsequent maintenance action timing.

All of the above aspects are intrinsic to the design compliance to the business case for delivering a failure monitoring service.

If any of the above are not considered, then at best the design will be sub-optimal and as a consequence, may not support the intended business case. If this is the case the business may suffer financially through failure to meet contractual arrangements and loss of goodwill and prestige.

A full understanding of these issues leads to more focused and tailored investment and business decisions with regard to the Servitisation offering being made.

2.10 A formal description of the methodology and how it is applied

This section provides a formal description of the proposed methodology, its notation and how it should be applied.

The formal description is provided by means of the application of the methodology and its supporting notation to simple example Hard and Soft Systems in a stepwise manner.

2.10.1 Hard and Soft System example description

The Hard System to be assessed is a reliability monitoring system which is to be developed in order to monitor the reliability of an engine tank system.

The engine tank is an existing product and is shown in block diagram form in Figure 2.10.1 below. This arrangement forms the Hard System. There is scope to add to the monitoring capability of the Hard System if required to satisfy the business case.
There is at present no defined function for handling the reliability data once it has been collected from the various monitoring sensors in Figure 2.10.1 following a failure event but an example flowchart process like that shown below in Figure 2.10.2 is suggested for this purpose.
As this is purely an example for demonstration, no business case requirements are defined. The object of the analysis process for this example is purely to define the model of both the Hard and Soft Systems and then aid in the identification of the modelled design weak points in reliability terms.

Taking the key requirements listed in section 2.2 as a guide and taking note of the methodology discussion developed in this chapter the following steps are used to formally describe the methodology.

2.10.2 Step 1 – Determine the Hard System model

First, look at the Hard System to be monitored in order to obtain an overview of how the system components relate to each other in context.

This is often not an easy step when considering a complex or multifunctional system and is especially difficult for a system that has either been developed over many design generations in order to extend its product life cycle or has been
developed to retain commonality between successive generations of design but still performs the same function. In such a case modifications to the system are not usually modular additions but are enhancements to functions. It is possible in some circumstances that such modifications can lead to unexpected emergent system properties which affect the operation of the system as a whole. Such properties can be difficult to identify and isolate as specific entities.

The main aim of this step though is to identify major components of the Hard System and establish their relationship in context. Understanding this context and the relationships between the major components is critical for the following reasons:

- Commercial responsibility;
  - Some systems or components may not be covered by the design responsibility of one agent, such an interface needs to be evaluated both technically and commercially.
- Monitoring capability.
  - The monitoring capability may in some cases be a shared interface activity (for example bearing failure could be detected both in an oil stream interface and through a vibration interface).

Understanding context and interfaces in a general sense as an overview helps to ensure that design and commercial boundaries are recognised, defined and exploited optimally.

2.10.3 Step 1a - Establish the system platforms

The first task is to establish the main system 'platforms'.

A 'platform' could be defined in a number of ways. For the purposes of this research it is classified as a boundary which defines the main and clearly identifiable part of the system which either consists of a number of sub-systems or has some action of control or monitoring performed upon it by an ancillary system. If a third party contributes to the system design, the platform boundary
should be considered to be the boundary of commercial and legal responsibility of that third party.

From our example, it is obvious from this definition that one platform will be that of the engine tank as it is the subject of the system as a whole.

The second platform would be the reliability monitoring hardware and software that monitors the hydraulic flow regime through the tank due to its functionality and a third platform would be that of the control hydraulics for the tank as these are of a specific interface 'type', in this case, all hydraulic. Note that these platform definitions can be redefined at any point and the system remodelled accordingly if the definitions are found not to be appropriate. As discussed earlier in the definition of Hard System modelling notation, once a platform is identified, it is represented by a grey box (see Figure 2.10.3).

![Engine Tank](image)

**Figure 2.10.3 Modelling Notation for a Hard System Platform**

### 2.10.4 Step 2 Defining Hard System platform components

Each platform will have particular components assigned to it. Referring again to Figure 2.10.1, there is a choice to be made regarding which components belong to which of the platforms in the given example.

Resolving this issue is primarily a function of the modelling constraints referred to above which may be commercial or technical. There is no definite correct answer as it depends upon the perspective of the modeller as to which view is correct but in this example we can see that there are both hydraulic interfaces to the tank and electrical interfaces to the tank.

The prime platform which all of the interfaces connect to however is the engine tank. In this case the engine tank is both a platform and a component of that platform. This has to be the case as there is a need to show the nature of the interface to the ‘component’ engine tank but also have to recognise that ‘the
reason for being’ of the system as a whole is to monitor the ‘platform’ engine tank.

This may seem to be an odd approach, but it must be recognised that the engine tank itself may also be considered as a sub-system of a larger engine 'system'. It just happens that we have decided to model at this level of abstraction and not at the higher level of abstraction of ‘engine’.

Once this position is set then it follows that the component parts of the other platforms are defined by their particular interface. With the isolation valve being part of the control hydraulics platform and the sensor components which belong to the electrical interface with the reliability monitoring hardware and software platform.

Each such component should be shown as an annotated black box to represent its particular component. Each black box should then be placed inside the platform to which it belongs. This arrangement is shown in Figure 2.10.4.

![Diagram of engine tank and related components](image)

**Figure 2.10.4 Hard System platforms with their respective components**
2.10.5 Step 3 - Linking the components and adding interfaces

The diagram should then be developed to show how the components are linked together in each platform. This is done by connecting the components with a vertical line in the order in which they are connected in terms of ‘flow’ from a ‘source’ to a ‘sink’. This is less problematic for hydraulic systems as they are usually connected in series in the direction of hydraulic fluid flow. Electrical systems may be more problematical as they are more usually connected in parallel (for example redundant systems).

Once this process is complete, there should be an indication of how the platform components interface with each other. The notation used for interfacing between platforms is that of a ‘white box’ connector.

A white box is placed in each of the platforms; the boxes are linked by a line indicating directional flow of the interface from one platform to the other. The white box is then linked by a vertical line into the component to which the flow passes and then either to another component by a vertical line or (in order to either return to the originating platform or to interface with another platform) to another white box in another platform by a horizontal line.

There must always be a pair of white boxes (one to express flow away from the platform and one to express flow into the connecting platform) when any interface is modelled.

In the case of ‘monitoring’ components such as ‘high fluid level switch’ in Figure 2.10.5, an ‘apparent’ flow from the primary platform of ‘engine tank’ to the ‘reliability monitoring hardware and software’ platform is used. Here a monitoring component is shown in ‘parallel’ to the component it is monitoring, connected by two white boxes about itself in its platform connected to two white boxes about the component it is monitoring in its platform, no directional flow arrows are used in such cases. This establishes a contextual relationship between the monitored and monitoring components.
A completed picture of the engine tank system components with their interface is shown below in Figure 2.10.5.

![Figure 2.10.5 Hard System platforms showing interfaces and linked platform components](image)

**Figure 2.10.5 Hard System platforms showing interfaces and linked platform components**

### 2.10.6 Step 4 - Using the model for analysis

This step depends entirely on the type of analysis that is required. Once the Hard System has been modelled, the diagrams can be reformatted to provide any interface view that is required to support the analysis.

The way in which these drawings are constructed provides a clear indication of responsibility in reliability terms (as the components are clearly 'bounded' by the platform in which they are shown). This informs the analyst with regard to the design and potentially the commercial responsibility for reliability of the system (and therefore the risk exposure to system failure to each party in the design).
Once this position is reached, the next step is to develop a view of risk exposure in the design in purely reliability (not safety) terms.

This ‘reliability risk’ view is intended to provide a clear indication of reliability weakness in terms of:

1) Which components are vulnerable to reliability issues;

2) The coverage of the monitoring system;

3) The capability of the monitoring system to detect all failures.

It allows the opportunity for mitigating action to be determined either in terms of strengthening the design (by adding redundancy) or by identifying areas for further monitoring capability development to determine incipient failure mechanisms.

The ‘reliability risk’ view determines the level of exposure of the system to reliability issues graphically and, once consequent mitigation to the exposure is determined, the commercial viability of the system as a whole can be determined.

2.10.7 Step 4a – Examine system for vulnerability to reliability issues.

In this step we make the following assumptions:

1) All system information is available in terms of what components are used and how they are used;

2) The system is modelled completely and correctly in the proposed notation.

Firstly an assessment is required regarding which components are likely to affect reliability if they fail. This can be done by inspection in early stage and conceptual designs, but is most properly done in conjunction with the system FMEA.

If a system FMEA is not available, components that are considered vulnerable in reliability terms by this research are those which are either not monitored or are monitored, but not all of their failure modes can be detected. This definition holds
regardless of whether or not a component that is monitored has the same reliability attributes (in terms of MTTF and MTTR) as one that is not.

This holds only for this research as a basic premise of the research is that the reliability in question is not that of a product, but that of a service. The only way in which reliability of service can be maintained is by a clear understanding of the state of the product in purely reliability terms. This state can only be informed by a system which can be monitored automatically such that components failure can be predicted, diagnosed or classified as ‘abnormal’ (that is the system may function correctly, but behave differently to that which is classified as a ‘normal’ mode of operation) to act as an inducement for further investigation by maintenance personnel.

This step is therefore used to examine the system for exposure to unreliability and maintainability issues. Specifically, this means looking for:

1) Components that are liable to result in loss of service, but which do not have a monitoring system to predict or detect failure;

2) The ‘level’ of monitoring around ‘sensitive’ components. Where ‘level’ refers to whether or not all failure modes can be detected by the current monitoring system and ‘sensitive’ components being either those most likely to be unreliable or those that have a long lead time for replacement – usually mechanical systems;

3) What failure modes of the sensitive components can be accurately predicted;

4) What failure modes of the sensitive components can be accurately detected?

2.10.8 Step 4b – Description of notation used to highlight system reliability risk

If available, use the FMEA to determine the following information.
2.10.8.1 Monitored components

Monitored components are those components which have all aspects of their state of operation monitored by either single or multiple sensors, or their failure can be detected by some form of algorithm derived from sensor inputs.

Further, all of their failure modes should be identifiable such that failure diagnosis is well established and can hence be confirmed by some algorithm or set of defined circumstances that can be derived without human intervention.

Typically components in this category are:

- Monitoring Sensors;
- Mechanical bearings (failure analysis techniques for bearings are well established).

Represent these components as ‘Green box’ components.

2.10.8.2 Non-Monitored components

Non-Monitored components are those components which are not monitored by any system other than by a routine maintenance programme. Their reliability of operation relies entirely upon the efficacy of the maintenance programme.

This state should also be considered as the default for any component for which knowledge of any monitoring regime is unknown or for which failure detection techniques have not been established.

Typically components in this category are:

- Sensors which are monitored manually;
- Components which are manually operated;
- Components whose failure modes are not known.

Represent these components as ‘Red box’ components.
2.10.8.3 Components that do not fall into either the monitoring or the non-monitoring categories

For any component which does not fall into the definition provided by either of the above categories, a third category is defined which account for all such components.

This category is seen as one where more definition needs to be provided on the component failure modes or the detection capability is weak for this particular component (for whatever reason either technical or commercial).

This category is one which, should a component be placed within it, the best efforts should be made to resolve the status of the component to drive it toward a 'monitored' component state. It is recognised that commercial or technical constraints may prevent this aim from being achieved, but a mitigation case should be provided which details why such a situation is deemed acceptable.

Represent these components as 'Amber box' components.

Performing these tasks provides:

1) An indication of which system components are most at risk;
   a) This is an enhancement to the 'safety' view as it provides an idea of system exposure to reliability risk (which components can and cannot be monitored).

2) An appreciation of which sensors are required to perform the fault detection function;
   a) This substantiates the added importance of some sensors whose primary purpose may be to 'control' the system. Note it is recognised that software plays a part in this detection function, but only as far as interpreting and comparing actual readings from the sensors. Model based fault detection alone is considered not sufficiently robust technique to provide actual fault detection analysis without some physical evidence of either potential or actual failure.
3) A view of which components are and are not monitored by fault detection routines.

   a) This perspective forces a design review purely for reliability purposes to establish if the design approach is sustainable. This is only part of the question however as specific mitigation strategies could be adopted using both commercial and engineering approaches by a maintenance process triggered by fault prediction/detection.

Expanding on point 1)

Failure modes detectable: This needs to be further refined (after an initial review above) to develop an idea of system risk. A ratio of failure modes compared with those modes that are detectable (per component – transferred to system risk) is a potential solution.

Expanding on point 2)

Feedback to the design process regarding the importance of sensors for reliability assessment:

   • Are the sensors accurate enough to detect failure?
   • Is the design redundancy appropriate?
   • Is there an additional effect to the FMECA/safety analysis?
   • Is the monitoring capability appropriate?

Expanding on point 3)

Development of the mitigation process:

   • Mitigation is required against defined failure conditions detectable by designing appropriate ‘on detect fail’ and ‘on predict fail’ maintenance processes;
   • Development of process FMECA for components where failure cannot be detected is required. This will involve specific checks and/or a regular maintenance procedure to be put in place;
• In turn the mitigation processes will need to be analysed to determine the ‘failure rates’ of the process tasks and the MTTR for each component where failure can be detected.

The diagrams below show how the steps described above are put into practice. Figure 2.10.6 below is the same as Figure 2.10.5 but is reproduced here to demonstrate the changes that are made to the model to determine the Hard System Reliability Risk View which is shown in Figure 2.10.7 below.
Figure 2.10.6 The Hard System to be analysed

Figure 2.10.7 The Hard System Reliability Risk View
The Hard System Reliability Risk View (Figure 2.10.7) provides the following:

1) An indication of which components are a reliability risk and are either not or only partially monitored (red and amber boxes respectively in Figure 2.10.7);

2) An indication of which components are monitored and which sensors are required to perform the reliability monitoring (green boxes in Figure 2.10.7);

3) An indication of flow source and sink. In Figure 2.10.7 there is no defined flow source, this is an issue that will need to be resolved at some later point.

The only component highlighted red is the isolation valve. This is marked as such as there is no monitoring function that is applied to this component and the valve itself is manually operated. Only a regular scheduled maintenance action can ensure the reliability of this component.

The components highlighted green are all sensors and are therefore part of the monitoring function. Sensors are marked green both because of this and because they themselves are more often than not subject to some kind of monitoring themselves in terms of ensuring they are functioning correctly.

The component highlighted amber is the engine tank. This is a precautionary assessment position and may change according to advice from a FMEA or from the business case, or from a change in design to accommodate unmonitored failure modes.

The interface lines indicate flow source and sink and also show a contextual relationship between monitored and monitoring components. This provides a clear view regarding system boundary (the flow source will need to be identified) and which components are responsible for monitoring specific components.

It is noted that the monitoring of the engine tank is quite advanced, but pressure monitoring could be added if necessary or leakage monitoring could be considered as an addition. If all identified failure modes of the engine tank are
mitigated by a monitoring scheme, then the engine tank should be reassessed and highlighted as a green component.

It should be noted that the Hard System Reliability Risk View is different from that of a purely control and functional view of the design. For instance sensors that are purely used for control and not monitoring (not the case in this example) would be marked green only if they have some kind of self monitoring. This effect may be the cause of some confusion, but in practice it is more likely than not that all sensors will be capable of some self monitoring function.

The purpose of the Hard Systems Reliability Risk View is to form a basis on which a design review can be carried out. Ideally it is developed using safety analysis documentation, but this is not strictly necessary as indicated in section 2.10.7. Such a view provides an opportunity to show graphically information provided by the FMEA on potential failure detection methods or components that are vulnerable to reliability issues. This is a distinct benefit as the information in the FMEA is shown clearly and in a context which is easily understood.

This view is used to determine:

1) Is it possible to improve fault detection of those components that are a reliability risk but are either not monitored or partially monitored?

2) Are the sensors that are being used for fault detection appropriate in terms of sensitivity and robust enough for this extra duty (is redundancy required)?

3) What is the consequent effect on the FMEA if a system redesign takes place based on the above factors?

2.10.9 Step 4c - Determine System mitigation

Following the design review, where some of the risks listed above may be resolved through more robust design strategies there must be a mitigation phase, which will take the form of a developed and analysed maintenance response to a predicted or detected failure. In other words, a development of a Soft System
which supports the Hard System and which together with the Hard System forms the reliability monitoring system as a whole.

Note that this Soft System mitigation is acceptable as it is an expected consequence of a fault detection monitoring strategy to ensure an optimised availability from the system.

It is a recognised aspect of this technique that this Soft System mitigating response process itself has to be modelled and analysed for reliability (see step 5 onwards) as it is a facet of the overall ‘reliability’ of the service.

Depending on the system and the nature of the detection mechanisms, some or all of the following processes have to be developed:

- Required action on predicted failure;
- Required action on diagnosed failure;
- Required action on potential failure – a monitoring system has detected ‘abnormality’ of operation which could signify an impending failure.

Note that routine or scheduled maintenance is not considered here as any failures identified in this manner are identified in step 4b as exposure threats and therefore mitigated against in the maintenance schedule plan.

2.10.10 Step 5 – Model the Soft System mitigation process

The following steps describe the approach taken by the methodology in producing a Soft System mitigation strategy for those components that cause risk to the reliability of the service being offered in that a process is developed to cope with the maintenance action for a fault condition.

The proposed modelling notation can not only describe and contextualise Hard System components but also allow for a similar approach to be taken to perform an analysis on the process governing any resulting maintenance action that is taken on failure, predicted failure or suspected failure of a particular component.
The Soft System mitigation strategy is initiated by the reliability monitoring system providing an indication of failure, impending failure or suspected failure rather than a routine maintenance action.

The aim here is to provide the reliability analyst with a view of the reliability monitoring system ‘capability’ rather than a view of just the Hard System and its components. This ‘capability’ viewpoint (which includes the Soft System process modelled in Figure 2.10.12) is an indicator of how supportable ‘Servitised’ products are using the reliability monitoring system.

2.10.11 Step 5a - Establish the Soft System boundary

The first task is to establish the primary system boundaries in the maintenance response to a detected failure. This step is essentially the same as that of determining the main platforms of the Hard System. Soft System boundaries could refer to either a person or a team or departmental responsibility. Interactions between these boundaries could for instance be either passing of information, data or an approval statement. These interactions and the task responsibilities of these roles in the Soft System determine system boundaries in a similar manner to components and component interfaces in the Hard System.

This is best done by deciding which are the principal departments (or specific individuals) in a company (or companies) that will be responsible for fulfilling the proposed maintenance action in response to either a detected failure, a predicted failure or a suspected failure.

In the main here we are attempting to show those departments (or individuals) that are accountable for the various tasks within their sphere of influence in the process as a whole. For example one of these main system boundaries might include authority to remove equipment for repair.

Using the information provided by the flowchart in Figure 2.10.2 as an example, as with the Hard System modelling approach which applies the same technique, each main system boundary shown in Figure 2.10.2 is represented by a grey box (see Figure 2.10.8).
2.10.12 Step 5b Defining tasks of the Soft Systems boundaries

Once the Soft System boundaries have been established, a detailed view of the actual process is required.

The points of interest are the nature, type and function of the tasks carried out within the identified system boundaries and the nature, type and function of the interfaces between each system boundary.

By describing these interfaces and with knowledge of the tasks that are performed within each system boundary, an analysis can be performed on the Soft System or parts of the Soft System as detailed.

In practice, this stage would also require negotiation and agreement between those involved in delivering the Soft System response as there may be contractual implications raised by the final representation of process responsibilities. These aspects are not dealt with explicitly here but the Soft System process modelling will help to highlight such issues.

For the process to execute; specific tasks in each system boundary must be performed; these tasks define the system boundary responsibilities. These tasks are usually defined in any company as part of the documentation set that is used to confirm adherence to ISO 9001:2008 standards. Note, such documentation usually requires that a particular task is performed, but does not necessarily prescribe how that task should be performed. It should also be noted that the processes described in this documentation are not always integrated across
departmental boundaries, but do provide useful information as to how each
department executes its responsibilities.

The format used to define these activities could be in the form of a flowchart, a
Gantt chart or by some other process modelling technique as discussed earlier.

Agreed tasks relating to each system boundary should be represented as black
boxes and placed under the appropriate system boundary in the diagram in the
same order in which that are carried out according to the ISO 9001:2008
documentation (see Figure 2.10.9). Each black box can be considered to be a
placeholder for information regarding the attributes of the task. This could
include estimated time to complete the task (in person hours) and the variance of
actual time to complete over a number of iterations of the task. Such information
could be used in the analysis to provide an indication of the time taken to carry
out a maintenance action and also the repeatability of that maintenance action
time over a number of instances. Other information for example, cost and risk
levels could also be added if necessary but the prime here is to develop an
understanding of the time taken to understand the consequences of a system
failure. Such information is not included in this example.
Figure 2.10.9 Showing the tasks that need to be carried out apportioned amongst the system boundaries that have responsibility for those tasks

As can be seen from the example, the tasks listed in the flowchart in Figure 2.10.2 are included here except for a decision box which is not included as such in Figure 2.10.9; instead this action is described as a task in line with the modelling notation developed for this technique.

2.10.13 Step 5c Linking the tasks and adding the appropriate interfaces

Each task in the process should then be linked in series in the role to which it belongs and in the order sequence in which the tasks should be performed.

Tasks are completed sequentially; the top most task in any Soft System must be completed before the task below it can be started. This task sequencing is represented by the vertical line which connects the task boxes (See Figure 2.10.10 which shows this linkage).
Figure 2.10.10 Showing linkages of tasks, indicating task sequence

Any system boundary can complete its series of tasks independently from any other system boundary but if the sequence of tasks that one system boundary relies on, either for information to be received or for a sequence of tasks to be completed before it can continue with its sequence of tasks, that interruption is represented by an interface notation indicating that interface. Note that the process halts at that point in the system boundary until the information requested is received, but will not cause a halt in the process of the ‘sending’ system boundary which will be able to continue its own sequence of tasks.

The interface notation consists of a white box, placed beneath the task which produces the information in the ‘sender’ system boundary via a straight line to connect to a white box in the ‘receiver’ system boundary (see Figure 2.10.11). An arrow or explanatory note can describe the nature of the interface and indicate which system boundary is the sender and which the receiver.

Again these white boxes will have similar attributes to the black boxes in that they will have a time to complete associated with them and a ‘repeatability’ time associated with that action. This time will not be associated with either system
boundary as it is a shared attribute, but it must be accounted for (especially if it crosses an organisational boundary and is therefore constrained by contractual arrangements).

It is suggested that these interfaces are the main sources of failure in any process and are therefore very important attributes.

**Figure 2.10.11 Showing sequencing of tasks and interfaces between roles**

Communication within system boundaries is assumed to be instantaneous when passing information between tasks unless otherwise stated. This may or may not be the case in point of fact, but if communication between tasks is not instantaneous, the time associated with the communication effort will be assumed to be an intrinsic part of one, other or both of the communicating tasks time. Communication between system boundaries should always have a time associated with it, but this is not shown in this example.

It is recognised that this approach differs from the more formal flowchart method in that for example, ‘if – then’ decisions are not shown and therefore there is a loss of the opportunity to ‘skip’ to other points in the process on particular
conditions. This is tolerated as the flexibility or lack of formality of the approach means that a number of process sources can be used to populate the process model. The above example uses a flowchart, but the source information could easily be derived from a Gantt chart where tasks would be the analogous to the activity bars of a Gantt chart and major deliverables would in effect be interfaces between system boundaries, assuming they cross a system boundary.

This flexibility and lack of formality can lead to a less strict definition of the tasks that need to be completed, those that are responsible for those tasks and an examination of the sequence in which those tasks must be performed in the wider organisation. As with Hard System concept definition however, an early appreciation of the design concept of the process is more important in a reliability sense as it allows an early evaluation to be made of the system as a whole before design decisions are made formal and detailed. Such an early evaluation has the potential to save expensive later project stage redesigns.

It is also recognised that human factors will play a part in the delivery of the Soft System mitigation process, as will competency within the system boundaries along with other factors that are not easily modelled such as morale and organisational culture. It is acknowledged that these factors will have an effect on the Soft System as a whole but although their effect will not be explicitly captured, measurement of the overall system response is the important point for the reliability analysis. The above mentioned factors will manifest themselves as variance around this overall system response.

This is acceptable as the main purpose of this notation and analysis is:

- To determine a clear picture of the process needed to deliver the maintenance action response described by the modelled process in the same way as an analyst would seek to determine a clear picture of the design of a physical system for analysis;
- To provide assurance of the ‘fitness for purpose’ of the system design and process design respectively;
• To provide a basis to measure and monitor variations in performance according to human factors.
  
  o These will be accounted for once a number of instances of the process have been completed which will allow them to be associated with repeatability and variance figures.

2.10.14 Step 6 – Analyse the modelled Soft System mitigation process

The main differences between this reliability analysis and the analysis defined by the Hard System Reliability Risk View above, are in general that the main points of vulnerability are likely to lie in the interfaces rather than the ‘tasks’ themselves.

Whilst Hard System interfaces are acknowledged to be points of vulnerability, these are more likely to be ‘vulnerable’ in the design phase than the actual operation phase of the system (loose definition of design interfaces between system ‘platforms’ is often a cause of system failure during the development phase). It is also more difficult to determine interface failure through monitoring systems as these systems generally concentrate on monitoring either particular component or subsystem behaviour attributes.

In Soft Systems terms, communication within system boundaries is assumed to be perfect. Often the system boundary is defined by a named person, or is a responsibility that is usually shared by a co-located team.

Passing information between people and between teams however is problematic and so is the method of communication (for example, phone calls may not be taken, data transfer may not be smoothly accomplished, or the required data may not be sent, or is sent in an incorrect or incomplete format).

Each of the Soft System modelled tasks and interfaces will have attributes assigned to them, for analysis purposes therefore the following analysis terms are of interest:
1) An appreciation of the time it will take to respond to any fault identified by the automatic monitoring system;

2) An appreciation of how accurate and repeatable the time to respond to any fault identified by the reliability monitoring system.

The attributes that are of particular interest therefore are:

- The time taken to perform a particular task;
- The time taken to transfer and receive information between roles in the process;
- How repeatable these times are over several instances.

Once these times are determined (and this information should be freely available in a well managed organisation through time and motion studies, or through normal planning activity) the task and interface times should be added together to provide the time to respond to the fault.

It is also important to remember that the Soft System model will differ from the Hard System model in that there are potentially a variety of instances of the use of that model depending upon the input to the model which in turn is reliant upon the capability of the reliability monitoring system. A Hard System is designed to perform a specific function, but a Soft System is flexible and can change its mode of operation to perform different tasks depending on what it is required to do.

For instance if only indicative failures can be detected then a simple response process will be required where (as in the Gantt chart example shown in Figure 2.7.1) a brief confirmation of the alert and the data supporting the alert is required before ordering a maintenance action.

If however a predictive or suspected failure is capable of being detected then a more involved process may be required before a maintenance action is scheduled. Such a process may involve a series of checks and balances to ensure that a system is not made available for repair unless absolutely necessary.
as to do otherwise could have detrimental commercial consequences for both the Servitised system provider and customer.

Such situations, where there is doubt surrounding the propagation of a failure mode may also require explicit consent from the operator in terms of how they wish to provide their service before further action is taken.

Each type of alert will therefore require a different mitigation response process model to be developed.

Hence, it is recognised that there may be more than one ‘time to respond to a fault’ as there are three possible scenarios provided by reliability monitoring systems:

1) Time to respond to a detected and diagnosed fault;

2) Time to respond to a detected, predicted fault;

3) Time to respond to an unrecognised fault condition.

Expanding on point 1) this time relates to a fault condition or failure mode that is recognised and detected.

Expanding on point 2) this time relates to a fault condition which predicts failure. That is precursors to failure have been recognised and detected by the system. There is therefore another time associated with this response time and that is the likely time to failure of the particular failure mode that is predicted bounded by specified future working conditions of the equipment.

Expanding on point 3) this time to respond relates to a potential new failure prediction condition, which has not been detected previously during system development. In effect, this time will involve the inclusion of a specialist data review of the particular abnormality that has been detected. Although this time to respond is not the main driver of the analysis, it is an important component of mitigation, to provide some form of insurance for the service provider for outlier conditions.
Although it is valid that a number of these responses will exist, it should be an aim to rationalise all process charts that are so developed. If there are opportunities to develop areas of commonality between these processes, these should be acknowledged and implemented. Such a simplified approach is more likely to improve reliability of the system as a whole and the modelling method chosen will facilitate such an activity.

As an aside, although not part of either the mitigation process or this reliability assessment technique, it should be remembered that providing a service which monitors systems in order to provide prognostic or diagnostic information has uses beyond service provision. For example, monitoring data from a system either under test or in actual operation has quantifiable benefits to the organisation providing the service in more ways than purely that of providing an operating capability. Data of this sort is invaluable for future design improvement, spares provision and managing the monitored fleet in terms of modification programmes and scheduling maintenance. All of which adds to the business case for the Servitisation system provider enabling them to optimise profit and customer satisfaction through an improved level of service from the service contract applied to the system.

2.10.15 Step 6a – description of notation used to highlight system reliability risk

A means of showing the actual Soft System reliability risk exposure is required and this, as previously with Hard Systems, is best shown graphically. Although such a view does not improve or add to the analysis provided in the above steps for Soft Systems, it does enhance that analysis and provides the system designer with cue points as to where either improvement in the process is required or whether organisational or infrastructure improvements are required to speed up the response to an operational failure.

As with the Hard System, the diagram which describes the system is used as a basis and interactions in process systems for our purposes are described as either ‘automatic’ or ‘manual’ where:
- **Automatic** Means that an interaction is made without human intervention, for example, transfer of data is made immediately, completely and without human intervention;

- **Manual** Means that an interaction is made either solely by human intervention or when some form of human intervention is required for the interaction to take place (for example, a repair or maintenance action needs to be signed off before the equipment can re-enter service).

This is done as it is assumed that the interfaces between system boundaries are the most unreliable aspects of the Soft System. This assumption is made as the analysis outlined in section 2.10.14 which deals with the monitoring of task times within system boundaries addresses internal boundary unreliability. It is acknowledged that timing is not the only issue that affects task unreliability, but in terms of the aims of this research it provides sufficient guidance as to the causes of unreliability which can be addressed following analysis.

It is also often the case that manual paths of communication are informal and may (unless properly proscribed) arrive in different or unexpected contexts to those which move the process forward efficiently.

Automatic systems are assumed to be predictable and regulated and are therefore considered to be less of a hazard in terms of process, but still may cause issues in failure of infrastructure to deliver the process (for example computer network loss). It should also be noted when developing Soft System Reliability Risk Views that automatic interactions are likely to already exist as part of a legacy system and are therefore of less risk than manual systems. It is acknowledged however that this may not always be the case when considering early concept phases of development.

Both of these types of interface therefore require particular attention and in themselves require some form of mitigation activity, which may take the form of either a redundant back up computer system for automatic interfaces or process improvement/proscribed definition of role interface for manual interfaces.
The Soft System Reliability Risk View is therefore developed from the model in Figure 2.10.11 and is shown below.

**Figure 2.10.12 Soft System Reliability Risk View**

In Figure 2.10.12 above the interface to the specialist is assumed to be automatic as the failure assessment assumes comparison with all known modes. The interface boxes for automatic interfaces are coloured green. Whereas the specialist reply is likely to take some time and may require extra data from other sources. This is therefore a manual interface. The interface boxes for manual interfaces are coloured red.

Such details may require either development of the process chart in order to better understand the process of the specialist’s actions or requirements or better definition of the assessment task but are not discussed further here as the intention is only to show the methodology proposed.

The resulting Soft System Reliability Risk View is a qualitative appreciation of Soft System vulnerability.
2.10.16 Step 7 – Determine resultant risk exposure to failure of the system following analysis

This step is used to combine analysis aspects from steps 4 (Determine Hard System mitigation) and 6 (Analyse the modelled Soft System mitigation process).

Some aspects of the analysis carried out in step 4 may remain without mitigation and must therefore remain as commercial risks for the service provision (components that cannot be monitored, or cannot have all of their failure modes detected). These must be accounted for in any integrated risk assessment for service provision.

The analysis carried out in step 6 will also provide valuable commercial information both in terms of carrying out the service and in terms of developing metrics for carrying out the service provision in terms of providing an estimate of cost and organisational resource involved.

There may be an additional benefit in performing the reliability assessment of the Soft System mitigation processes as other business benefits may be identified through the analysis (reduction of effort duplication or the identification of leverage in other parts of the business for data collected or in service performance monitoring of design and operator behaviour).

As has been previously stated, it is unlikely that a reliability figure for the holistic system can be obtained or rather if it is obtained, be meaningful as the difference between Hard System and Soft System reliability figures potentially vary by orders of magnitude. Even so, a view of the holistic system provides important and quantifiable insights to the acceptability in business terms of such a system concept. Given the precept of a business moving to Servitisation from a purely product manufacturing and servicing background, the business benefits and pitfalls made apparent by such an analysis are a key issue in the development of a business case for product Servitisation.
2.11 Summary of the methodology proposed to model and analyse simple systems

The process defined and developed in this chapter is a long one, so a summary showing the main analysis stages and which process steps belong to each stage is listed below:

1. Determine the Hard System model to be analysed.
   a. Establish the Hard System ‘platforms’
      i. Which are the main ‘platforms’ of the system as a whole?

2. Define the Hard System platform components.

3. Link Hard System platform components within the platform and where appropriate define interfaces between platform components.

4. Use the resultant Hard System model for analysis.
   a. Examine the Hard System for vulnerability to reliability issues;
   b. Use notation to highlight Hard System Reliability Risk View;
   c. Develop System mitigation.
      i. The responsibilities and interfaces between each platform should be clearly defined in this model.

5. Model the Soft System mitigation.
   a. Establish the Soft System boundary;
   b. Define the tasks of the Soft System boundary;
   c. Link the tasks and add the appropriate interfaces between boundaries.

6. Analyse the modelled Soft System mitigation.
   a. Add notation to Develop the Soft System Reliability Risk View

7. Determine resultant risk exposure to failure of the system as a whole following analysis.
2.12 Chapter Summary

This chapter attempts to develop and formally define the methodology of the proposed assessment technique which is used to assess reliability monitoring systems which support a Servitisation business case in terms of engineering capability and effort.

The key requirements for such a methodology are defined and discussed. The key aspects required of a modelling notation to support the assessment are also defined and discussed.

Finally, an example Hard and Soft System is provided and the assessment methodology is applied to it such that the methodology can be demonstrated and also described in terms of how it should be applied. This demonstration provides the formal description of the rules of application of the methodology in a step wise manner. These rules of application are also provided as a summary.
Chapter 3 - Application of method to a simple system

3.1 Introduction

In order to fully demonstrate the methodology it must be applied first to a simple example. This chapter shows how the methodology is applied first to a simple Hard System and then to a simple Soft System.

In each case the systems are modelled using the notation described in the previous chapter and are then assessed using the techniques described in the previous chapter.

Assumptions are made regarding both systems; these assumptions are stated where necessary.

3.2 Simple Hard System example description

A water tank system is illustrated in Figure 3.2.1, this example system and its description is sourced from Hurdle et al [47]. The system aims to maintain the level of water between two pre-determined limits. In normal operational mode water flows out of the system through valve V2 which is opened and closed manually to provide water when needed.

V1 is an air-to-open (A/O) inlet valve controlled by C1. The level sensor S1 detects the height of the water in the tank. In normal operating mode if the water in the tank falls below the required level (as indicated by S1), the controller C1 would open valve V1 allowing water into the tank. Conversely, if the water in the tank rises to the required level then C1 will close V1.

Valve V3 is an air-to-close valve (A/C) that operates as a safety valve controlled by C2. This will only become operational when a failure occurs that causes a very high level of water in the tank. A signal from S2 would cause the controller C2 to open valve V3 to reduce the level of water in the tank. The valve then closes when the level returns to within acceptable bounds.
The overspill tray, located underneath the tank collects any spillages that may occur due to a failure in the system. So, water in the overspill tray will occur if the tank has ruptured, is leaking, or if the water level overflows from the top of the tank. There are six sections of pipes identified by the labels P1 to P6.

The system has two operating modes; these being ‘ACTIVE’ when the operator opens valve V2, or ‘DORMANT’ when V2 is closed.
Figure 3.2.1 A simple tank system
3.2.1 Assumptions regarding the overall system boundary of the example

In this simple case, the system boundary is determined by the system diagram and description as there are no other factors to take into account. In this case the assumption is that the source of fluid flow (main supply) in to valve V1 is uninterruptible and that flow from V2 (the outlet valve) passes freely to its intended drain.

There is also reference to a pneumatic system used as a servo mechanism to control the opening and closing of the hydraulic control valves. In our case, we assume that the supply to this pneumatic system is uninterruptible.

In practice, these considerations may need to be taken into account with regard to legal and technical constraints and their potential effect on the system within our currently proposed boundary. An extension of the model would need to be developed in order to understand the consequences of these additional constraints.

3.2.2 Assumptions regarding the control hierarchy of the example

In this particular case, it can be seen that the control valves do not have autonomy (as would be the case in a distributed system) and therefore the system can be said to be of a radial type (with a central controller, controlling individual system elements) although this is not explicitly stated.

This radial type of control hierarchy lends itself to this method of analysis.

3.3 Modelling the example Hard System

There are some key requirements that have to be addressed in developing the Hard System model, these are discussed in the following sections and modelled using the notation described in the previous chapter.
### 3.3.1 Determining the main Hard System boundaries or platforms

The first step to take is to define the System platforms. Firstly the assumptions of the overall system boundary are considered and taken into account.

From the system diagram and description it can be seen that the ‘passive’ objects of control and monitoring are the tank and the spill tray.

Therefore these are the components of the first platform which shall be called ‘tank assembly’. Fluid system control for the tank is provided by a set of hydraulic control valves, which in turn are controlled and monitored by an electrical system.

The hydraulic control system interfaces with both the electrical system and the tank assembly. Therefore components that have both a hydraulic interface with the tank assembly and an electrical interface with the electrical system will be those bounded by the second platform which shall be called ‘tank hydraulic controls’.

The electrical system is the other main interface, but its components serve different system functions. It is useful to differentiate the purpose of these functions when identifying components that are either vulnerable in reliability terms or have a multiple purpose to their role. This differentiation is an acknowledgement of functionality, but the way the diagram is constructed does not necessarily reflect an understanding of the technical function that is carried out by the component.

Looking at the diagram of the simple system, we can see that there is an obvious control aspect to the electrical system with components that drive the hydraulic system components. There is also an obvious monitoring function, with sensors used to monitor the tank assembly and support control functions but there is also a reliability monitoring function of the tank system (effectively this monitors the operation of the hydraulic control system) which is provided by the flow meters. From the system description, these flow meters do not form part of the control strategy or the monitoring strategy for the control regime; they therefore exist...
purely to support reliability monitoring and consequently have a separate interface.

The electrical system in this particular case therefore has three separate and distinct interfaces and hence three distinct system platforms:

- As a controller with the ‘tank hydraulic controls’ interface;
- As a monitoring function with the ‘tank assembly’ interface;
- As a monitor for reliability with the ‘tank hydraulic controls’ interface.

Each of these interface definitions partition the system into system ‘platforms’, which are defined by their interface type and are represented in a diagram as grey boxes (see Figure 3.3.1).

It should be remembered that these system boundaries that provide the platform definitions may be altered once the model is complete. Such a change may be necessary if:

- Assumptions made regarding a function were incorrect;
- A function of the modelling that has taken place suggests a re-evaluation.

An example of the latter case would be a re-definition of the use of a sensor from a purely control function to that of a control and monitoring function.
Figure 3.3.1 The principle system platforms of the tank system
3.3.2 Allocating components to the system platforms

Once each platform has been defined, the components that belong to that platform must be allocated to it.

The primary platform is that of ‘tank assembly’, this contains the ‘passive’ controlled and monitored components, these components interface with each of the other platforms.

All components that are connected hydraulically to either the tank, spill tray or each other fall within another system platform called ‘tank hydraulic controls’.

The other control and monitoring system platforms are those of electrical systems, they are:

- A monitoring for control function of the main tank and of the integrity of the main tank (via the spill tray);
- A control function for the ‘tank hydraulic controls’ platform;
- A reliability monitoring function of the tank system (which in effect monitors the hydraulic control system).

The electrical components that will be allocated to these platforms belong to the overall electrical system. Performing the allocation activity differentiates these components and provides a clearer understanding of the purpose of each system platform and its components. This enables:

- A wider understanding of the overall functionality of the component;
- A differentiation of those components that are of direct importance to the monitoring function.

In the example case the following components are assigned to each system boundary:

Tank Assembly
Main Tank
Spill tray
Tank Hydraulic Controls
Valve V1
Valve V2
Valve V3
Flow meter VF1
Flow meter VF2
Flow meter VF3

Tank System Electrical Control
Controller C1
Controller C2

Tank System Electrical Control Monitoring
Sensor S1
Sensor S2
Sensor SP1

Tank system Electrical Reliability Monitoring
Flow meter VF1
Flow meter VF2
Flow meter VF3

It should be noted here that no specific electrical control or monitoring function is mentioned in the tank system description. It is an assumption from the paper [47] that defines the system that data is retrieved from the flow meter for analysis to take place. This implies electrical signal processing is part of the flow meter component package, and therefore exists as a component in both the ‘tank hydraulic controls’ and the ‘tank system electrical reliability monitoring’ interface partitions.
The components identified in this section are added to the diagram as annotated black boxes and are shown in whatever platform they are assigned to (see Figure 3.3.2).
Figure 3.3.2 Assigning components to platforms created by the interface definition
3.3.3 Arranging and connecting system platforms and their components to produce the Hard System model

Once the components are allocated to their respective system platforms, the interface relationship between each component must be established.

For fluid systems (air, water, hydraulic) and mechanical systems, this is a case of defining the route of flow through the components that make up the system.

This is accomplished by starting at the top of the diagram with the source or drive input for each specific interface being 'fed' to the first component. Referring to the design will provide information on which component is next in line for the system flow as it is the component which is connected to the component which first receives the source flow. Using the design to follow the connections of the component allows the modeller to build the picture of component connections through to the point where either the flow goes to 'sink' or the interface is passed to another system platform. The connections which show the flow through the platform are vertical lines which connect the black boxes together from the top of the diagram to the bottom or to the 'sink' (marked ‘???’ in Figure 3.3.3).

The ‘sink’ is marked as ‘???’ as no reference is made in the system description regarding the outflow destination or how the system is supposed to deal with outflow. This is an important point as the modelling technique identifies a design ‘unknown’ which needs to be further investigated and evaluated.

When the flow passes between system platforms in this way, the white box notation form is introduced to indicate the interface and direction of flow.

The ‘white box’ shows a connection point between components in each system platform. For example in Figure 3.3.3 where ‘flow’ transfers from the components ‘valve VF1’ within the ‘tank hydraulic controls’ platform to the main tank component in the ‘tank assembly’ platform it is represented by the white box notation. This connection in reality is likely to take the physical form of hydraulic pipe (although it is not necessary to specify this). Direction of flow is indicated by
an arrow head on the interface line which connects the two white boxes. In the example given flow is in the direction to the main tank flowing from valve VF1.

Flow then returns from the tank assembly platform to valve V2 as shown by a white box below the main tank connected via an arrow indicating directional flow to a white box above valve V2 in the ‘tank hydraulic controls’ system boundary as the ‘flow’ continues through components in that system platform.

There are other uses of the white box notation in Figure 3.3.3. The use is that of flow ‘source’ and ‘sink’. There is no indication of either the ‘source’ of the flow into the example system or the ‘sink’ or destination of that flow from the system from the description given of the system. This may be of no consequence, but it should be noted in case further investigation is required.

This is done through a single white box which is not assigned to any platform but which either acts as a point at which flow enters the system or leaves the system. In the example in Figure 3.3.3, the ‘source’ white box feeds Valve V1 in the tank hydraulic controls platform’ and is annotated as ‘From Mains’ (as an example). There are two ‘sink’ white boxes, one from valve VF2 and one from valve VF3, these are both marked as ‘???’ in order to indicate further investigation is needed.

It should also be noted that components that share an interface on the same platform are joined by a vertical line. This can be seen in Figure 3.3.3. Here in the Tank Assembly platform the Main Tank is linked to the Spill tray as flow may ‘source’ from the ‘Main Tank’ and ‘sink’ into the ‘Spill Tray’. The white box outlets from ‘Main Tank’ to ‘V2’ and ‘V3’ are connected by this line, but this merely indicates that they are in the same flow direction from the ‘Main Tank’ component as the ‘Spill Tray’.
Figure 3.3.3 Showing fluid system flow transfers between system platforms
It is recognised that such an approach is tenable for ‘fluid’ systems such as hydraulic or pneumatic systems, but a different approach is required for electrical systems. This is not because electrical systems behave in a different way to fluid systems as the principle that they are analogous is established, but is rather because the operation of the electrical components in such a system is not in providing ‘power’ but performing a control and monitoring function.

For this type of control and monitoring function and in this type of application (which is not intended as a functional representation of the system) it is felt more appropriate that where possible such monitoring components should be shown in ‘parallel’ (as opposed to the ‘serial’ fluid system) interfaces to the components that they are intended to control or monitor in order to graphically demonstrate their relationship and importance in context to the system as a whole.

This parallel relationship is demonstrated by using a similar white box ‘connection point’ notation as described above. Here though, white boxes are placed above and below both the component being controlled or monitored in one system platform and above and below the component performing the control and monitoring function in a separate system platform and are linked to each other by a horizontal line with the top most white box in one system platform linking to the top most box in the other system platform and the bottom most boxes linking similarly. This arrangement is demonstrated by valve V1 ‘tank hydraulic controls’ platform which is controlled by controller C1 ‘tank system electrical control’ in Figure 3.3.4.

There may be some components that are monitored by more than one sensor. An example from Figure 3.3.4 is ‘Main Tank’ in the ‘Tank Assembly’ Hard System platform. This component is monitored by Sensors S1 and S2. There is no significance in the order in which these sensors are placed on the diagram in relation to ‘Main Tank’. The intention is purely to show that these are electrical components, performing a monitoring function.
Figure 3.3.4 The complete modelled view of the system described by Figure 3.2.1
Note that the ‘tank system electrical control monitoring’ interfaces only with the ‘tank assembly’, the ‘tank system electrical reliability monitoring’ and ‘tank system electrical control’ interface only with ‘tank hydraulic controls’. This is an important point as it demonstrates that reliability issues are not confined to the system being controlled (the main tank) but are also a function of the hydraulic control system which could potentially be less reliable than the main tank system.

Such a viewpoint allows questions to be raised about not only the design of the system but also the purpose of the system as a whole. For example, is the use of flow meters to monitor the control system an expensive, but necessary course of action? Could other means of modelling flow (from the control system outputs) be used to provide a lower resolution, but serviceable (in both technical and commercial terms) solution to discovering failure? Would both types of monitoring approach be required to optimise or provide a further check and balance to provide a high resolution solution to reliability modelling?

This viewpoint also supports modelling of components which have more than one interface. The flow meter is a prime example of this, there is an interface with this component and the hydraulic system (as it measures flow) and an interface with the electrical system (for data analysis and processing of the transducer measuring the flow). Hence the component ‘VF1’ appears both in the ‘tank hydraulic controls’ and the ‘tank system electrical reliability monitoring’ system platforms in Figure 3.3.4.

In referring to this developed Hard System model, the following should be remembered:

- This model is not intended to act as a functional system model;
- The model can be used to define technical and contractual boundaries;
- The model defines system interfaces and partitions which may not be obvious from a system design viewpoint, but gain significance in reliability terms (e.g. is the component monitoring for reliability and control purposes or solely for control purposes);
• The model defines a contextual relationship between components (such a viewpoint may therefore be used to establish failure modes, combinational failure scenarios or failure indicators);

• The model shows which components are directly monitored and how they are monitored;

• The model shows which components have more than one system interface, which differentiates between those components that are purely mechanical and those that are mechatronic;

• The model is not meant to provide information for safety calculations but could feasibly be used as a basis for a qualitative reliability analysis;

• The model can be expanded to include other systems at either a higher or lower level. This is allowed through the interface modelling approach adopted.

### 3.4 Developing the Hard System Reliability Risk View

The Hard System Reliability Risk View is an annotated version of the Hard System model (Figure 3.3.4) and gives a graphical indication of:

- System vulnerability;

- Monitoring system capability;

- Monitoring system coverage.

Component categories are represented on this view highlighted by particular colours. Each category indicates system component vulnerability. Once complete, reliability monitoring system coverage and depth of coverage (in terms of whether or not all failure modes are detectable) becomes clear, the Hard System Reliability Risk View then provides a graphical indication of where (or if) further development of the system is required.

There is a further opportunity to include FMEA information with respect to monitoring on this diagram if it is available. If it is and there is information on
which monitoring systems are used to detect failures of which components. This information can be written on the diagram in the following format:

<fault mode type>, <component label>

Where fault mode type can be expressed as either of these 3 letters, I = Indicative of failure, P = Predictive of failure, S = Secondary, Supporting or Suspect failure>

Taking such an approach identifies which components are central to the monitoring function and as such provides an opportunity to reassess the robustness of the design taking into account this view.

### 3.4.1 Monitored components

Firstly the Monitored components should be added to the Hard System Reliability Risk View. This is done by deciding which components fall into the category defined in section 2.10.8.1 and then by changing the colour of the component box from black to green in order to indicate the component is of the monitored component category.

Figure 3.4.1, shows the monitored components. Monitored component assumptions:

- The tank level sensors are monitoring sensors and as such their failure will be detected by software;

- The controller is a monitored component (its action monitored by particular flow meters and driver circuitry for the controller) and its failure will be detected by fault accommodation in software;

- Valves V1 and V3 are automatically operated and therefore their failure can be detected by fault accommodation in software and by flow meter data.

- The main tank levels are controlled and monitored and therefore tank failure can be detected by fault accommodation in software, through a modelled response which corroborates component control demands and sensor readings.
Figure 3.4.1 Hard System Reliability Risk View – monitored components
3.4.2 Non – monitored components

Next non - monitored components should be recorded, again, the defined category guidelines provided in section 2.10.8.2 and the following assumptions develop this view. Again as above, all those components that are believed to be non-monitored components have their ‘black box’ notation coloured red.

Non - monitored component assumptions:

- The spill sensor (SP1) is:
  - Not part of a closed loop control system;
  - Is therefore a back up rather than primary failure detection sensor;
  - May not have associated Built In Test (BIT)/ fault detection software.

- The failure of SP1 may not be detectable therefore the failure modes that are detected using the spill tray sensor may also not be detectable. The spill tray therefore should also be categorised as non-monitored until otherwise determined;

- The flow rate sensors (VF1, VF2, VF3) play an important role in determining various system failure modes as they are referenced as being able to detect failure modes in Hurdle et al [47].
  - We can therefore reasonably assume that their output is automatically monitored but even so, there is doubt on how this is done and so the component should be categorised as such;
  - We also have no information on flow meter failure mechanisms and how they are detected;
  - In this case therefore the component is designated non monitored until we can substantiate our assumptions on both how flow meter outputs are recorded and the ability to detect flow meter failures.
Figure 3.4.2 Hard System Reliability Risk View - monitored and non – monitored components
3.4.3 Remaining components

All those components that do not fall into the above categories must now be considered, guidelines for this type of component are given in section 2.10.8.3. These components have their 'black box' notation coloured amber. Both these components and those components that are coloured red indicate to the observer that further definition work or design mitigation required against these components. Just how much mitigation is required depends upon:

- The criticality of the component in terms of system availability;
- The maintainability of the component (how easy is it to repair or replace);
- The availability of maintenance resource at the location of system failure.

Generally though, those components in the amber category are those which provide either difficulty in being monitored completely or being able to have all of their failure modes categorised and so the mitigation case for them is assumed to be more difficult.

Figure 3.4.3 shows the Hard System Reliability Risk View in terms of component categorisation, it takes into account the definition of remaining components by assuming that V2 is just such a component.

The V2 valve is manually operated and so the assumption is made that the failure of this valve is difficult to detect and verify as the command to open and close the valve is not known to the system and as such can be considered to be a disturbance emanating from outside the system boundary as a whole. Only the effect of a failure can be determined by the system and this failure effect cannot be internally corroborated as a definite failure mode.
Figure 3.4.3 Hard System Reliability Risk View showing all component categories
3.4.4 Adding FMEA information regarding detection modes to the Hard System Reliability Risk View

There may be occasions where the FMEA explicitly states which sensors are used to monitor the reliability of a particular component. It is of particular benefit to show this information graphically as such information is:

- Likely to be dispersed throughout the FMEA document;
- Usually partitioned on a functional basis, and so is not available as a holistic view.

The Hard System Reliability Risk View however, enables this FMECA information to be seen in context.

This approach requires that the information is shown as in Figure 3.4.4. For example, if controller C1 fails, then valve V1 is not controllable, this lack of control suggests that C1 has failed. This is denoted as follows:

```
<component>  <I=indicates failure>    <of component>
  "<P=Predicts failure>    "
  "< S= Supports or Suspects failure >  "
```

This notation is placed against the component which is performing the monitoring function as shown in Figure 3.4.4.

Note that the following components have this notation applied to them:

- Controller C1    IV1 (valve V1 is not controllable if C1 fails)
- Controller C2    IV3 (valve V3 is not controllable if C2 fails)
- Sensor S1        IV1, IV2, IV3 (there is a potential failure in one or all of these valves if this sensor shows an exceedence)
- Sensor S2        IV1, IV2, IV3 (there is a potential failure in one or all of these valves if this sensor shows an exceedence)
Some components will be able to detect similar failures, but are not listed on this initial view as their status is unresolved (they are presently categorised as non-monitored components). If their status is resolved and they were confirmed to be monitoring components then the following notation would apply to these components as follows:

Flow meter VF1  SV1;
Flow meter VF2  SV2 (although this cannot be achieved automatically – manual inspection is required);
Flow meter VF3  SV3;
Sensor SP1     main tank, SV3, SV2 (the sensor could either indicate a failure of the main tank or could imply suspected failure of either V2 or V3 or both valves).
Figure 3.4.4 Hard System Reliability Risk View – adding FMECA information
In summary, this Hard System Reliability Risk View diagram shows:

- The vulnerability of the system (exposure to reliability risk, indicated by how many components are not monitored);
- The vulnerability of the monitoring system (a measure of how effective the monitoring system is, indicated by yellow or red boxes which show either no monitoring or undefined monitoring conditions);
- Dependency on other systems which feed this system;
- Dependency of other systems on this system;
- Which sensors are responsible for monitoring mitigation (provides a graphical view of any failure detection defined in the FMEA) or suggested, if there is no FMEA available.

The Hard System Reliability Risk View should be considered to be a ‘live’ diagram during the design process. Any system ‘improvement’ should be recorded as the design changes iteratively.

### 3.5 Simple Soft System example description

The Gantt chart in Figure 3.5.1 shows how an organisation may respond to a maintenance event indicating a fault that has been produced by a monitoring system.

The chart is read from left to right and shows tasks that must be performed (the blue bars), in the sequence in which they are performed to bring the response to the maintenance alert which indicates and confirms a known failure before scheduling and making a repair. The open diamond shapes represent phase deliverables where there must be a handover of information or action before the plan can be progressed.

In this chart, a new task is not started before its previous task is completed. This is not always the case, but is shown as such here to provide a more clear understanding of the technique adopted in this research.
Figure 3.5.1 A Gantt chart representing a maintenance action
Such a chart provides a great deal of information on the tasks in hand. These charts can however be limited in terms of providing an understanding of the process that lies behind the actions that need to be taken and do not necessarily indicate who has the responsibility to complete, validate the completion of, or manage the task described. The methodology described in the following sections attempts to correct this by assigning responsibility.

For complex programmes the charts become large and often difficult to assimilate especially if there are a number of dependencies between tasks or organisational boundaries.

3.5.1 Assumptions regarding the overall system boundary of the example

In this case, the overall system boundary is defined as the extent of the Gantt chart.

In practice, this boundary is defined by the steps that have to be taken in order to complete the task in hand. It should be remembered that the modelling technique suggested here has the potential to greatly increase the boundaries of the system if issues such as responsibility for various steps are included (for example approving and signing off a repair action). Such issues may not necessarily be reflected on a Gantt chart but they are important considerations when the total process is under analysis.

This attribute of the modelling technique is an important one. When modelling Hard Systems the system may be modified but is not likely to change greatly. When modelling Soft Systems, especially those of large organisations, the systems are subject to constant change and improvement. A full understanding of the effect of those organisational changes (and therefore the limit of the overall system boundary) is absolutely necessary when attempting to analyse the Soft System.
3.5.2 Assumptions regarding the organisational structure of the example

Organisational structure, be it matrix, hierarchical or any other form is also not well represented using either Gantt Charts or flow charts and so the modelling technique will need to take account (but not be driven by) the organisational structure. In analysis terms, the structure of the organisation is an important factor as the aim of the analysis should be to reduce the amount of task and information sharing between various parts of the organisation, thus streamlining the operation.

It should also be remembered that organisations (for a variety of reasons) may not wish to change their structure to provide a 'streamlining' of the service. This is acceptable as long as the model and analysis reflects this position accurately such that the necessary boundary interfaces created by the organisational structure are well understood and can be serviced appropriately.

3.6 Modelling the example Soft system

Note that the notation used is similar to that of the previously defined Hard System model representation. This is no coincidence as it is an aim of this research to develop a position from which both Hard and Soft System artefacts can be combined in the same representation. Of course such a representation is only valid in a reliability sense if the 'reliability' attributes of both systems are compatible.

3.6.1 Determining the Soft System boundaries for the given example

Determining Soft System boundaries from a flowchart representation is not easy unless responsibility for the tasks and decisions are made clear on the flowchart.

For the Gantt chart representation (providing resources are identified) it is relatively straightforward. The resource allocation guides the definition of the system boundary and that boundary is defined by either personal or departmental responsibility or function.
In this particular case therefore there are three boundaries, one is the Hard System output represented by the ‘failure alert’ being provided to the maintenance organisation, one is the maintenance organisation itself and the other is the organisation with which the maintenance organisation has to interact (the organisation that is providing the service to the customer).

The Hard System output represented by the ‘failure alert’ in the flow chart may not normally be included in legacy documentation but is included here for clarity and also to raise the importance of the interface between the Hard and Soft aspects of the reliability monitoring system. How data or monitoring system control inputs to the Soft System are transferred and how quickly that transfer is made across this interface is of prime importance as it may affect contract guarantees of speed of maintenance diagnosis and repair.

It should also be noted that there are different aspects to this interface for each system type. The Hard System interface can be thought of purely in terms of its physical properties and can be assessed in those terms, but the nature of that interface directly affects the performance of the Soft System.

If not enough data, incorrect data or data which arrives in an untimely fashion is transferred to the Soft System across the interface this will impact on the reliability of the system as a whole. Appropriate and detailed modelling of this aspect of both Hard and Soft Systems is therefore very important.

Note here that the Gantt chart accounts for time in which it expects the alarm decision to be created and sent. Although this task may be facilitated by the Hard System, there may also be some kind of human interaction involved in obtaining and sending the data, therefore this is reflected in the Gantt chart as an interface.

Each of these system boundaries partition the system and are defined by either personal or departmental responsibility and are denoted in a diagram as grey boxes (see Figure 3.6.1)
Figure 3.6.1 System boundary definitions of the response to a maintenance alert based on the Gantt Chart shown in Figure 3.5.1 for 'predicted fault'.
3.6.2 Allocating tasks to the System boundary definitions

Once all the System Boundaries have been defined as above, tasks should be allocated appropriately to each system boundary in a similar manner to that adopted for Hard Systems where components were allocated to their specific system boundaries.

Again, this is a fairly simple step as to some extent the resource definition and responsibility is inherent in the task if the plan is produced correctly.

Taking the Gantt chart in Figure 3.5.1 as our reference for this step, the ‘bars’ on the Gantt chart are those that are considered to be tasks. They are considered such as they will have a resource component attached to them (that is they last for a finite time defined primarily by the length of time a person or person(s) skilled in the art will take to perform the task). Adding these tasks to the system boundaries as annotated black boxes provides the model shown in Figure 3.6.2.
System Fault accommodation
- Abnormal behaviour detected – indication of failure

Maint org fault accommodation
- Confirm failure
- Undertake repair

Service scheduler fault accommodation
- Schedule maintenance action

Figure 3.6.2 Model showing tasks assigned to Soft System boundaries for ‘predicted fault’
3.6.3 Arranging and connecting system boundaries and tasks to produce the Soft System model

Allocating tasks will only show who has the majority of task responsibilities. This however is still a useful view as this representation could provide an insight into the process which some involved may not recognise or agree with. This is seen as a benefit as it allows scope for negotiation of task responsibility apportionment between those involved in the process.

It should also be remembered that the plan that is being used as a source for this model could be incomplete or incorrect. Again this is seen as an advantage of the technique as identifying such deficiencies should lead to an overall process and subsequent planning improvement. We shall assume for our purposes however that the Gantt chart is complete and correct.

The next step is to arrange the system boundaries and connect them such that they provide a model of the Soft System as a whole. To enable this, an understanding of the interfaces between each system boundary is required.

Interfaces in this context are suggested by the plan ‘milestones’ or the empty diamond shapes on the Gantt chart given in Figure 3.5.1. This is because milestones usually indicate programme deliverables. Programme deliverables, by extension, are usually produced to satisfy a need from another system boundary for information, data and approval. Therefore, in the majority of cases milestones can also be seen as interaction points in a plan. Note also that this holds true in general when departments produce internal milestones. The deliverable milestone may well be for internal consumption, but it satisfies needs between those involved in process delivery in the department. Such internal milestones are not recorded here as they are assumed to be part of the task scope and resource allocation for that system boundary.

It is also worth noting that for a Gantt chart, the progression of the plan is in general sequential in nature (but can also be concurrent) and this is also reflected in the modelled representation, with ‘process flow’ in Figure 3.6.3
moving from top left of the model (the start point – ‘abnormal behaviour detected – indication of failure’) to the bottom right of the model (the end point – ‘confirm repair complete’).

Such a sequential and logical progression allows us to place the interaction ‘milestones’ in the order in which they occur on the Gantt chart. The milestone is represented by a ‘white box’ in each system boundary between which the interaction takes place. These boxes are connected by a directional flow line which indicates the direction of ‘process flow’ (in the form of information/ approval exchange).

Note that before a task may ‘begin’ information via an interaction must reach it. The linkage between the interaction and task notation in a system boundary itself is denoted by a horizontal black line. The ‘process flow’ follows this path until it comes to the next action which then takes place.

For example, the ‘maint org fault accommodation’ system boundary receives a failure alert from the ‘systems fault accommodation’ system boundary. The ‘maint org fault accommodation’ system boundary then performs the task ‘confirm failure’. Once this task is complete, it informs the service scheduler (the details of how this is accomplished are not shown, but are important in analysis terms). The process continues in the same manner through the ‘service scheduler fault accom’ system boundary and the ‘maint org fault accommodation’ system boundary goes into a ‘wait’ state until the ‘service scheduler fault accom’ system boundary provides its input ‘inform of service schedule’.
Figure 3.6.3 The completed model of the Soft system represented by the Gantt chart in shown in Figure 3.5.1 for ‘predicted fault’.
3.7 Developing the ‘Soft System Reliability Risk View’

Once the Soft System model is developed, the Soft System Reliability Risk View is developed by highlighting the reliability vulnerability of the model in much the same way as with the Hard System Reliability Risk View in order to promote understanding of system vulnerability and form a basis for analysis.

In this example, we are attempting to understand not only overall response time and repeatability of that response time as discussed in section 2.10.14 but also in this case the type and variety of responses that are possible to different monitored system alerts (for example, indicated faults, predicted faults and suspected faults). This variety of responses is driven by the monitoring capability identified in the Hard System Reliability Risk View and requires us to define Soft System models beyond that defined in Figure 3.5.1 which only accounts for actions to be taken on an indicative fault being detected.

Such views are therefore not tied in to current system documentation as it is assumed that no such documentation currently exists.

As with the Hard System Reliability Risk View, the reliability risk for the Soft System model is highlighted by particular colours in order to provide a graphical indication of where (or if) further development of the Soft System is required.

In the case of the Soft System, for each particular model (depending whether an indicative, predicted or suspected fault has been detected) only the interfaces are annotated as the interfaces are generally the points of vulnerability and are often not described in as much detail (or monitored) as the tasks which precede them. This is not to say that the tasks will not form part of any analysis as their times will be known and when summed will contribute to the overall maintenance response figure, but this is of less importance as an issue to the reliability of the process as a whole than the interfaces between system boundaries. Task times are not shown for this particular example.

The Soft System Reliability Risk View developed below does not use the Soft System model shown in Figure 3.6.3 as a basis as this would have only
automatic interfaces. Instead a new model is developed using the same system boundaries but showing how these would interact if a ‘suspected fault’ were detected.

3.7.1 Automatic interface

Firstly, automatic interfaces should be shown, following the guideline definition provided in section 2.10.15, the interfaces on the model are shown as automatic interfaces by colouring the interface boxes in both system boundaries as green (see Figure 3.7.1).

The assumptions made in developing this view are as follows:

- The failure alert will be automatic;
- Maintenance scheduling should be automatic through use of a scheduling software package.
Figure 3.7.1 Showing automatic interactions of the ‘Soft System Reliability Risk View’ for ‘suspected fault’
3.7.2 Manual interfaces

Next manual interfaces should be shown (see Figure 3.7.2), following the guideline definition provided in section 2.10.15 and assuming that the other model interfaces are manual interfaces, colour the remaining interface boxes in both roles red.

The assumptions made in developing this view are as follows:

- Any request for data will be made by someone skilled in the art and the decision will be made manually;
- There is a conservative assumption that there is a restriction to data access (large scale downloading of data could be either a technical or commercial restraint);
- Failure recommendations based on data will be manual as it will require someone skilled in the art to make those decisions;
- Authority to confirm completed repair will be made manually.
Figure 3.7.2 Showing manual and automatic interactions of the ‘Soft System Reliability Risk View’ for ‘suspected fault’
3.8 Resultant risk exposure

As discussed in step 7 (Section 2.10.16) both the Hard and Soft System Reliability Risk views are reviewed and changes made to the design of either system if necessary.

Such changes will be dependent on product design, organisation and commercial factors.

3.9 Chapter Summary

This chapter has described the application of the proposed methodology to a simple Hard System which has a notional reliability monitoring system function that provides alerts to maintenance personnel in case of failure. For this simple example, the actual transmission mechanism for these alerts is not modelled. Instead it is assumed that any monitoring sensor modelled in the Hard System will initiate instantaneously the Soft System process at ‘Abnormal behaviour detected – potential fault’ in the System Fault accommodation Soft System boundary (as shown in Figure 3.7.2). Such a scenario is likely to be a common one when considering the transformation of a legacy system to Servitised business.

This chapter also describes an application of the proposed methodology to a simple Soft System maintenance response.

By modelling both systems a viewpoint is provided through which aspects of both Hard and Soft Systems, such as the design of alert transmission methods and mechanisms, can be fully considered and consequently the design of both systems optimised in concert to provide the best service, thereby improving Servitised system availability.

The methodology is seen as an enhancement to existing methods of reliability assessment and does not seek to replace them. It does however seek to extend the reliability assessment approaches developed for physical or Hard Systems to task based or Soft Systems defined by the maintenance response.

The methodology seeks to answer the following questions:
• What is the exposure to the business of product failure?
• What mitigation can be made against that exposure?
• Is the technological and enterprise infrastructure in place in Hard and Soft Systems terms to accommodate a service model of this type?

These questions require an understanding of the design weaknesses of both Hard and Soft Systems. Modelling these systems and adding ‘reliability’ information to the model aids this process.

3.9.1 Hard System Reliability Risk Modelling review and general guidelines

A model of the Hard System has been developed, this model is not a functional representation of the system, but does establish, platforms, components within those platform boundaries and interfaces between those components in order to establish a context for those system components and the type of interface which connects them.

The importance of establishing these facets of the system is one which is of as much a commercial concern as it is technical. Responsibility for certain aspects of the system whether they are technical or commercial, is a necessary part of developing and integrating the system into the overall business plan.

Once these facets of the system are modelled, they are further analysed in order to ascertain:

• Which components are monitored;
• Which components perform the monitoring function;
• Which components are vulnerable (those either not monitored or partially monitored);
• The level of coverage of the monitoring system (are all system components monitored automatically);
The capability of the monitoring system (are there any components which could cause major reliability issues either by not being monitored or having a failure mode which cannot be monitored)

The final two points are most important, especially when reviewing early stage system designs for reliability monitoring systems.

In the case of vulnerable components, one which is not monitored but fails regularly could adversely affect an availability contract by failing at a time that causes an extra unplanned maintenance burden.

The level of coverage is also important in terms of the Soft System maintenance response. If there are only a few components which can be monitored, there is little point in developing a large support network to manage the data and resulting maintenance actions arising from alerts provided by such components. The wise thing to do in this case would be to choose to provide a regular maintenance contract rather than to attempt to develop a Servitisation business contract.

Information from other reliability sources can be added to the diagram to enhance understanding of the monitoring system and its capability such that these points can be more clearly understood. This is done by listing monitoring fault detection modes against the monitoring components that are responsible for detecting them.

Once this full understanding of the reliability monitoring system, its coverage and capability is developed there is an opportunity to either proceed with the current design; modify the design to achieve the desired capability which is likely to be able to support the availability targets of a Servitisation business contract or to tailor the availability target offering such that it is consistent with the reliability monitoring system design.

3.9.2 Soft System Reliability Risk Modelling review and general guidelines

A model of a Soft System has been developed as an example maintenance response to anomalies detected by the Hard System reliability monitoring components. Although not shown here, there could feasibly be a number of
such models depending upon the type of failure detected or the type of monitoring system component which detects the anomaly (for instance if a failure is detected and confirmed, there may be a specific process used to deal with this type of failure).

The type and variety of such models will depend upon the capability and coverage of the reliability monitoring system in terms of the type of alert that is produced. For example one type may be used for a definite and confirmed failure and another type for potential failure requiring further analysis. The organisational response required for either to meet availability targets set by the Servitisation contract may differ considerably.

In addition if not all failure modes can be detected, there will need to be a separate Soft System model which will mitigate these types of failures. Such models are not required to be integrated with the Servitisation Soft System models and so are not discussed further here but it is recommended that they should be produced in order to ensure a complete understanding of the maintenance response as a whole is developed.

The Soft System models are produced in order to establish boundaries of responsibility around the tasks that are required to be performed as part of the maintenance response. They also outline the interfaces between those tasks and boundaries, providing a context for both the working relationships and division of responsibility between each Soft System boundary.

The importance of establishing these facets of the system is a commercial and also a general business operational concern. Understanding which aspects of the organisation (or partner organisations) have responsibility for certain tasks is the key to developing profitable support contracts and efficient, integrated business operations.

Once these facets of the system are modelled, they are analysed to ascertain:

- Points of system vulnerability (these relate specifically to the type of Soft System interface as it is important that these are well defined so that the information required to carry on the process is available to all parties);
A quantitative view of the maintenance response time, such that business improvements can be made or contractual agreements monitored. This view can only be developed if plans or some other information source that refers to resource loading is used to develop the Soft System model. Such a view takes into account not just the interface but also the time it takes for tasks to complete.

Such an analysis provides confidence that the organisation that supports the maintenance response required by reliability monitoring system indications is capable of performing that function at a performance level and cost base that is in line with contractual response times to availability issues identified by the reliability monitoring system.

If the analysis suggests that the contractual response times cannot be met then the options are to either redesign or reconfigure the Soft System to ensure times can be met, or to make infrastructure improvements to, for instance, increase the speed and efficiency of data handling or failure analysis support techniques.

The Soft System Reliability Risk Model will help to identify where these improvements can best be made.
Chapter 4 - Critical appraisal of the application of the methodology to a simple system

4.1 Introduction

For the most part the requirements for the methodology have been met and the simple system discussed in the previous chapter has been modelled and in addition a nominal assessment has been made regarding the vulnerability of both the Hard and Soft Systems to reliability issues.

From this activity however there are areas which have been identified where the methodology will need to be developed further. These identified constraints are listed and discussed further along with corresponding suggested enhancements to the methodology below.

4.2 Identified methodology constraints

There are a number of issues that can be laid against the proposed methodology following implementation using the simple example in the previous chapter. They are listed as follows:

- The lack of formality of the notation in comparison to the more formal Role Activity Diagram notation from which it was derived can lead to;
  - Ambiguity in applying the notation.
  - Interpretation of the system model in a number of ways.

- The Hard System example shown did not include or take into account;
  - Redundant systems.
  - Distributed Systems.
  - Multi lane (hot or cold standby) systems.
  - The impact of software on the operation of the hard system.

- The Hard System assessment technique does not take account of software reliability;
The Soft Systems notation does not allow for;
  - When a part of the process is to be repeated for any reason.
  - When the process has conditional restraints which guide the path of the process

The example given was a small scale system in terms of both Hard and Soft Systems. Larger and more complex systems would mean that the diagrams produced using this notation would themselves become very complex and more difficult to read;

Modelling the system runs the risk of duplicating effort as in most cases the system will already have some form of representation. Integrating an assessment technique with existing models or available system information is therefore desirable;

Hard and Soft System reliability assessments are primarily qualitative (although a quantitative approach for Soft Systems has been suggested) and therefore require an approach to quantitative assessment to be developed.

4.3 Discussion in response to the identified constraints

4.3.1 Lack of formality of notation

The relative lack of formality of the notation is not entirely a disadvantage, it can also be said to have some advantages.

It must be remembered that the notation is used primarily to develop a model of the concept in as simple a manner as is possible and in assessment terms does not necessarily have to accurately reflect function. The main requirements of the model for this methodology are that:

- For the Hard System:
  - A boundary of either a technical or commercial nature is identified as a system platform;
  - Components within these system platforms are identified and represented according to the boundary constraints;
o Interfaces between those components are identified;

o The interfaces show an accurate relationship with the components they are related to.

- For the Soft System:
  
o The boundaries of responsibility for each aspect of the Soft System are identified;

o All tasks are identified and allocated within their respective system boundaries;

o The interfaces between system boundaries are identified and defined at least provisionally.

As long as these main requirements are satisfied, the model has validity as far as this research is concerned.

In terms of ambiguity of interpretation, the main aim of developing a modelling notation such as this is that it is accessible to all and not just the technically literate. It has to be recognised that certainly in terms of Soft System modelling, those that are not technically literate will have a say in defining what part they play in the Soft System as a whole and so this approach will encourage them to contribute. There is a caveat that someone who is technically literate must then validate this contribution in order to develop an analysis. This approach allows the analyst to develop a thorough understanding of the system under analysis. This approach partially justifies the lack of formality, but the most important aspect of the modelling notation is that its aim as a whole is to be used collaboratively.

In this respect all those involved in the analysis of both Hard and Soft Systems should gain an appreciation of the model through this collaborative process in order to collectively develop a singular view of the model.

Although Role Activity Diagrams themselves have a very formal and precisely defined notation to support Soft System modelling, this formality is not necessarily required when modelling a Hard System. The aim of modelling the Hard System is not to perform an analysis of the design (which is possible with the formal notation of Role Activity Diagrams) but to establish
relationships in the same manner (but with less rigour) of that of a pipe and instrumentation diagram. The aim of the Hard System model is purely to highlight interface relationships and the relationship of the component to the sensor which is capable of monitoring that component. This information may not be explicitly stated in the design drawings which specify the Hard System. The modelling notation therefore is felt to be sufficiently formal to support the main requirements bullet points listed earlier in this section and so no improvements in this regard are suggested.

It is however recognised that there is a limit to the size of such models beyond which a holistic appreciation becomes more problematic, this issue is discussed further below in section 4.4.4.

4.3.2 Hard System redundant and standby systems

These types of system are an important consideration in any safety and reliability assessment as they are often used to improve reliability and hence the availability of the system.

Such systems however relate mainly to the functional safety of the Hard System. Their relationship to component reliability is implied but is not central to the safety argument. A premise of the research is that it will not take the place of the current formal methods of safety assessment of the system design as a whole but will provide a complementary approach which will inform system design based on its assessment parameters which relate solely to the operational effectiveness of the reliability monitoring system.

In addition, a Servitised product will often, but not always, be a legacy product and therefore will have been designed to a different set of reliability targets to that now required to meet the new Servitisation business case where the existing monitoring system will have been introduced to accommodate safety and not necessarily reliability design issues. In this case, there is a risk that credit may be taken for the system in both a safety and a reliability sense. Such a situation requires careful consideration as the safety case which is required to achieve certification of safe product use takes precedence over reliability issues which are governed primarily by commercial considerations.
The proposed methodology also includes Soft System appraisal which also may have an effect on Hard System design once complete giving a holistic viewpoint of the operational reliability of the reliability monitoring system. It is therefore appropriate to segregate functional safety and operational reliability aspects of the system design.

In terms of the proposed methodology, in dealing with Hard System redundant and standby systems, the same rules of modelling notation apply and if the redundant or standby system is part of the same Hard System platform, then it should be represented as such.

Although there is a potential for more complex drawings to result as long as the rules regarding systems platforms are complied with then these should be manageable.

Some extension to the notation is therefore required to model such systems.

When doing this it is worth remembering that the majority of safety critical redundant systems consist of electrical and software control system components. This is due to the nature of the failure mode of this type of system which is usually of a rapid nature whereas the systems that are controlled by electrical and software components (mechanical, hydraulic and pneumatic) are relatively too complex to duplicate, too robust to require duplication or too expensive to recreate unless under exceptional circumstances.

4.3.3 Distributed systems

Distributed systems have not been modelled using this approach but it is felt that the approach will apply to a limited extent as there is a flexibility of use in such systems controlled mainly by software strategies which the methodology has a limited capability.

The application of the methodology to these types of system is discussed further below in section 4.4.3.
4.3.4 Software reliability as part of the Hard System

In any automatic control system and in any reliability monitoring system software will play a crucial role in either the control of the system or interpretation of the state of reliability of the system.

This assessment methodology has to assume that any software function involved in the system whether it be for data acquisition, data processing or data communication is fault free and accessible at all times.

Obviously this is not always the case, but this assumption is supported by current assessment techniques for safety and reliability where software is used in often safety critical applications but does not form part of the reliability assessment of the hardware. It is analysed separately using specific and defined methods particular to software applications.

In terms of software reliability for Soft System applications, the same assumption is made as for Hard Systems, here though the effect of software/computer network unavailability is potentially exposed through timings of ability to process or transfer reliability monitoring system data around the Soft System.

There is therefore no need to make an enhancement to this approach for the reasons discussed in this section.

4.3.5 Repeatable and conditional Soft System steps

Often Soft Systems are represented as lengthy flowcharts (regardless of which technique is initially used to model the Soft System process). Using a flowchart technique encourages close attention to detail as it has a level of formality due to the use for which it was developed as a guide to software development.

Using such an approach, the Soft System can break at certain points depending upon certain conditions being met and re-enter the flow of the flowchart at points determined necessary. In other words the flowchart acts like a Soft System program which leads the viewer to different parts of the system should particular conditions be met. This level of detail in the Soft System model can sometimes be an advantage and can seem to be
necessary. This approach does however lead to large and complicated Soft System representations which may or may not actually reflect what people do on a day to day basis.

The Soft System is represented this way in order primarily to provide guidance for how staff should go about their business and for that process to be officially audited, but in reality it is unlikely that all those who have a role in the Soft System will behave in the way described by the flowchart. Some leeway of interpretation of the process will be allowed either knowingly or unknowingly due (for instance) to levels of planning experience. This is evidenced by the fact that Gantt charts for programmes or projects will differ from the flowcharts that are meant to inform them. Again, this may not always be due to the fact that flowcharts and Gantt charts show the same information but in a different way, it can be forced through pragmatism of business circumstances not considered at the time of the development of the flowcharted standard process or because the flowchart was incorrect when it was developed initially.

To resolve this, if we compare the flowchart approach with that of a Gantt chart, which is a standard approach used to plan programmes, the Soft System representation is kept to a minimum. The main attributes are:

- What are the tasks;
- Do the tasks have predecessor tasks;
- Is resource (ownership) assigned to the tasks;
- What are the deliverables (outputs);
- What is the context between the tasks and the deliverables?

Here the level of detail of what people do to fulfil the aims of a Soft System is much more restricted, there are no defined opportunities to for example, if a task is not complete, branch out of the Gantt chart and suggest that all other tasks halt until that task is complete or the whole process is started again from a different point. These are by no means the only differences between the capabilities of both these or any other process mapping techniques, but it does show that unlike Hard System modelling, Soft System modelling is
relatively subjective and has a perspective which is in line with the final use of the information that is modelled.

The methodology’s modelling notation is consistent with view that the end use of the Soft System model shapes (to a certain extent) the notation and is also consistent with the aims of the two modelling techniques discussed which in the first instance relate only the minimum level of information used to determine tasks, their relationship and ownership (like the Gantt chart) and in the second instance show those tasks in context with other tasks in the Soft System in which there is a complex interaction in the performance of these tasks (as would be possible with the flowchart technique).

It does not allow for repeatability of tasks or for non–sequential (other than concurrent) scheduling of tasks. This would be allowed in a flowchart but is suggested to be a feature of flowcharts that is of questionable value as it by necessity sets the task definition at too low a level for useful analysis to take place.

The methodology defined here assumes tasks at a level of abstraction which provide some potential for measurement in hours or even days to complete (if human based and requiring decisions of how to proceed given certain circumstances). Such an approach allows for sufficient differentiation between tasks but also provides a method for identifying task areas which may be resource intensive and thus affect the Soft System operational reliability.

If the breakdown of tasks and their content in this way highlights operational reliability concerns then the options are:

- To investigate the task content at a more detailed level in order to understand whether improvements can be made;
- To investigate the infrastructure which supports task execution.

For automatic interfaces and tasks, any kind of decision making of the type discussed above is expected to be inherent in the rules of the automatic interface and therefore be consistent and repeatable. In such a situation any timings relating to the interface will be bounded by these inherent rules and so
will not affect operational reliability as would more variable human decision making processes.

There is therefore no need to make an enhancement to this aspect of the methodology for the reasons discussed in this section.

4.3.6 Large system complexity

All Soft Systems modelling techniques have problems in addressing large scale systems as inevitably large complex systems lead to large complex models where interactions within the system become more difficult to identify along with a greater difficulty in defining and ultimately understanding their importance. This is particularly true of the modelling notation developed here. The modelling examples developed so far have been relatively simple examples are already large even though they are not necessarily complex.

A method is required therefore for extending the technique to address this larger scale complexity which both retains model integrity and context and allows important interactions for both Hard and Soft System modelling approaches.

4.3.7 Quantitative assessment techniques for Hard and Soft Systems

Although the assessment techniques demonstrated on the simple system are useful and may in some cases be all that is necessary, it would also be beneficial to develop separate quantitative assessment for the models. Failing that it would be advantageous to integrate the approach with other quantitative assessments that are performed as part of the safety case, especially with respect to Hard Systems.

It is important to gain an understanding of these quantitative assessments as they provide an indication of which components are most likely to fail and therefore which are most in need of a monitoring strategy to support them.
4.4 Suggested extensions and enhancements to the methodology due to the constraints listed above

As identified by the above discussion, there are clearly areas of the methodology that need to be developed further in order to deal with the problems of:

- Modelling redundant Hard Systems;
- Modelling multi lane (hot or cold standby Hard Systems);
- Modelling distributed Hard Systems;
- Modelling larger, more complex Hard and Soft Systems;
- Integration with existing or suggested new assessment techniques.

Proposed extensions and enhancements to the methodology relating to these areas are discussed further in sections 4.4.1 to 4.4.8 inclusive.

4.4.1 Modelling redundant Hard Systems

In order to apply the notation to redundant systems a new Hard System example is required. Extending the simple tank example discussed previously to that shown in the Figure 4.4.1 provides such an example.

The Hard System design in this case has a tank which is supplied with fluid by two pumping systems which are independent but identical. The pumping systems are controlled and monitored by separate, but identical control and monitoring systems, providing system fluid input redundancy.

The tank level is controlled by a tank control and monitoring system based around inputs from sensors monitoring tank fluid levels and thence by control of the powered outlet isolation valve and the inlet powered isolation valves.
Figure 4.4.1 Multi input tank system
The two flow supply systems provide flow to the tank either separately or jointly with the tank control and monitoring system being the arbiter of tank fluid level and control of the flow to the tank dependent upon control of powered isolation delivery valves which set the contribution level from each pumping system to the tank. If each isolation delivery valve is set open then the level of fluid supply is determined by the control settings of the pumps through each pump’s control and monitoring system. There is no defined input source to the tank.

4.4.1.1 Applying the notation to the example of system redundancy

Using the rules of the methodology, the above description and the schematic shown in Figure 4.4.1 the main platforms of the Hard System can be said to be:

- The tank being supplied with fluid;
  - As it is the subject of the control & monitoring system.
- The fluid level control and monitoring of the tank;
  - As it has an electrical interface and performs a specific function on the monitoring system subject.
- Pump System 1 Hydraulics;
- Pump System 2 Hydraulics;
  - Pump system 1 & 2 Hydraulics provide a hydraulic interface.
- Pump System 1 Hydraulic controls & monitoring;
- Pump System 2 Hydraulics controls & monitoring;
  - Pump System 1 & 2 Hydraulic controls & monitoring is an electrical interface which controls the hydraulic components.
- Pump System 1 motor electronics;
- Pump System 2 motor electronics;
  - Pump System 1 & 2 motor electronics is an electrical interface which controls the pumps.
Attention should also be paid to the inputs and outputs from the overall system boundary. In this case, there is no defined input source for the tank, so the input is denoted in the model as ‘???’ to highlight that as there may be some input conditions to the system that affect the assessment. Output from the overall system boundary is via the drain valve D1.

The platforms themselves are not determined by function but by nature of interface. The tank is the main platform as it is the subject of the control and monitoring system.

The pump system motor electronics are shown as separate platforms as in Figure 4.4.1 they are shown as distinct from the pump system control and monitoring system. In a real application, further explanation would be sought here regarding the reasons why these platforms are indicated as separate items in Figure 4.4.1 as they appear to share electrical interfaces with the pumps in the same manner as the control and monitoring platforms. They may be represented in this way purely for clarity or there may be a different explanation such as these platforms are supplied by a third party. This is an important differentiation and must be acknowledged as such even though the interface is of the same nature as that of the control and monitoring platforms.

There is no technical reason why this differentiation is made; it is merely represented in such a way to determine system accountability. For more complex systems such a differentiation has to be made as such accountabilities will impinge on the party that either supplies or has technical responsibility for that part of the control and monitoring system and potentially the reliability monitoring system. If ownership is established in this way then a contractual agreement must be reached on the responsibility of each party in the event of a system failure.

In the example given, there is redundancy in the electrical control and monitoring systems, there is also a degree of redundancy in the other platforms, those of the pump hydraulic systems, the hydraulic interface of the system. However, in the case of this interface, there is not true redundancy. Both hydraulic systems flow through a joint inlet pipe.

Technically speaking, two different modelling solutions could be applied:
• The joint inlet pipe can be modelled as part of the tank system boundary with each redundant system shown with separate system boundaries, each making its own connection to the joint inlet;

• The redundant hydraulic systems could be portrayed as a single platform instead of two distinct and separate system boundaries with the output of this system boundary (effectively the joint inlet pipe) interfacing with the Tank system boundary.

Both approaches are valid, but the approach that offers most value for assessment purposes is the latter. This is because there is no real description of what form the joint inlet takes, this may be an issue for further debate and therefore should be highlighted. It is accepted that this modelling approach may not show redundancy as clearly as having distinct platforms for each system as with the pump system control and monitoring platform, but it does indicate a potential point of weakness (the joint inlet) in the system design approach to redundancy and may consequently lead to a design change where each system has a separate and dedicated inlet to the tank.

The differentiation of system boundary for redundant systems could also be made based on whether or not the redundant system uses the same sensors to monitor the Hard System under assessment. If both (or more) control and monitoring redundant systems use data from the same sensors then both (or more) redundant systems should be represented in the same platform. If the sensors are duplicated, and are therefore dedicated to the particular redundant system, then this system should be shown as a separate platform in its entirety.

Finally, it is also acknowledged that there is more than one approach that could be used here to model this Hard System. The view could be taken that each system is independent and identical and so only the tank, tank control and one pumping system, its control and monitoring functions could be modelled (as the same assessment applies to both pumping systems). This would provide a much simpler system model for assessment. This approach is valid as long as each system is truly redundant and is stated as such. In the example given above this is not the case for the hydraulic system.
The model as described above including its respective platforms, components and interfaces is shown in Figure 4.4.2.
Figure 4.4.2 Modelled representation of the Multi input tank system
4.4.1.2 Summary guidelines for modelling redundant Hard Systems

The overall aim of the model is to partition the system with respect to interface even in the case of a redundant system.

Inputs and outputs to the model must be defined and if no source or sink is identified it is shown as such. In the example given in Figure 4.4.2 the input is defined as ‘???’ as the source is not identified. The output however is defined and is through the drain valve D1.

The aim of the system partitioning is to make it clear which components having the same interface:

- Are subject to control & monitoring;
- Perform a control & monitoring function;

Partitioning the system in this way provides a view of:

- What components are already involved in control and monitoring, the use of these components may be extended to monitor other components or corroborate their failure;
- How the monitoring components relate to the components being monitored (in the system design, this may not always be obvious);
- The system components interface relationship.

This final point is important as the interface type will potentially influence the methods used to monitor failure or provide corroboration of that failure. For instance increased oil temperature (electrical interface sensor) and oil stream debris detector (hydraulic interface sensor) would respectively indicate and confirm a bearing failure.

This initial partitioning of the system should be performed along these lines. This is how the system in figure 4.4.2 has been partitioned.

It may however be necessary to conglomerate these platforms for redundant Hard Systems if:

1. They have a direct functional relationship;
2. They are provided by a particular supplier.
Using Figure 4.4.2 an example of the direct functional relationship suggested in point 1 is that of the pump system control & monitoring and pump system motor electronics. These platforms have an electrical interface and combining them reduces the number of platforms. There may also be the opportunity to develop a monitoring relationship between the control and monitoring electronics and pump speed.

To illustrate point 2 it is desirable to highlight platforms provided by a particular supplier. These systems may require a different monitoring approach according to in service contractual agreements which differ from that of the rest of the system. For instance a dedicated monitoring system provided by the supplier may be used on the components in these platforms.

Conglomeration of platforms is only acceptable when they share the same physical interface. Figure 4.4.2 shows an exception to this approach with regard to the tank control hydraulics. Here two identical systems are shown in the same platform. They are redundant systems but they share a joint inlet to tank, so they are not strictly separated redundant systems. It is for this reason that they are shown on the same platform but as complete and separate systems.

Finally if there is true system redundancy, there is no need to model both redundant systems, only one system will need to be modelled and a note made of the fact that it has an identical redundant system which mirrors it. This would be done to provide a clearer view of the model. Any design changes resulting from the modelling need to be included in both systems.

4.4.2 Modelling multi lane systems (hot or cold standby Hard Systems)

The same modelling approach is taken here as for redundant systems in section 4.4.1.

In that if a ‘hot’ standby system has two control lanes of, for instance, electrical switchgear, then those two lanes should be shown as a conglomerate on the same platform, whereas if the same system were
redundant on a ‘cold’ standby, the standby system would be shown as a separate system platform.

There is also an option to model either the whole system or just a significant part of the system if the system is symmetrical and identical as long as this is stated to be the case as discussed in Section 4.4.1.2.

The only points of focus in the case of hot or cold standby systems (more so than with redundant systems although the same conditions may apply as discussed in the example shown in section 4.4.1.1) would be those sensors that are shared or the conditions that are used to invoke the standby system. As in section 4.4.1.1, the approach to the model taken should be that which provides the most appropriate view for assessment in the opinion of the analyst.

In general and apart from the specific conditions discussed in this section, the guidelines for redundant Hard Systems outlined in section 4.4.1.2 apply to this type of system also.

### 4.4.3 Modelling distributed Hard Systems

The modelling approach described in the simple example in section 4.4.1.1 can also apply to distributed Hard Systems such as the typical Supervisory Control And Data Acquisition (SCADA) system shown in Figure 4.4.3 which includes Hard System components capable of sensing control parameters and affecting changes to the system based on those changes.

Here the same general guidelines apply in defining Hard System platforms as in previous examples in this chapter and as outlined in section 4.4.1.2. In the case of the system shown in Figure 4.4.3 however there is a difference in the hierarchical structure of the system and the way it is controlled.

In Figure 4.4.3 unlike the other systems considered in this chapter, the highest authority control functions are performed using computer networks (High level network, Mid level network). The modelling approach suggested by this research does not extend to the software based control schemes typified by these networks.
A model of this type of system could only be made if the system boundary starts from the central plc controller and ends at the motor/sensor level. This is because software protocols drive the High and Mid level networks and there is no provision in the methodology to assess their effect on reliability. In other words it is more difficult to assess the monitoring strategy provided by these networks as software control may call for a number of parallel monitoring solutions depending upon the state of the components in the distributed system. That is that if one particular monitoring component is disabled at one point in the network, various components from around the network as a whole may be used to compensate for this failure.

It is suggested therefore that the methodology suggested by this research is not suitable for assessment of this type of monitoring system and if it is applied to these types of system the assessment must be qualified in its scope and extent.

By extension, the methodology is therefore less likely to be suitable for other types of distributed systems (such as those distributed systems which consist entirely of computer networks).
Figure 4.4.3 An example of a SCADA system or typical distributed Hard System
4.4.4 Modelling larger, more complex Hard and Soft Systems

As can be seen from all the examples modelled so far and especially when redundancy and multi lane systems are introduced for Hard Systems, the complexity of the model increases both in terms of relationships between system boundaries and model size.

The same is also true for Soft Systems, here, the model becomes more complex when the number of departmental, company, cross departmental and cross company interactions involved in the Soft System increases.

These interactions are also likely to exist at various levels in each organisation, not only task performers, but also task managers will necessarily need to be included in the overall Soft System model. This implies that the methodology will have to be able to take into account an increase in the ‘breadth’ of the model (capable of dealing with increased numbers of interactions) but also an increase in the ‘depth’ of the model (capable of dealing with increased number of levels of interaction, from manager, to supervisor to subordinate) from the simple examples that have been modelled in previous chapters.

This situation is similar when dealing with Hard Systems. It is possible that some systems under assessment will have more than just one ‘level of abstraction’ and a number of ‘levels of abstraction’ will have to be examined to ensure the reliability monitoring system is ‘fit for purpose’. It is also possible that the system under assessment will have its control authority subject to a higher level of abstraction for safety reasons.

An illustrative example would be that of an aircraft and an aircraft engine. In this case there is a responsibility of the engine manufacturer to ensure the safe operation of the engine in its own right, but any engine related safety issues that may cause aircraft operation to be unsafe must be dealt with at an aircraft level. Such a case can also be used to illustrate the complex nature of control (and safety) hierarchy.

Primary safety and control features of modern aircraft engines would be performed where possible by the use of electronic control, but there are safety
issues that require mechanical safety features which override those of electronic control. Also there are aircraft level safety features to consider. All of which leads to a complicated view of system authority across these specific system boundaries.

This situation also extends to reliability issues. These boundaries and their dependencies need to be modelled in order to be able to carry out not only a holistic assessment of the Hard System, but also that of the system as a whole which includes the Soft System as it is often the main system boundary (the aircraft in this case) that is the main means for transmission of reliability data to the Soft System for analysis. If all those aircraft subsystems that require data in order to monitor their performance do not get the data necessary to perform that analysis in a timely manner according to the nature of the reliability fault they are intended to detect, the reliability monitoring system and its supporting business case is prejudiced.

The notation developed so far is capable of modelling such complex systems but as the systems (both Hard and Soft) get larger and more complex it is more difficult to accurately identify the main dependencies that are important to reliability assessment in an appropriate relational context and the models themselves because of their size become confusing and difficult to read.

The options therefore are either to model at a more coarse level of abstraction (say subsystem to subsystem level or department to department level) or to continue to model at an appropriate level and group those models in such a way to reflect their hierarchy.

The first approach would lead to a less convincing assessment and may not identify important reliability issues. Therefore, the second approach is one that needs to be developed further.

Ideally, if the system can be represented on a single sheet of paper, subsystems and components retain their context and ease analysis. This should be the ultimate aim of the model, to ensure that the specific tasks or components of interest for a specific part of the model under analysis can be identified by a modelling segment and then assessed in relation to all other components or tasks with which they interface. A Hard System analogy
would be to have the ability to take a complex system such as an aircraft and be able to identify a particular sub-system of that aircraft and assess all components that relate to that sub-system.

It is also necessary for this research to be able to provide a modelling perspective for the type of system which is designed and developed by more than one party. This type of system is becoming increasingly common as systems development is being performed by revenue sharing partners who have responsibility for one subsystem which contributes to the system as a whole.

Apportioning reliability targets amongst subsystem suppliers are decisions which have far reaching engineering and commercial consequences when analysing systems which are designed to be implemented on a service contract basis.

A need for a method of representation which is convenient for both describing the system in context and which is in sympathy with the notation that has already been developed and described above is therefore required.

4.4.4.1 The Role Context Chart concept

Role Context Charts were introduced for Soft System modelling by research performed by Murdoch et al [44]. These charts were used to address a perceived deficiency in the Role Activity Diagram business process modelling notation developed by Ould [25].

The perceived deficiency these charts overcame was that the Role Activity Diagram business process modelling notation used was too detailed and was carried out at too low a level of abstraction for the user to gain a contextual view of how the business processes being reviewed tied in to those of the department as a whole. The Role Activity Diagrams also became too unwieldy in describing the processes which made up the Soft System under examination as some processes were so large that they could only be described using several sheets of paper. There was no way of connecting these charts at the time this research was conducted, and so this method was put forward as a solution to the problem.
This method therefore provides some indication of how one ‘role’ interacts with another, but without providing the detail of that interaction (which essentially is the cause of the unwieldy nature of the Role Activity Diagram modelling notation).

The Role Context Chart overview is developed using Pugh’s Total Design Model [48] shown in Figure 4.4.4 as its basis. Pugh suggests that the design and manufacturing process is akin to a stack of ‘coins’ with each ‘coin’ forming part of the process life cycle. Hence one coin might be ‘marketing’, another ‘manufacturing’. For process to progress, each ‘coin’ or process stage must interact with each other ‘coin’ in the stack immediately above it and immediately below it. It is further recognised that other ‘stacks’ (other companies, or perhaps even departments) will need to interact with this ‘stack’ but only at appropriate levels.
This approach allows the modeller to compartmentalise the Soft System into various stage levels which recognise both precedence and dependence of various roles in the Soft System and their associated tasks at particular points in time.

This has the benefit of partitioning the Soft System into more manageable portions for assessment whilst retaining the relational integrity between those portions.

It is suggested that the same approach can be made in Hard System terms. The demarcation point for the Soft System ‘coin stack’ is one of time (a development phase period) but for Hard Systems the demarcation points would be that of control authority which in safety terms naturally equate to
control systems risk levels. That is, the system boundary which exerts executive control authority also has to embody mitigation at the highest safety risk level as discussed earlier in section 4.4.4 in the example of the aircraft system.

The Soft System approach adopted here is conceptually the same as that proposed but not formally developed by Murdoch et al [45].

Here each Soft System boundary becomes a ‘stack’ as shown in Pugh’s Design model above. The ‘stack’ effectively describes the impact of each of the Soft System boundaries with respect to time.

Some Soft System boundaries will be short lived in comparison with others, some will be present at each level of the ‘stack’ and others will only be present at particular points in the Soft System and so may only appear at one point in the ‘stack’.

The ‘stack’ in Figure 4.4.4 is used as a purely conceptual notion as a 3 dimensional object and is never described by the notation as it is shown here. Rather a view looking down upon the stack (as if the stack was viewed in the same way as the direction shown by the ‘Core phases’ arrow in Figure 4.4.4) is drawn which illustrates these individual layers and the relationships of each Soft System boundary at that point.

All of the main Soft System boundaries are drawn as circles (reflecting the top most view of the stack) and all of those Soft System Boundaries which have a relationship with each other have interaction lines drawn between them. Figure 4.4.5 is derived from Murdoch’s research [45] and illustrates this concept.

Once the defined tasks belonging to each Soft System boundary completes within each defined Soft System boundary for a particular level of the process (or ‘coin’ in the stack to maintain the analogy with Pugh’s design model) this stage of the process can be said to be completed and the Soft System drops to the next ‘coin’ in the stack so that the process can be carried out in the Soft System boundaries ‘coin’ that are associated with it at that point in the process. These Soft System levels (or coins) are notionally bounded by time
in the terms of this research, but they could also be bounded by other constraints, for example decision review points or availability of key data.

Figure 4.4.5 Role Context Chart - the notation proposed by Murdoch et al

This approach provides an opportunity for the modeller to outline what is thought to be the necessary organisational structure of the process without laying any detail on the structure (the detail would be defined by the Role Activity Diagram business process modelling notation referred to earlier).
The Role Context Chart therefore allows a conceptual view of the entirety of the Soft System in a context which not only identifies the main Soft System boundaries, but also is partitioned according to particular stages of the Soft System at cogent points that are particular and inherent in that Soft System and that only the particular Soft System boundaries and their relationships are identified at each of those stage points. This method therefore identifies specific relationships at specific times and the specific relationship at that point in the Soft System enabling targeted and realistic assessment.

In terms of aiding human understanding of complex systems, the Role Context Chart also shares characteristics with Buzan’s ‘mind map’ [49] approach, in that all roles to be included in the Soft System are identified (but not necessarily defined exactly – that is left to the Role Activity Diagram notation) immediately and then developed as required. This is particularly useful when developing new processes.

4.4.4.2 The Soft System Context Chart. A redefinition of the Role Context Chart concept

The research performed by Murdoch et al [44 & 45] demonstrated how these charts could be used in a process modelling context but it did not formally propose a modelling notation for complex Soft Systems, it only described in conceptual terms the approach that could be taken. The following builds on work that was discussed at the time of that research and adds to it in the light of this research.

There are three principle notation devices, they are:

- The main ‘Soft System boundary’ is defined as a grey box (compare with the grey circles labelled ‘A’ and ‘B’ in Figure 4.4.5);
- Specific functions of the Soft System boundary are shown by the white boxes (labelled ‘a1’ & ‘b1’ in Figure 4.4.5) in apparent ‘orbit’ of the grey circles;
- Indication of Soft System boundary function interaction is made by the lines that run between the white boxes. Soft System boundary to Soft System boundary (grey circle to grey circle) interaction lines are
disregarded in the notation as they are considered superfluous in this research.

The main idea here is to provide a view of which Soft System boundaries are participating in the Soft System as a whole. There is also the need to provide an acknowledgement of which functions of each Soft System boundary need to work together to deliver the Soft System (indicated by the interaction lines) the exact nature of that interaction is intended to be shown by the modified Role Activity Diagrams. These ideas are demonstrated below in Figure 4.4.6 using the same model as Figure 4.4.5.

Figure 4.4.6 Soft System Context Chart – showing the revised notation concept of the Role Context Chart applied to the model shown in Figure 4.4.5
Figure 4.4.6 above shows the key differences in the notation when compared to the conceptual notation described in Figure 4.4.5. They are that the core process and Roles become ‘Soft System boundaries’. In the original proposition, the core process described activities that were implicitly part of the Roles’ responsibilities (so called in the terms of that research and Soft System boundary functions with respect to this research) which had no interaction with other Roles, they were therefore left unspecified.

In this research all functions of the Soft System boundary are stated, whether there are interactions with other Soft System boundaries or not. This clarifies the Soft System purpose and does not add to the complexity of the chart unduly.

Specialist functions become Soft System boundary functions, there is little difference between how these functions are used between the original and the developed notation, except that they now become an explicit description of the function of the Soft System boundary. The one main difference is that in the Murdoch notation (Figure 4.4.5) it was suggested that each specialist function was independent of the others in the Soft System boundary ‘orbit’. Whilst this is still the case in the developed notation, the position is taken that each Soft System boundary function is implicitly ‘aware’ of the status of the other Soft System boundary functions in the orbit. This view is taken, as to do otherwise would require inter Soft System boundary descriptions of how information is passed between Soft System boundary functions, which would complicate the process at this highest level of abstraction.

Finally, the other main difference between the original Role Context Charts and developed Soft System Context Chart notation is that interactions between Soft System boundaries can only occur between Soft System boundary functions. This was allowed in the original notation, but in addition, interactions between core processes or ‘Roles’ were also allowed. This ‘Role to Role interaction’ was found to be confusing during this research as the ‘Role Context Chart’ did not explicitly show what parts of the Soft System in particular the ‘Roles’ were interacting with, just that there was a perceived and unspecified linkage of some kind. This lack of definition would not be helpful to any reliability assessment.
4.4.4.3 Defining the Soft Systems boundary level limits of the model
As discussed, it is important to remember that Soft System boundaries could both appear and disappear at various stages in the Soft System as a whole and can also change their nature to potentially play different roles in the Soft System throughout the duration of the process. For example it may be that authority to begin and end a process will come from a senior manager. That senior manager once the process is started may not take any further part in the process until all other roles have completed their tasks and the senior manager is then required to return and review and approve what has been produced. Therefore the senior manager will have a part to play in the first part of the process (or the topmost ‘coin’) and then will not play a part in any of the intermediary process ‘coins’ until the bottommost coin is reached where the senior manager role will ‘reappear.’

This means that as the process is enacted, at various points the view offered by the Soft System Context Chart will change but having compartmentalised the charts in this way means that the process is much more easily assessed without losing context at any point in time.

Also, if for example, the complex Soft System is being developed ‘top down’ as a new business model, the Soft System boundary functions are effectively subsets of the Soft System boundary defined as functional blocks (in the same way as Buzan’s mind map is used to further define what is meant by principle points – or ‘system boundaries’ in this case) but not developed further until the whole Soft System Context Chart is defined and modelled at which point lower level modelling using Role Activity Diagrams can commence.

There may also be the case that different parts of the Soft System provide supportive rather than direct contribution to the success of the Soft System function (for example a finance department may support an engineering function) such a function may be a relatively unimportant function of the finance department but of vital importance to the engineering Soft System function delivery. Discovering and documenting these apparent ‘incidental’ functions allows for further development of the Soft System model and tightening of boundaries and delivery, but such an occurrence may not be
considered when the financial departments own processes (or Soft System) are developed for their own use.

The aim of showing the specific Soft System boundary functions provides clarity of the Soft System boundary purpose and also provides information on the specific accountability of that Soft System boundary, that is, an indication of what are the general responsibilities of that Soft System boundary.

When developing new business models, it is possible that two different role holders may find that, historically, each has part control of a particular function or sub-function of the new business model Soft System. Ownership of the function or sub-function for the new business model must be resolved and defined in order to achieve quicker and more reliable response times. This model identifies such areas of contention at a high level of abstraction such that they can either be resolved in principle at this level or when more detailed modelling takes place.

In addition, it must be remembered that, as in Pugh’s Total Design Model, the complex Soft System is best represented in a three dimensional sense where the partitioned layers be constrained at points which will be decided primarily by the nature of the complex Soft System being modelled.

This view correlates with that of the so-called ‘V’ – process [50] a typical model of which is shown in Figure 4.4.7. This view shows, as an example, the design process for a system and is a well known construct in Systems Engineering.

The figure shows the process for developing a system and is read by following the directional arrow on the ‘V’. The process starts from the ‘customer or user perspective’ partition level at the top and leftmost corner of the ‘V’ and continues down the ‘V’ and up the other side to end at the Customer level again.

This demonstrates that Soft System boundaries defined at the beginning of any process (in this case – the customer or user perspective) will also play a part in the ‘operation’ at the end of the process when the product has been developed. Although the Customer Soft System boundary operates at the same partition level of Systems abstraction when performing both of these
functions at either end of the design process, both roles are very different from each other. At the start of the process, the customer decides the important design features that must be met and specifies these and at the end of the process the customer ensures that these requirements have been satisfactorily met.

The purpose of the functions that lie within particular Soft System boundaries therefore change demonstrably over time or with function and some means of showing what actual function of the Soft System boundary is at a particular point in time in the Soft System process as a whole is required.

It is acknowledged that this is an idealistic model and the customer may be involved at lower levels of abstraction in some capacity even purely as an occasional reviewer of progress.

Figure 4.4.7 The V-Process model

In notation terms, there is no difference in how this change in Soft System boundary function is described, but obviously some differentiation is required.
This is achieved by drawing Soft System boundary functions of the process at particular and defined stages of its cycle or to use the earlier analogy, ‘coin layer’. This has the added benefit of providing a less complicated picture of the process as a whole and also shows which Soft System boundary and what functions of those Soft System boundaries are active at any particular stage of the Soft System process.

This idea also aligns with the current practise of performing phase reviews of product design after a series of set functions have been performed. The idea is that the design will not progress until suitable progress has been made in all functions and therefore are allowed to pass through a ‘gate’. This is commonly known as a ‘gated’ design review process, in effect; each ‘gate’ will allow the process to continue to the next ‘coin layer’.

An example might be of that used in some maintenance organisations where there are three lines of maintenance:

The first line would be defined by the Soft System boundary that has tasks which are close to the equipment being monitored and who can:

- Make repairs that do not cause a loss of availability (or a loss of availability within defined contractual limits);
- Make repairs on easily exchangeable unit items (for example electronic control boxes);
- Make repairs without the need of specialist or heavy lifting equipment.

The second line would be specialist and would deal with equipment that had either been removed for specialist testing or would use specialist equipment to diagnose reliability issues of the equipment being monitored in situ.

Finally the third line would be the specialists who have access to more specialist diagnostic equipment and who could strip part or all of the equipment being monitored if necessary to perform a full diagnosis.

It can be seen from this example that there are lines of communication amongst all of the roles involved from the first line to the third that will need to be developed and maintained in order to provide an efficient service. So
there is a need to define the Soft System as a whole, even though this may be an extremely large Soft System model.

It can also be seen that if the issue is out of the scope of the first line Soft System boundary then the Soft System must change the Soft System boundary function of the first line to rapidly accelerate the issue to whichever of the other lines of maintenance is appropriate and provide diagnostic and operational data which characterises the issue. The nature of the first line Soft System boundary will need to be reflected differently at each succeeding layer.

The first line Soft System boundary function at the first line 'coin' level is independent of the 'coin' levels and self contained and may only send details of reliability incidents for record to the other levels for management purposes.

If the reliability issue becomes a second or third line issue, the first line Soft System boundary function changes to a support function at the second and third level, providing information to aid with the diagnosis.

These differences in function at different levels of abstraction must be modelled but are difficult to model in a large and complex Soft System, providing the opportunity to define strict levels of abstraction offers the opportunity to:

- Reduce the complexity of the model;
- Clarify the models purpose;
- Provide clarity for each Soft System boundary definition.

Over either a condition based change in Soft System boundary (as demonstrated above) or if necessary on a time period basis (the change from first line to second lines maintenance takes place if the issue cannot be resolved within a set time frame).

An 'unreliable' Soft System of this nature; or a Soft System that is not defined to perform this function will adversely impact the reliability of the business model regardless of the efficiency of the Hard System monitoring.
4.4.5 Extending the Soft System Context Chart model notation to Hard Systems

The description of the complex system modelling notation so far has been concentrated on the development and application of the notation to Soft Systems. One of the primary motives for this research is to develop a modelling notation that is suitable for both Hard and Soft Systems in order that weaknesses in reliability are equally obvious in either system.

Hard System Context Charts will be used to define the interactions between platforms as these may also cross design responsibility and contractual responsibility boundaries. If they do so, there is a potential weakness in safety and reliability terms if the interface is poorly defined. Identifying such weaknesses in a timely manner will impact both design and contractual arrangements if agreement on such issues cannot be resolved. If identified issues have to be resolved on a compromise basis, this also has the potential to compromise the proposed business case for the service.

As with less complex models developing a notation to model complex Hard Systems, like complex Soft Systems, requires that the model provides an accurate representation of the system without sacrificing either clarity or necessary detail.

There should also be a similarity between what is modelled in a complex Soft System and what is modelled in a complex Hard System in order that there is consistency in modelling the particular properties of either system which most affect reliability.

Although the assessment methods may differ, due to differences in the nature of each system, the primary points of reliability failure are suggested to be essentially the same for both system types although their importance in assessment terms will vary according to the nature of the system. For example, weaknesses in boundary to boundary interactions in Soft Systems can be equated to weaknesses in Hard System interfaces. In the case of a Soft System, such a weakness is a major contributor to unreliability whilst for a Hard System weaknesses in the interface, for example a leak in a pipe between two hydraulic components is potentially less critical.
It has to be remembered though that the sense of the Hard Systems approach is completely different to that used by modelling Soft Systems as what is being modelled will be, by its nature, more integrated, and will also demonstrate features of systems that are not usually found in Soft Systems – such as a definite system functional partitioning. That is functional independence is more often than not developed and strictly defined within Hard Systems platforms. In Soft Systems, such a definition is not the prime consideration of the Soft System designer either because when the Soft System was designed such integration between Soft System participants was not envisaged or because the new business model enforces such integration. It could be said that Soft System design is as much derived from organisational or political motives in complex organisations as it is for the efficient delivery of a Soft System deliverable.

The primary aim of modelling complex Hard Systems allows for an ‘overview’ of the extent of the Hard System to be assessed. There is by no means an assertion here that the method proposed is the best way of modelling a Hard System. The prime purpose for the notation described is to provide a representative overview of those parts of the system that will be under analysis.

In Soft System terms the ‘stack of coins’ analogy implied by Pugh’s design methodology represent stage of process development. The process moves through each ‘coin’ in turn as time and the process progresses.

A different approach is taken when considering Hard Systems. With this type of approach each ‘coin’ should be thought of as a major system which contains subsystems and forms a Hard System platform.

These platforms are directly related with the ‘lowest’ platform or bottom most ‘coin’ being not only a system of itself but also a subsystem of the highest platform or top most ‘coin’. The relationship between these systems is a strictly functional one with the ‘highest’ platform assuming ultimate control over its lower subsystems. This level of control also has safety and reliability implications as authority for the safety of the system as a whole resides with the ‘highest’ platform.
This approach is not intended to be used with networked ‘system of system’ Hard Systems unless there is an equivalent level of authority with the ‘highest’ platform. In practice this is unlikely to be the case as networked ‘system of systems’ are primarily bought together to solve specific mission objectives or to perform a specific operation rather than a functional objective.

An example of a Hard System with a functional objective using this approach would be an Aircraft system. The Aircraft would be the top most ‘coin’. Below that would be a subsystem of the aircraft, for example, the Engine.

What should be noted here is that unlike Soft Systems, functional authority would not be passed from one level to the other. Functional authority and control will always relate to the top most Hard System platform in general and in particular with safety critical systems.

It should also be noted that unlike Soft Systems (where functions are many and varied and not often shared by all development stages) functional responsibility for Hard Systems will more often than not share some level of functional responsibility across all levels. For example there will be an Aircraft air system Hard System platform function that will interface with air systems and air system controllers Hard System platform functions at the Engine level.

A further distinction is that in Soft Systems, the interfaces between system boundary functions shown may be of a varied nature which may have different and distinct attributes between various Soft System boundary and Soft System boundary functions. This is not the case when modelling a Hard System. In this case the interfaces have a defined commonality in that they have similar attributes in addition to a common interface between Hard System platform functions. This difference in itself does not have an effect on the analysis but it should be recognised.

In other aspects, the function of the notation is broadly similar. In a Soft System, Soft System boundaries are principals each with their own defined area of responsibility. Similarly, in a Hard System, major systems are recognised as principals that are comprised of subordinate functions. It naturally follows that those subordinate functions mentioned can be viewed to be similar in purpose.
It should be noted however that these System boundaries and System boundary functions are defined subjectively by the analyst according to the assessment required. This position is adopted as when modelling any system with the view to performing an analysis, the system perspective the analyst takes directly affects the analysis in terms of scope and detail. This arrangement means that although it is important to recognise that there is a hierarchy between Hard Systems platforms there is no need to make any further distinction regarding the level of hierarchy in the notation, as it is should be obvious from the platform names which is at the highest level of abstraction and from platform functions which is the ‘controlling’ function.

4.4.6 An example to demonstrate the extended modelling notation set used to produce the Soft Systems Context Chart

To demonstrate the principles discussed in section 4.4.4 with regard to larger more complex Soft Systems modelling, and to define the extended modelling notation set used to model the Soft Systems Context Chart consider the following example Soft System description.

A reliability monitoring system has been developed for an aircraft. This system provides data which monitors the health of the aircraft in reliability terms but cannot process all the data that monitors health on board the aircraft in real time. Some of the data processing and alerting of failure has to be performed at a ground station. Further, the data that is provided by the aircraft also contains data regarding the health of the aircraft’s engines.

The health monitoring data belongs in the first instance to the aircraft operator but the airframer also requires access to the data to monitor the health of the airframe and so does the engine manufacturer who has an agreement with the operator to maintain the availability of the engines on board the aircraft to optimise aircraft usage.

The aim of the example would be to develop and model a Soft System by which the engine manufacturer receives and processes data to provide a monitoring service which optimises engine availability in order that a reliability assessment can be made to support the business case of the monitoring service for the engine manufacturer.
In practice, the same model could be used to optimise service delivery both for the airframer and for the aircraft operator. The model would be developed in the same way and would only differ in terms of perspective and level of detail according to the end users requirements.

If in the first instance, it is considered that this is a new business model, the first thing that must be done is to consider the main ‘Soft System boundaries’ of the proposed Soft System.

4.4.6.1 Soft System boundary definition
As when modelling the simple example in the previous chapter, the Soft System boundary is defined by the person, organisation or department accountable for various functions in that Soft System. It can also be a system which has the function of producing data automatically.

It is assumed that either the person or people or systems within the Soft System boundary have the requisite capability and capacity to enable the functions with the Soft System boundary to perform its required function. The assessment of lack of capacity or capability to perform this function optimally is a natural consequence of the reliability assessment that will be performed.

The important consideration here is that the Soft System boundary has responsibility for the functions assigned to them and has the ultimate responsibility for ensuring that they are carried out as required.

In our example, there are clear Soft System boundaries, they are defined by:

- The Aircraft operator;
- The Airframer;
- The Engine Manufacturer;
- The System that produces the data to be operated on.

As this is a new business model, there may be more Soft System boundaries as yet unidentified or the definition of Soft System boundaries may change (for example there may be a specific department or departments in the engine manufacturer’s organisation that may be responsible for dealing with the customer). Such considerations are not important at this stage of modelling, but will need to be clarified before an assessment takes place.
The notation used in the Soft System Context chart for a Soft Systems boundary is the grey box as shown in Figure 4.4.8.

![Airframer](Image)

**Figure 4.4.8 Showing the Soft System Context Chart notation for the Airframer Role**

All of the other main Soft system boundaries listed earlier in this section would have similar grey boxes to represent them in the model.

4.4.6.2 **Soft System boundary function definition**

These are the prime responsibilities or functions of the Soft System boundary. In effect, they amount to the job description or specification of the person who has responsibility for the Soft System boundary. This definition holds good for data generating systems also.

Each of the main Soft System boundaries listed in the example above must then be defined further as to their purpose in the Soft System, as this is an illustrative description of the notation, no actual Soft System example description is given, it is hoped that the notation makes clear the purpose of each function.

As an example, the Airframer Soft System boundary will have a Soft System boundary function which, as discussed earlier, will receive data from the aircraft to process for their own reliability tracking needs.

The notation used in the Soft System Context Chart for a Soft Systems boundary function is the white box as shown in Figure 4.4.9.
Figure 4.4.9 Showing the Soft System Context Chart notation for the Airframer Soft System boundary function named ‘Airframer Product Support’

Note in this instance although the Airframer Soft System boundary function is to receive data from the aircraft to process for reliability tracking, a departmental name is used to define the Soft System boundary function. This could be the case if the function is already well documented and will stay the same or if this were not the case, the Soft System boundary function could be written in full in the box, that is ‘to receive data from the aircraft to process for reliability tracking.’

The same added definition of Soft System boundary function would occur for all of the other Soft System boundaries as the model builds. In order to ensure that each function’s ownership is clear in the model that function is placed close to the Soft System boundary to which it belongs and a dotted line is drawn around the Soft System boundary linking all of its respective functions as shown in Figure 4.4.10.

Figure 4.4.10 showing the Soft System Context Chart notation for Soft Systems boundary and Soft System boundary function ownership
4.4.6.3 Interface – how the Soft System boundary functions interact

It is both important and necessary for developing the lower level Soft System definition provided by the modified Role Activity Diagram notation and in terms of understanding the context of the Soft System boundaries for assessment purposes to define which Soft System boundaries share interactions with each other. The actual nature of that interaction is not strictly defined using this notation, rather an acknowledgement is made that an interaction exists. The actual nature of the interface is defined by the modelling phase that uses the modified Role Activity Diagram notation.

Such a representation of Soft System boundary function interaction is required in order to:

- Ensure that all interactions are ‘captured’;
- Ensure that all Soft System boundary functions acknowledge and recognise the need for that interaction;
- Form the contextual basis for the lower level more detailed modelling provided by the modified Role Activity Diagram notation.

It can be assumed in our example that the source of reliability data used by the Airframer will be provided by the maintenance organisation of the Airline (as they nominally ‘own’ that data). This interaction is shown by a full line between the two Soft System boundary functions responsible for that exchange of data. As shown in Figure 4.4.11. This notation does not and cannot imply process flow direction. This is because the interface between the Soft System boundary function is of a ‘compound’ nature, that is, it may involve a number of types of different interaction when modelled in detail, and therefore will also comprise process flow to and from each function in equal measure.

The direction of process flow therefore is described only when ‘modelling’ using the modified Role Activity Diagram notation.
Figure 4.4.11 Showing the Soft System Context Chart notation for an interaction between two Soft System boundary Functions
Note that the Airline has other Soft System boundary functions identified in the Soft System which as yet have no other interactions shown. These could either be developed further as modelling continues if an interaction with another Soft System boundary in the Soft System occurs, could be disregarded until a lower level of abstraction in the Soft System process has need to use them, or if no such link develops can be discounted and removed from the model.

The function of model building using the Soft System Context Chart is to clarify and contextualise the purpose of each Soft System boundary at a very top level in the complex Soft System and by doing so reduce the complexity (or perceived complexity of that model.)

4.4.6.4 Soft System Context Chart for the complex Soft System example

The Soft System Context Chart for the complex Soft System example is shown in Figure 4.4.12.
Figure 4.4.12 Soft System Context Chart for the complex Soft System example
The Soft System boundaries are defined as:

- Airframe;
- Airline;
- Airframer;
- Engine manufacturer (engine service provider and project failure response).

These Soft System boundaries have been decided upon based on the information provided and are developed to show in practice how these roles would be defined.

In the example given a Soft System boundary for the system which produces the data was called for. In practice this role would be split between the ‘airframe’ and the ‘engine manufacturer’ as both have functions which would constitute that Soft System boundary. In the case of the ‘airframe’, a sub-system would be required which would provide data collection and transmission.

In Soft System reliability terms, we are interested in this system not as a physical system in itself, but its operational parameters in terms of when it delivers data, how often it delivers the data and whether that data is complete and fit for purpose. As this is the source data delivery which starts the whole Soft System process, these are important questions. The ‘engine manufacturer’ also has a pertinent Soft System boundary function here as it forms the other end of that data transmission path and the same questions apply to this part of the system regarding the capability of that function in being able to operate on that data (from multiple aircraft sources) in a timely fashion.

In summary, this Soft System boundary function of ‘system which produces the data’ is effectively handled by the Soft System boundary functions ‘airframe – engine health monitor’ and ‘engine service provider – failure prediction.’

The ‘engine manufacturer’ Soft System boundary is split into ‘engine service provider’ and ‘project failure response.’ This is because we are assumed to be
performing the reliability analysis on the part of the ‘engine manufacturer’ and so the definition of those Soft System boundaries will be more defined than the others.

The second Soft System boundary of the ‘engine manufacturer’, ‘project failure response’ is intended to have a different emphasis to that of the ‘engine service provider’ in that whereas the role of the ‘engine service provider’ is to examine in service data it is the role of the ‘project failure response’ to act as a technical expert and interface to support both the customer ‘airline’ and ‘airframer’ with whom they may have to collaborate to solve customer problems.

Of the functions of the Soft System boundaries shown, some are connected and others are not. The Soft System boundary functions that are not interfaced are shown as they may take some part in the lower level model yet to be defined. If the lower level modelling does not require these functions, then they can be deleted from the Soft System Context Chart once the lower level modelling is complete, but it is just as well in the initial stages to note the functions as a reminder of the capability of the respective Soft System boundary as a whole.

In the example given, no information on the Soft System boundary functions was provided these can be added as required and in consultation with the ‘owners of’ or ‘those with responsibility for’ the other Soft System boundaries that are part of the Soft System. This negotiation and further definition process of the Soft System is a useful function of modelling activity as it enables all owners of Soft System boundaries to understand the part they must play in the Soft System as a whole.

This Soft System Context Chart is considered to show all of the Soft System boundaries and Soft System boundary functions that would need to be involved when responding to an Airframe failure event. It is possible that not all of the Soft System boundaries would have tasks relevant to that purpose involved all of the time or that they may be involved in a response in a number of different scenarios. The exact manner of Soft System boundary function and interfacing
is dependent on lower level modelling performed using the modified Role Activity
Diagram notation.

If the lower level detailed modelling activity develops interfaces or an
understanding of new interactions, these must be reflected back and force a
respective change to the Soft System Context Chart.

In order to perform this lower level modelling activity, the Soft System Context
Chart is used as a basis for determining what should be modelled. In the first
instance, the lines of interaction give a guide to what should be modelled at the
lower level as these outline the interfaces that have to be developed to produce
the lower level model.

If we say for example that we wish to model the response of the Soft System to a
known failure condition that is understood and documented and has a detector
for the condition embedded in the monitoring system, we can suggest that initially
the principle Soft System boundaries engaged in this activity scenario would be:

- Airframe;
- Engine Service Provider;
- Project Failure Response;
- Airline.

The Soft System boundary functions involved can be highlighted in the Soft
System Context Chart to show which interactions should be modelled to describe
this activity (the ‘thicker’ lines highlight the interfaces in Figure 4.4.13)
Figure 4.4.13 Soft System Context Chart highlighting the interfaces appropriate for more detailed modelling of a response to a known failure event
These Soft System boundary functions then become the foundation of the modified Role Activity Diagram. Each boundary function becomes a grey box as in Figure 4.4.14. The tasks associated with each boundary function are then placed in the boxes and interfaces between each of the boundary functions are detailed following the method shown in section 3.3. If new boundary functions are found to be needed, they are added at this stage. If any are added, they must also be reflected on the Soft System Context Chart for completeness. It is this lower level Soft System mapping shown in Figure 4.4.14 on which system reliability is formally assessed.

The Soft System Context Chart ensures not only that all of the Soft System boundaries are aware of each others function and their relative importance in the Soft System (for example if the airframe cannot provide data, the Soft System as a whole is compromised) but also an idea of the measure of ‘completeness’ of the Soft System. For example have all the Soft System boundaries that define the Soft System been identified? If not, there is an opportunity to develop these boundaries. By the same token once lower level modelling has been completed by the modified Role Activity Diagrams, Soft System boundaries can be removed or remodelled to suit the overall purpose of the activities that the Soft System is required to support.
Figure 4.4.14 Detailed Soft System definition of the Soft System boundary functions highlighted in the Soft System Context Chart in Figure 4.4.13
4.4.6.5 Summary guidelines for modelling large complex Soft Systems

There is a need for the modelling notation to be able to deal with larger and more complex Soft System models than those provided in chapter 3. As discussed in section 4.4.4, the model becomes more complex when the number of departmental, company, cross departmental and cross company interactions involved in the Soft System increases.

Such models can become very large, it is vital therefore in this case that a contextual perspective is maintained in order to provide a framework for more detailed modelling. This is the purpose of the Soft Systems Context Chart.

The first step in developing the Soft Systems Context Chart is to determine the Soft System boundary definitions. These could be either department names or company names, but must be at the highest level of abstraction of the system under investigation. In the example given in Figure 4.4.13 Soft System boundaries are ‘Airline’ and ‘Engine Service Provider’. It is possible that not all Soft System boundaries are defined in the first instance. This is acceptable as the aim of the model as a whole is to engender collaborative activity in defining the system. Those parties (companies or departments) not involved in the early stages may be identified as needing to be involved by this process and should be bought in as modelling progresses and becomes more detailed.

The process life cycle should also be considered at this point. If the complete process life cycle is modelled it may produce a very large or complicated Soft Systems Context Chart. It is desirable to provide a very clear Soft System Context Chart in order to simplify the next steps of the modelling process.

In such a case, consideration should be given to temporal partitioning of the Soft Systems Context Chart. The result will be a number of Soft Systems Context Charts which represent defined time slices points in the process life cycle. The start and end points of these time slices are governed typically by decision points. For example, a work process may take place, but before its result or deliverable is shared with other members in the process a management team may need to review the deliverable before the process can go to the next stage.
Such a point is considered to be ideal as a temporal boundary for a Soft Systems Context Chart.

The next step is to define Soft System boundary functions. These functions are important both to the higher level model of the system which is the Soft System Context Chart and the lower level model used to produce the Soft System Reliability Risk View as they form the connection point between the two models. These functions are the primary functions that are to be modelled, the Soft System boundary that they belong to infers ownership or responsibility of that function. As with Soft System boundaries, these functions may not exist when modelling begins or may exist with ownership disputed. Assigning functions to a Soft System boundary brings clarity to these issues. There is no limit to the number of functions applied to each Soft System boundary deciding which sensibly belongs where, is part of the modelling process.

The final step in developing the Soft System Context Chart is to determine the interfaces between the Soft System boundary functions. For the Soft System Context Chart no greater interface definition is required other than an acknowledgement that there is an interface between Soft System boundary functions common to one or more Soft System boundaries. More detailed modelling of the interface is performed in the lower level model which goes to form the Soft System Reliability Risk View. Details of how to perform interface modelling are given in examples in chapters 2 and 3.

Once the Soft System Context Chart is developed, then lower level modelling can proceed.

The aim of lower level modelling is to develop the Soft System Reliability Risk View. The Soft System Context Chart provides a guide on how this should be developed through the interfaces shown on it. Only Soft System boundary functions that are connected through a shared interface can be modelled at a lower level.

The Soft System boundary functions of interest and all those functions that they are interfaced to then become the principle Soft System boundaries of the lower
level model. This lower level modelling activity follows the same procedure as discussed in chapter 2. The aim is to develop the nature of the interfaces and identify tasks involved in the process in order that an assessment of the model can be made and the Soft System Reliability Risk View can be produced.

4.4.7 An example to demonstrate the extended modelling notation set used to produce the Hard Systems Context Chart

To demonstrate the principles discussed in section 4.4.5 and to define the extended modelling notation set used to model the Hard Systems Context Chart consider the following example Hard System description.

An aircraft has two engines. The engines and aircraft have integrated electric, hydraulic, fuel and pneumatic services. The engines are controlled by a Full Authority Digital Engine Control (FADEC) system which controls the engine services.

The aim of the example would be to develop and model the Hard Systems described in order that a reliability assessment could be made which recognises and potentially includes Hard System components which cross technical and commercial boundaries (for example the engine manufacturer and the aircraft manufacturer would be different companies). This assessment could be used to support design activity in addition to the business case for Servitisation of this equipment.

In practice, the same model could be used by both the airframer and the engine manufacturer to resolve reliability issues which either cross these boundaries or when design mitigation is required from either system to support the other. The model would be developed in the same way and would only differ in terms of perspective and level of detail according to the end users requirements.

The first thing that must be done is to consider the main platforms of the proposed Hard System.
4.4.7.1 Hard System platform definition
The Hard System platforms are described in the example given by the aircraft, the engine and the FADEC system.

Each of these Hard System platforms will have some level of interdependency, but will relate to each other in terms of a rigid hierarchy in safety terms with respect to safety critical systems. It is this hierarchy and interdependency that should be identified for our purposes of assessment, as lines of responsibility defined by that hierarchy will also impact on design and contractual issues related to reliability for the system under assessment.

The notation used in the Hard System Context Chart for a Hard Systems platform is the grey box as shown in Figure 4.4.15.

![Diagram](Aircraft)

Figure 4.4.15 Showing the Hard System Context Chart notation for the 'aircraft'

4.4.7.2 Hard System platform function definition
The interdependency between platforms is described by the Hard System platform function.

The Hard System platform functions are effectively sub-systems of the Hard System platform.

Each Hard System platform may share some level of sub-system functionality with any of the other Hard System platforms. It is likely however that the ‘top most’ Hard System platform in the hierarchy will have a number of its sub-systems that do not interact with other Hard System platforms, but it is unlikely that many of the other subordinate Hard System platform functions will not interact with it either directly or through another Hard System platform if the function is safety critical to the ‘top most’ Hard System platform.
Based on the given example, the ‘aircraft’ Hard System platform will have a fuel subsystem which is a function of the aircraft which supplies the aircraft engine with fuel.

The notation used in the Hard Systems Context Chart for a Hard Systems platform function is the white box as shown in Figure 4.4.16.

![Aircraft Fuel System](image)

**Figure 4.4.16 Showing the Hard Systems Context Chart notation for the Aircraft Hard Systems platform function ’aircraft fuel system’**

In the same way as for Soft Systems ownership of the System boundary function is shown by placing it close to the Hard System platform to which it belongs and drawing a dotted line around the Hard System platform which links all of its respective functions as shown in Figure 4.4.17.

![Aircraft Fuel System](image)

**Figure 4.4.17 Showing the hard System Context Chart notation for Hard Systems platform and Hard System platform function ownership**
4.4.7.3 *Interface – how the Hard System platform functions interact*

The physical interface between Hard System platform functions or sub-system is not defined using this notation it is implied by the nature of the Hard System platform function such as ‘aircraft fuel system’ which indicates that all components in this functional group are concerned with the flow of fuel through the system as a whole. An indication is made on the Hard System Context Chart that there is an interface between each of the Hard System platform functions which share a common physical link. This interface link is shown as a full line between the respective Hard System platform functions as shown in Figure 4.4.18. This link might also be for instance a shared electrical, or a shared hydraulic link between components in either Hard System platform function.

The indication on the Hard System Context Chart does not show how many individual links there are between each Hard System platform function, there may be one link, there may be many, nor does it show the direction of the flow of the interface for the same reason. This level of detail is left for the modified Role Activity Diagram to describe. For this reason, the interface link on a Hard System Context Chart does not indicate a ‘process flow’ from one Hard System platform to another.

The interface between each Hard System platform function is denoted by a line joining the respective functions. Only functions which have this line joining them have an explicit interface. However they may have interfaces, with other Hard System platform functions (for instance, a hydraulic flow control valve may have both a hydraulic and an electrical interface if the flow control through the valve is effected by means of an electronic solenoid). These interfaces are not specified in detail at this level of abstraction as the key point of this level of modelling is to obtain an overview of Hard System platform to Hard System platform interface issues.
If a new or improved design is being modelled, adding a Hard System platform function to the model to develop the design here is acceptable and similar to the approach taken when data flow modelling object oriented software. That is that you are aware of the need of the ‘object’ or Hard System platform function, but
have not yet assigned either attributes of that object or interfaces from that object to other objects in the design space.

Defining interdependency in this way also enables a definition of scope for the analysis for the Hard System and means that some sub-systems that may have initially been thought to impact the design can be excluded and in the same manner or functions that are initially thought to have no direct impact on the system under assessment can be introduced.

4.4.7.4 Hard Systems Context Chart for the complex Hard System example
Using the description of the complex Hard System example given in section 4.4.7 and building on the model developed in Figures 4.4.15 to 4.4.18, the Hard Systems Context Chart for the complex Hard System example is shown in Figure 4.4.19.
Figure 4.4.19 Hard Systems Context Chart for the complex Hard System example
The Hard System platforms are defined as:

- Aircraft;
- Engine;
- FADEC.

These Hard System platforms are as defined in the example given. The Hard System platform functions of the Hard System may include all of the sub-systems of each Hard System platform or may just refer to those which are of interest to the assessment.

As can be seen from the example in Figure 4.4.19, some of the Hard System platform functions included are specific to that Hard System platform (aircraft electrical system), some are shared between two of the Hard System platforms (engine thrust reverser control valves and aircraft hydraulic system) and some are shared between all three of the Hard System platforms (aircraft fuel system, engine fuel system components and FADEC fuel system components).

As with the Soft System example, the Hard System interfaces can be highlighted to show which systems in particular are to be modelled in greater detail such that an assessment can be made using that lower level, more detailed model. Hard Systems are much better defined functionally than Soft Systems, so this is not as important a process as it is for Soft Systems which are usually less well defined. It is however important to understand not only the context of each of these Hard System platform functions and how they relate to the various platforms but also in order to ensure that all Hard System platform functions are considered in a complete and correct assessment of reliability and design activity.

This approach is shown in Figure 4.4.20. In this approach the engine oil system and the interfaces with the engine control system and its components are chosen to be modelled in more detail. It is recognised that the oil system is an engine function and it would be possible to model this system at an engine level without the need to break out the control system as a separate Hard System platform function. The differentiation shown here though could be due to:
The need for and level of differentiation of modelling of the Hard System depends on the perspective of the reliability assessment.
Figure 4.4.20 Hard System Context Chart highlighting the interfaces involved in monitoring the oil system
The more detailed model of the Hard System platform functions with highlighted interfaces is made using the modified Role Activity Diagram approach using the perspective defined in Figure 4.4.20 is shown in Figure 4.4.21. Here all of the Hard System platform functions shown in the Hard System Context Chart are replicated and the components of the respective Hard System platform functions are added and their interfaces and the direction of those interfaces are defined.

It should be noted that there are two extra additions to the notation here which provide further definition to the representation of a monitoring system and that is the connecting boxes in the ‘engine oil system’ which show that the system is a closed system, that means that there is no input from or output to another Hard System platform function and therefore the whole ‘engine oil system’ is encompassed by that Hard System platform function.

The other addition is that of the dotted interface line to the vibration monitoring device. This is meant to represent the fact that there is no physical connection between the monitor and the components being monitored. In this case, there is only an energy link. That is vibration energy is the only link between the component being monitored and the component monitoring it. A similar notation would be used in the case of acoustic monitoring. The model requires such a notation as there is a need to identify the fact that that component has a monitoring function that relates to it.
Figure 4.4.21 Detailed Hard System mapping of the Hard System platform functions highlighted in the Hard System Context Chart shown in Figure 4.4.20
4.4.7.5 Summary guidelines for modelling large complex Hard Systems

There is a need for the modelling notation to be able to deal with larger and more complex Hard System models than those provided in chapter 3. As discussed in section 4.4.4, the model increases in size when viewed over a number of levels of abstraction. This in turn complicates the reliability monitoring solution as legal and contractual responsibility for reliability modelling may extend across a number of these levels of abstraction, or may involve a close working relationship between different suppliers which requires an integrated monitoring system.

As these models increase in size, it is important that the system under study and its various dependencies (such as control authority and flow sources) between levels of abstraction are fully understood so that a contextual perspective can be provided as a framework for more detailed modelling. This is the purpose of the Hard Systems Context Chart.

The first step in developing the Hard Systems Context Chart is to determine the Hard System platforms. These are typically major systems artefacts and must be at the highest level of abstraction of the system under investigation. In the example given in Figure 4.4.20 Hard System platforms are ‘Aircraft’ or ‘Engine’. Unlike Soft Systems all Hard System platforms are likely to be defined in the first instance as they are designed for a specific purpose.

The next step is to define Hard System platform functions. These functions are important both to the higher level model of the system which is the Hard System Context Chart and the lower level model used to produce the Hard System Reliability Risk View as they form the connection point between the two models. These functions are the primary functions that are to be modelled, the Hard System platform that they belong to infers they are major subsystems of that platform and as such are likely to interface with major subsystems of other Hard System platforms.

The final step in developing the Hard System Context Chart is to determine the interfaces between the Hard System platform functions. For the Hard System Context Chart no greater interface definition is required other than an
acknowledgement that there is an interface between Hard System platform functions common to one or more Hard System platforms. More detailed modelling of the interface is performed in the lower level model which goes to form the Hard System Reliability Risk View. Details of how to perform interface modelling are given in examples in chapters 2 and 3.

Once the Hard System Context Chart is developed, then lower level modelling can proceed.

The aim of lower level modelling is to develop the Hard System Reliability Risk View. The Hard System Context Chart provides a guide on how this should be developed through the interfaces shown on it. Only Hard System platform functions that are connected through a shared interface can be modelled at a lower level.

The Hard System platform functions of interest and all those functions that they are interfaced to then become the principle platforms of the lower level model. This lower level modelling activity follows the same procedure as discussed in chapter 2. The aim is to develop the nature of the interfaces and identify tasks involved in the process in order that an assessment of the model can be made and the Hard System Reliability Risk View can be produced.

4.4.8 Integration with existing or suggested new quantitative assessment techniques for Hard and Soft Systems

The methodology as it stands and the enhancements discussed in this chapter are applicable to a wide range of systems. When modelled using this methodology some form of meaningful assessment can be made by identifying either vulnerable components in the case of Hard Systems or vulnerable interfaces in the case of Soft Systems with which it is possible to develop systems design in mitigation of that assessment.

It has to be acknowledged however that a number of other assessments will be made on the systems as part of the normal course of events in system
development, for example, design compliance assessments such as FMEA for Hard Systems and quality assurance audits for Soft System compliance.

It is difficult therefore to justify the extra resource required to develop specific models and assessments such as those proposed to support the development of reliability monitoring systems. A means therefore of integrating the proposed methodology, where possible with existing assessment techniques is therefore desirable.

In the case of Soft systems it is suggested that these models and the method for modelling does provide added value and specific and targeted benefit in understanding how the Soft System operates which does not exist in current assessments. Quantitative assessment of these models can be derived by adding resource allocation information and time and motion study information (time to perform a task in person hours and if available, the repeatability of that task time) to these models in order to understand where and how to optimise the processes they represent. For instance, can the process be automated or does more resource need to be allocated at a certain point in the process or does the process need to be changed or modified in any way to enable improvement?

For the Hard System assessment there is benefit in developing models based solely upon existing FMEA work, should an FMEA be available for the system under consideration.

In addition, such an integrated method would provide an enhanced overview of Hard System risk based on the information obtained from the FMEA.

A methodology based on the concepts discussed in this and preceding chapters for Hard Systems but modified such that it uses the FMEA rather than a system diagram to model the system is discussed in the following chapter.

### 4.5 Chapter Summary

This chapter critically appraises the methodology proposed so far and provides either mitigation against that criticism or extensions and enhancements to the methodology proposed.
Explanations have been provided for how to use the methodology to assess various types of Hard System and issues regarding which systems or attributes of those systems can be modelled or taken into account by this methodology.

The conclusions are that the methodology as defined and used in chapters 2 and 3 respectively is suitable for:

- Modelling redundant Hard Systems;
- Modelling multi lane (hot or cold standby Hard Systems);

Examples were given and modelled and general guidelines were provided for such instances.

The methodology was not recommended for modelling distributed Hard Systems because control authority is defined by computer networks rather than the dedicated hardware platforms which typify embedded safety critical systems.

In order to model larger and more complex Hard and Soft Systems, extensions to the methodology were required. The concept of Hard and Soft System Context Charts was introduced.

These Hard and Soft System Concept Charts are intended to provide a more holistic view of each system and the associated implications of that ‘wider viewpoint’ but are not meant to be models from which an assessment is made. They provide a contextualised framework upon which such an assessment can be made. This is done in order to ensure lower level modelling reflects the complete system with interfaces that are correctly identified and have a defined relationship in context with other Soft System Boundaries or Hard System platforms respectively within the system as a whole.

General guidelines were provided to describe how such models could be created.

Extensions to the modelling notation were also made in the case of those systems where system flow is contained within a single platform (see ‘engine oil system components’ in Figure 4.4.21 where the oil system flow is contained
within that platform function). Another extension to the modelling notation at that level was that for monitoring devices which share no physical interface with the component being monitored (in this case the interface flow link for the vibration monitoring device in Figure 4.4.21) such an interface is shown in the same way as a normal electrical interface except with a dotted interface line instead of a full line.

It should be noted that the purpose of this chapter has been to discuss extensions to the modelling notation. No actual attempt has therefore been made to link the Hard and Soft System models developed in this chapter, although it could be imagined that a sensor fault detected in by one of the sensors represented in the Hard System model shown in Figure 4.4.21 (for instance 'oil filter blocked pressure switch') could initiate the Soft System task of ‘Receive sensor input T11’ in Soft System boundary ‘Airframe – Engine Health Monitor’ described in Figure 4.4.14.

Finally it was noted how the methodology could be used to provide quantification information by integrating and displaying information from other sources on the low level or modified Role Activity Diagram models.
Chapter 5 - Application to more complex
Hard Systems

5.1 Introduction

In this chapter the ideas discussed in earlier chapters are extended to show how they would be applied to Hard Systems which are complex in nature or are larger in scale. They may also be existing systems that have been developed and established over a long period of time and now need to change to support a new Servitisation business model.

The approach will be to attempt to integrate the ideas of Hard Systems modelling that have been developed (in terms of system platforms, their components and related interfaces) and to extend those ideas by means of integration with established analysis techniques. The aim being to develop a modelling approach for those systems that have been formally analysed and documented in safety terms by the FMEA technique. The FMEA form of safety analysis is widely used and often forms part of the safety case for systems.

The ideas regarding modelling in previous chapters are still valid but the approach discussed here will be different as the default assumption is that the system has already been analysed in safety terms, and therefore modelling will be carried out using this as its source rather than design documents.

The object of this adapted Hard System modelling process will be, as in previous chapters:

- To determine which components are at risk, thus determining system vulnerability;
- To highlight the importance of particular monitoring functions and sensors in order to optimise the design giving a view of the monitoring system capability.
5.2 Considerations in Hard System Monitoring Systems

analysis for established or evolving complex designs

Before the modelling techniques used are discussed in full an outline of the problems that are addressed by this technique and the philosophy of the complex or large system modelling technique application are discussed more fully.

It is often the case that new products are based essentially on the design of previous products with slight changes that are intended to improve product performance or reduce product cost. Such an approach has a number of business and engineering advantages in that systemic issues are well understood and documented and experience of the design in service can be documented and addressed in later designs.

The approach used in developing the technique so far described has been to develop a view of system vulnerability by examination of the system design. This is a valid approach and especially in early stage concept design analyses, is the only approach possible.

For long established systems, or for those systems for which the information is available however, it is more appropriate to take advantage of the FMEA to form the base of the model as the FMEA will have established design weaknesses. It is those design weaknesses that are most pertinently addressed by a reliability monitoring system. A model which is derived from Hard System weakness is therefore desirable.

The FMEA document itself provides details on which components are the root cause of system failure. This analysis is, by its nature, an exhaustive one and the number of failure modes for large systems can be numerous, resulting in a large document from which it is difficult to develop a holistic view of the system. Using the FMEA as a basis for Hard System modelling does not invalidate the ideas and notation developed for modelling in previous chapters but the approach is different as the system is assumed to have already been analysed in safety terms therefore:
- It is not as important for the model to establish functional relationships as the FMEA should be organised such as these are obvious;

- It is not necessary to establish component design relationships (for example if one component supplies input to another component) as important component relationships such as these will be highlighted by the FMEA if there are safety consequences that relate to that relationship.

Producing a model of the Hard System from the FMEA in this way introduces a graphical perspective of the issues raised by the FMEA. When read in context, this should provide a clearer, more holistic and more accessible idea of the issues that the FMEA raises. These issues are addressed in this research and are discussed in further detail in this chapter.

5.2.1 Developing additional Hard System reliability monitoring system design features in response to identified safety issues

System design is optimised to its intended function, but this design is not necessarily ‘fit for purpose’ in terms of delivering the high level of system availability required for Servitisation business cases. This is because the prime driver for the design is that it not only fulfils its function but it also does so in a safe manner. Reliability and hence availability issues are considered (as they are related), but availability issues also encompass access and repair times and as such are judged ultimately by financial criteria. This has to be balanced by the need to comply with design features that are mandated by customer contract engineering issues on fitness for purpose or governing body certification issues.

Safety considerations also often require that the system is forced into a safe mode of operation or operation itself is limited or degraded which may affect its desired performance adversely and could potentially also affect system availability. For instance, loss of a component or function on an aircraft engine may force a safe mode of operation which means full engine power is not available.
When the business case requires increased availability, such events need to be either accurately predicted in order to schedule appropriate maintenance effort, which limits the consequences of system unavailability or to increase diagnostic capability, identify and isolate a fault in order to reduce the time taken to effect a system repair. Developing additional reliability monitoring system design features is therefore a potential design response to ensure increased system availability.

Of course, additional design features incur extra design cost and therefore require justification if they are introduced over and above existing design features. This justification is in itself problematical as such a proposal does not easily lend itself to a cost benefit analysis (as does a design feature which is introduced to perform a particular function) as the reliability monitoring system benefit is not easily quantifiable. This is because its design cost benefit has to be balanced against only a potential failure mechanism, which may occur infrequently, but if it were to occur would involve either a heavy financial penalty or adversely affect supplier reputation and customer relations.

For these reasons, where possible, systems or components that are already part of the established system design are used to support the reliability monitoring system effort. In itself this approach is a good one as it reduces design cost and effort. The disadvantage of this approach however, is that not all vulnerable parts of the system are necessarily covered as such sensors that do form part of the design as a whole are usually implemented to effect control schema. In addition, if the additional responsibilities of the sensors that are identified as being able contribute to the reliability monitoring system are not recognised, their impact is not properly assessed in reliability, cost and quality terms.
5.3 Required approach to address these Hard System considerations

5.3.1 Understanding the risk to system reliability

Although the FMEA is an extremely useful document, it does not easily lend itself to an open interpretation of its contents in a holistic manner. In addition it is biased toward a system safety viewpoint and not a reliability viewpoint (as safety considerations are of greater importance than reliability considerations in terms of developing and certifying a system).

What is required is an approach that uses the information contained in the FMEA to produce a view that establishes the systems risk with respect to purely reliability issues thereby identifying which of the system components put the system most at risk of reduced availability. This is similar to the aim of the Hard Systems modelling approach used in earlier examples.

If the system risk can be further qualified and quantified with regard to the type of risk that component failure presents to the system by information contained in the FMEA, then this provides a view against which a reliability monitoring system design (and its associated costs) can be assessed.

It could also be said that an added benefit of such an approach is that it will act as a check against any assumptions made in the FMEA as component risk in this view would be transformed back in to design context. For example, the perceived risk of an individual component across a number of modules may not be obvious in a large document, but if a model were to be developed where such a components impact is more obvious across a number of modules, it may lead to a reappraisal of that particular component.

5.3.2 Developing Hard System reliability mitigation

The response to this perceived Hard System risk is essentially related to the design requirements of the reliability monitoring system as a whole. In previous examples, it has been sufficient to show through Hard System modelling which
components are monitored and how that monitoring mitigates the risk to availability in conjunction with the development of a Soft System that supports the monitoring function.

When systems are more complex or have a large number of components, there is the potential for either a greater number of Hard System monitoring functions or a more integrated and optimised combination of existing monitoring functions. Providing a model for these scenarios becomes part of the mitigation for both the reliability case and the reliability monitoring system design and all of these mitigating scenarios must be accounted for.

Account should also be made of corroboration of reliability failures. If it is possible for more than one sensor or monitoring function to distinguish a failure and detect its root cause then, this should be noted as it provides confidence in the ability of the system overall to detect and provide diagnosis, giving useful information for the following maintenance action, which in turn provides increased system availability.

A model or more likely, a number of models of each type of reliability monitoring scenario for each particular failure case is required to be developed in order to define this mitigation in system design terms. It should be noted that system design that provides this mitigation will of necessity not only involve Hard System monitoring components, but also a Soft System definition of the processes required to support the human response to failure events identified by the reliability monitoring system, the confirmation of those events and the decisions required to ensure continued system availability.

The definition of these aspects of the reliability monitoring system as a whole should provide not only the requirements of the system design, but also an understanding of the exposure of risk to that part of the system that is either not capable of being monitored or which is not cost effective to monitor.
5.4 Types of Hard System reliability mitigation

There are a number of methods of Hard System mitigation for at risk components. They are:

1) A routine maintenance programme;

2) Replacement of components on a calculated life basis whether failed or not;

3) Diagnostic maintenance;

4) Predictive maintenance;

5) A combination of two or more of the above.

Which mitigation type or combination of types is used will depend on the business case adopted. Increasingly those adopting Servitisation business models provide maintenance contracts which are supported by predictive and diagnostic maintenance routines which of necessity require reliability monitoring system sensors to provide information on which to base maintenance procedures in order to maximise system availability.

These systems are more costly to implement initially, but are beneficial in the long term and are the most cost effective if the whole of the life cycle of the system is taken into account as:

- The systems are adaptable to detect new failure mechanisms in the field (potentially introduced through system modification and update);

- A Regular maintenance programme does not guarantee availability and in addition has the potential to increase failure rates by inducing faults when ‘healthy’ equipment is removed and checked during the maintenance operation;

- Maintenance will only be carried out as and when required and not as a major regular overhaul programme;
• There is the potential to collect data regarding the use of the system for future systems improvements;
• There is the potential to collect data on how the system is operated by the lessee of the Servitised product, to ensure contractual obligations are not exceeded.

It is these systems that are predominantly addressed by this research.

5.4.1 Predictive and Diagnostic reliability monitoring systems

In some instances (for example, distributed systems, have components with high levels of autonomy in terms of prediction and diagnosis of health), it is accepted that there may be a blurring of the distinction with these reliability monitoring functions.

Predictive monitoring is defined in terms of this research as a generally more complex task, which takes data from a variety of sensors in order to predict against a modelled behaviour or according to an appropriate algorithm, when a system will fail. Due to data acquisition, processing power and large scale data transmission constraints (and in some cases the need for expert interpretation) such monitoring is expected to be performed remotely from the system operation.

Diagnostic monitoring is defined in terms of this research as being monitoring that is usually performed in situ by a sensor or suite of sensors dedicated to determining either a failure condition or to establishing conditions of incipient failure.

Whilst it is accepted that the Hard System designer will provide mitigation against both safety and reliability aspects as much as possible with system design, both of these methods can be used as complementary monitoring strategies towards risk mitigation of system availability.

It is important to the success of this approach that the type of reliability monitoring system adopted (or combination of systems) in mitigation is identified in order that, as with the simple system example, exposure of risk to the business case can be determined. Each type of reliability monitoring system has attendant
design constraints, which have a bearing upon the overall business case for the service, the long term consequences of which must be appreciated in full.

5.5 **Summary of the required Hard Systems modelling approach for complex systems**

In summary, the required approach will use existing or evolving Hard System documentation, preferably either an FMEA or as in the example below, a functional FMEA to establish a model of system reliability risk or vulnerability.

This modelled view will then be used to develop mitigation against that System risk or vulnerability through the development of a model or models of reliability monitoring design solutions which will provide an appreciation of reliability monitoring system coverage against those components deemed to be most vulnerable to reliability issues in either an engineering or a commercial sense (in other words any condition which provides a negative impact on the overall business case). The type of design mitigation solution that is proposed will be dependent as much on the vulnerable component being monitored as it does on the overall business case.

The key approach used in the example given below however will be to suggest design mitigation using predictive and diagnostic reliability monitoring system approaches.

Those components that are for whatever reason vulnerable but are not mitigated through a monitoring system shall also be identified through this process.

The process as a whole provides:

- Information on what sensors are required to provide predictive and diagnostic reliability monitoring mitigation;
- Identification of sensors that have a reliability monitoring role in addition to other (such as control) roles;
- Identification of the means of data transfer that will be required to support the reliability monitoring effort;
• A model of the reliability monitoring system approach in terms of what level of predictive and diagnostic monitoring is required and whether or not the approaches are complementary;

• Identification of individual system component vulnerability;

• Identification of mitigation approaches against that system vulnerability;

• An understanding of the components that will be required in monitoring functions and the type of component or function they are monitoring;

• An indication of the level of component risk to reliability in terms of the system design in order to check that design assumptions made in the FMEA are valid in a system wide context.

5.6 Hard system modelling using an example functional FMEA

In this modelling application, instead of using the system design documentation such as design drawings and schematics to produce a model to evaluate system vulnerability, a functional FMEA is used as a source document for a system model.

Producing such a view it is suggested:

1) Aids understanding of system vulnerability by identifying components that are either partially or not monitored;

2) Tests assumptions made in the FMEA with regard to monitoring functions;

3) Tests the veracity of the FMEA;

4) Apportions monitoring functions to vulnerable components.

5.6.1 Review of the Functional FMEA modelling approach

Using the functional FMEA to model system vulnerability has the following advantages:

1) System/ component weaknesses are already derived;

2) Effects of component failure are identified;
3) Causes of component failure are identified;

4) An evaluation of the effect of the reliability failure is available and can be depicted;

5) The reliability model forms part of the integrated design suite and informs design of the Hard System (and ultimately the Soft System).

The disadvantages are:

1) The assumptions made in the FMEA are not always visible (without a full understanding of the design);

2) The actual system design is not apparent from the FMEA and therefore will not be fully represented by the Hard System model only functional aspects of the design will be represented;

3) Effects analysis is carried out on a module by module basis. More than one module may refer to a single component effect in relation to their module, so there may be duplication;

4) The FMEA because of its size; is not easily interrogated in order to ultimately develop a view of Hard System Reliability Risk;

5) The FMEA is assumed to be complete and correct, if this is not the case the resulting model is compromised.

These disadvantages need to be addressed in some way if a representative Hard System model is to be produced.

The example FMEA shown in Table 5.6.1 is itself organised into modules, which represent in this case, engine sub-assemblies. There are a number of functional failures identified for each module, and there is a component failure identified for each line item of functional failure. Each line item also has assigned to it a Performance Effect Criticality (PEC) number, which reflects how easily the operator can observe the failure (Evident effects) and those that are not easily observable by the operator (Hidden effects). There is a further classification in
terms of the effect that is whether it is either a safety issue, or a non-safety issue, or an operational issue or an economic issue.

5.6.2 Using the FMEA to identify vulnerable components

As the FMEA is used to effectively 'model' the system, no formal description of the engine (the subject of the functional FMEA) is provided here. It is assumed that the FMEA is an acceptable emulation of the system describing those functions which are at most risk of reliability issues, whilst not being a full description of the system.

This approach does not mean that a system could not be modelled from design documentation. If such an approach were taken instead of using the FMEA however, a further step would be required to add information (or make assumptions) either regarding component vulnerability or the functional importance of each component as was the case with the simple system example demonstrated earlier.

Using the FMEA as the primary source document allows modelling to take place in a focussed way with regard to the proposed analysis technique and also provides a check against FMEA assumptions once the model is completed. In addition, it is more likely that in a real world application, this approach would be favoured as it integrates into any documentation set that would be required to be produced for a system design.

5.6.3 Identifying vulnerable modules and components from the FMEA

In order to identify vulnerable components a table such as the Table 5.6.1 is created for each module listed in the FMEA with its associated information.

The example FMEA describes a large engine which is started using a pneumatic arrangement to turn the engine and reduce inertia before fuel and ignition systems start the engine. The FMEA lists all of the modules which have components with identified safety and reliability failure effects along with a Performance Effect Category (PEC) which describes the effect of failure of that particular component in that module. If the component is not listed in the FMEA
it is assumed not to have an impact on reliability and therefore will not impact on availability.

The key for the PEC in the following tables is as follows:

1  =  Evident safety effect;
2  =  Evident reliability effects;
3  =  Evident commercial effects;
4  =  Non-evident safety effects;
5  =  Non-evident, non safety effects.

The vulnerable components from each module in each line item in the FMEA are placed in this table alongside the Performance Effect Criticality. Failure effects are ignored, the idea is to develop as 'condensed' a model as possible which conveys known information regarding product structure, interface and vulnerability.

The FMEA provides the vulnerable component table shown in Table 5.6.1. It should be noted that some components have more than one effect noted in the FMEA. In this case, both or more PEC effects are recorded in the table.
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERIPHERAL GEARBOX MODULE</td>
<td>Bevel Gearshaft</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Gears</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Housing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Casing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Seals</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Deoiler</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Adapter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Rotor Group</td>
<td>2</td>
</tr>
<tr>
<td>DRIVE ROD</td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Shaft</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Adapters</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Spline</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Seal</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>PAS Shroud</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Shroud Seal</td>
<td>3</td>
</tr>
<tr>
<td>EXTERIOR GEARBOX</td>
<td>Gearshafts</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Casing</td>
<td>2 &amp; 3</td>
</tr>
<tr>
<td></td>
<td>Nozzle</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 5.6.1 Table of vulnerable components by module
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENGINE TRANSITION</td>
<td>Intermediate Casing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>A Frame Attachment</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Feedpipe/ Bearing Jet</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Pipe</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Jet</td>
<td>5</td>
</tr>
<tr>
<td>PNEUMATIC STARTER AND VALVE SYSTEM</td>
<td>Starter Control Valve</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Starter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Ducting</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>External Gearbox</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Drive Rod</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Exterior Gearbox</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>2</td>
</tr>
<tr>
<td>OIL DISTRIBUTION</td>
<td>Pressure Pump</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Feed System (As A Whole – Blocks, Leaks)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Pressure Filter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge System (As A Whole – Blocks, Leaks)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Pump</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Pressure Relief Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Filter</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.1 Table of vulnerable components by module (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>OIL DISTRIBUTION (continued)</td>
<td>Scavenge Filter By Pass Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Cooler</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Vent Line</td>
<td>2</td>
</tr>
<tr>
<td>OIL STORAGE</td>
<td>Oil Tank</td>
<td>2</td>
</tr>
<tr>
<td>FUEL AND AIR CONTROL SYSTEM</td>
<td>Engine Controller</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Filter Differential Pressure Transducer</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Speed Probe</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Dedicated Alternator</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Quantity Sensor &amp; Transmitter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Ignition System</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Meter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Fuel Master Lever</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Flow Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Temperature Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5.6.1 Table of vulnerable components by module (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUEL AND AIR CONTROL SYSTEM</td>
<td>Scavenge Oil Filter</td>
<td>5</td>
</tr>
<tr>
<td>(continued)</td>
<td>Differential Pressure Transducer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Start Air Valve</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Instinctive Disconnect Push Button</td>
<td>5</td>
</tr>
<tr>
<td>Engine Controls – General</td>
<td>Timer Module</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Master Lever</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Time Delay Relay</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Mode Selection Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Start Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Selector Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Status Relay</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td>IGNITION GENERAL</td>
<td>Ignition System (&amp; Connection)</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>3</td>
</tr>
<tr>
<td>INDICATING</td>
<td>Fuel Flow Sensor &amp; Transmitter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Filter Differential Pressure Transducer</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.1 Table of vulnerable components by module (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUEL DISTRIBUTION</td>
<td>Primary Fuel Filter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Secondary Fuel Filter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Main Engine Pump</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td>ANALYSERS</td>
<td>Vibration Monitoring Unit</td>
<td>3 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Vibration Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Electro Magnetic Chip Detector</td>
<td>5</td>
</tr>
<tr>
<td>OIL SYSTEM INDICATING</td>
<td>Oil Pressure Sensor &amp; Transmitter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Oil Quantity Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Temperature Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Low Pressure Switch</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Vibration Monitoring Unit</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Electro Magnetic Chip Detector</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5.6.1 Table of vulnerable components by module (continued)
5.6.4 Partitioning the module components according to interface type

Table 5.6.1 is a large table which has to be further partitioned in order to build a sensible model for the Hard System. The main consideration is how best to partition the table in order to build the model.

As with the design centric approach to modelling Hard Systems adopted in earlier chapters it is important to develop:

1) System platforms;
2) System platform functions;
3) Identify interfaces between functions.

System platforms are less important here as the system platform functions are already defined as modules. Interfaces are however important, as components that share interfaces are those most likely to provide opportunities of developing corroborative failure techniques. The best way of partitioning the modules therefore is on the basis of interface.

Using the vulnerable component list shown in Table 5.6.1, it can be seen that these components are partitioned on a module by module basis. It can also be seen that some components appear in more than one module; this is often because these different failure modes relate to different interface conditions.

Partitioning the system in this way highlights the information provided by the FMEA and does so in the context of shared interface, this partitioning also provides an indication of component functionality within the same context. The components on the vulnerable component list are therefore taken and partitioned further on the basis of the type of interface that they have with other components. Interface types considered are:

1) Pneumatic;
2) Hydraulic;
3) Electrical;
4) Mechanical.
Software interfaces are assumed to be electrical.

If a component has more than one interface it is shown in each of its respective interface lists, however, the link between components and the modules in which they exist must be maintained.

If components appear in more than one FMEA module, they should be noted as duplicated components in order that they can be dealt with appropriately at a later stage. Table 5.6.2 shows this partitioning by interface using Table 5.6.1 as a basis. The Module name may also appear in this table on more than one occasion. This is the case for the ‘Peripheral Gearbox module’, this is because this module supports components in both mechanical and fuel interfaces. This is shown in the table by interface type being placed in brackets underneath the Module name so that it is apparent which components belong to which interface type.

<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERIPHERAL GEARBOX MODULE (Mech)</td>
<td>Bevel Gearshaft</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Gears</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Housing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Casing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Rotor Group</td>
<td>2</td>
</tr>
<tr>
<td>PERIPHERAL GEARBOX MODULE (oil)</td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.2 Partitioning along the lines of interface
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERIPHERAL GEARBOX MODULE (fuel)</td>
<td>Fuel Seals</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Deoiler</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Adapter</td>
<td>2</td>
</tr>
<tr>
<td>DRIVE ROD (oil)</td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Seal</td>
<td>2</td>
</tr>
<tr>
<td>DRIVE ROD (mech)</td>
<td>Adapters</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Shaft</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Spline</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>PAS Shroud</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Shroud Seal</td>
<td>3</td>
</tr>
<tr>
<td>EXTERIOR GEARBOX (mech)</td>
<td>Gearshafts</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Nozzle</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Casing</td>
<td>2 &amp; 3</td>
</tr>
<tr>
<td>EXTERIOR GEARBOX (Oil)</td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>3</td>
</tr>
<tr>
<td>ENGINE TRANSITION MODULE (mech)</td>
<td>Intermediate Casing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>A Frame Attachment</td>
<td>2</td>
</tr>
<tr>
<td>ENGINE TRANSITION MODULE (oil)</td>
<td>Oil Seals</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Feedpipe/ Bearing Jet</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Pipe</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Jet</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5.6.2 Partitioning along the lines of interface (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNEUMATIC STARTER AND VALVE SYSTEM (air)</td>
<td>Starter Control Valve</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Starter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Ducting</td>
<td>2</td>
</tr>
<tr>
<td>PNEUMATIC STARTER AND VALVE SYSTEM (mech)</td>
<td>External Gearbox</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Drive Rod</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Exterior Gearbox</td>
<td>2</td>
</tr>
<tr>
<td>PNEUMATIC STARTER AND VALVE SYSTEM (electrical)</td>
<td>Engine Controller</td>
<td>2</td>
</tr>
<tr>
<td>OIL DISTRIBUTION (oil)</td>
<td>Pressure Pump</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Feed System (As A Whole – Blocks, Leaks)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Pressure Filter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge System (As A Whole – Blocks, Leaks)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Pump</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Pressure Relief Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Flow Control Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Filter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Filter By Pass Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Cooler</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Vent Line</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.2 Partitioning along the lines of interface (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>OIL STORAGE (oil)</td>
<td>Oil Tank</td>
<td>2</td>
</tr>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (electrical)</td>
<td>Oil Pressure Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Temperature Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Quantity Sensor &amp; Transmitter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Dedicated Alternator</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Filter Differential Pressure Transducer</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Flow Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Speed Probe</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Ignition System</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Instinctive Disconnect Push Button</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5.6.2 Partitioning along the lines of interface (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (fuel)</td>
<td>Fuel Meter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Fuel Master Lever</td>
<td>2</td>
</tr>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (air)</td>
<td>Start Air Valve</td>
<td>5</td>
</tr>
<tr>
<td>Engine Controls – General (fuel)</td>
<td>Master Lever</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td>Engine Controls – General (electrical)</td>
<td>Mode Selection Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Timer Module</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Time Delay Relay</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Start Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Selector Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Status Relay</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td>IGNITION GENERAL (electrical)</td>
<td>Ignition System (&amp; Connection)</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>3</td>
</tr>
<tr>
<td>INDICATING (electrical)</td>
<td>Fuel Flow Sensor &amp; Transmitter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Filter Differential Pressure Transducer</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>2</td>
</tr>
<tr>
<td>FUEL DISTRIBUTION (fuel)</td>
<td>Primary Fuel Filter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Secondary Fuel Filter</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.2 Partitioning along the lines of interface (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUEL DISTRIBUTION (fuel - continued)</td>
<td>Main Engine Pump</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td>ANALYSERS (electrical)</td>
<td>Vibration Monitoring Unit</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Vibration Transducer</td>
<td>5</td>
</tr>
<tr>
<td>ANALYSERS (oil)</td>
<td>Electro Magnetic Chip Detector</td>
<td>5</td>
</tr>
<tr>
<td>OIL SYSTEM INDICATING (electrical)</td>
<td>Oil Pressure Sensor &amp; Transmitter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Oil Quantity Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Temperature Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Low Pressure Switch</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Vibration Monitoring Unit</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Electro Magnetic Chip Detector</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5.6.2 Partitioning along the lines of interface (continued)
5.6.5 Grouping module components according to interface

The point of this partitioning process is to eventually arrive at a 'model' which shows modules and components that belong to a particular interface type that are known to be vulnerable (as they are identified as such in the FMEA). Note that only vulnerable components are identified and modelled, there may be other components that also 'belong' to the interface, but are not considered as vulnerable. These are not shown on this model.

The final adjustment made to the tabular data sourced by the FMEA is therefore to group modules and their corresponding module components according to their interface type. In this way all modules and components that belong to specific interface types are represented in that interface type.

Tables 5.6.3 to 5.6.7 inclusive show the necessary adjustments to the information provided by Table 5.6.2 for each interface and hence provide the final arrangement from which modelling can take place.
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERIPHERAL GEARBOX MODULE (Mech)</td>
<td>Bevel Gearshaft</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Gears</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Housing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Casing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Rotor Group</td>
<td>2</td>
</tr>
<tr>
<td>DRIVE ROD (mech)</td>
<td>Adapters</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Shaft</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Spline</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>PAS Shroud</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Shroud Seal</td>
<td>3</td>
</tr>
<tr>
<td>EXTERIOR GEARBOX (mech)</td>
<td>Gearshafts</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Nozzle</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Casing</td>
<td>2 &amp; 3</td>
</tr>
<tr>
<td>ENGINE TRANSITION MODULE (mech)</td>
<td>Intermediate Casing</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>A Frame Attachment</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Tail Bearing Housing</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Baffle</td>
<td></td>
</tr>
<tr>
<td>PNEUMATIC STARTER AND VALVE SYSTEM (mech)</td>
<td>External Gearbox</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Drive Rod</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Exterior Gearbox</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.3 Modules grouped by mechanical interface
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNEUMATIC STARTER AND VALVE SYSTEM (air)</td>
<td>Starter Control Valve</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Starter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Ducting</td>
<td>2</td>
</tr>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (air)</td>
<td>Start Air Valve</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5.6.4 Modules grouped by pneumatic interface

<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>OIL DISTRIBUTION (oil)</td>
<td>Pressure Pump</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Feed System (As A Whole – Blocks, Leaks)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Pressure Filter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge System (As A Whole – Blocks, Leaks)</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Pump</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Pressure Relief Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Flow Control Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Filter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Scavenge Filter By Pass Valve</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Oil Cooler</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Vent Line</td>
<td>2</td>
</tr>
<tr>
<td>OIL STORAGE (oil)</td>
<td>Oil Tank</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.5 Modules grouped by hydraulic interface
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXTERIOR GEARBOX (oil)</td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>3</td>
</tr>
<tr>
<td>EXTERNAL GEARBOX (oil)</td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>2</td>
</tr>
<tr>
<td>DRIVE ROD (oil)</td>
<td>Bearings</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Seals</td>
<td>2</td>
</tr>
<tr>
<td>ENGINE TRANSITION MODULE (oil)</td>
<td>Oil Seals</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Feedpipe/ Bearing Jet</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Pipe</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Jet</td>
<td>5</td>
</tr>
<tr>
<td>ANALYSERS (oil)</td>
<td>Electro Magnetic Chip Detector</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 5.6.5 Modules grouped by hydraulic interface (continued)

<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERIPHERAL GEARBOX MODULE (fuel)</td>
<td>Fuel Seals</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Deoiler</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Adapter</td>
<td>2</td>
</tr>
<tr>
<td>FUEL DISTRIBUTION (fuel)</td>
<td>Primary Fuel Filter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Secondary Fuel Filter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Main Engine Pump</td>
<td>2 &amp; 5</td>
</tr>
</tbody>
</table>

Table 5.6.6 Modules grouped by fuel interface
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (fuel)</td>
<td>Fuel Meter</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Fuel Master Lever</td>
<td>2</td>
</tr>
<tr>
<td>Engine Controls – General (fuel)</td>
<td>Master Lever</td>
<td>2 &amp; 5</td>
</tr>
</tbody>
</table>

Table 5.6.6 Modules grouped by fuel interface (continued)

<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANALYSERS (electrical)</td>
<td>Vibration Monitoring Unit</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Vibration Transducer</td>
<td>5</td>
</tr>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (electrical)</td>
<td>Oil Pressure Sensor &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Temperature Sensor &amp; Transmitter &amp; Transmitter</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Filter Differential Pressure Transducer</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Oil Quantity Sensor &amp; Transmitter</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 5.6.7 Modules grouped by electrical interface
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>OIL SYSTEM INDICATING (electrical)</td>
<td>Oil Low Pressure Switch 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Vibration Monitoring Unit 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Electro Magnetic Chip Detector 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Sensor &amp; Transmitter 2&amp;5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Quantity Sensor &amp; Transmitter 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Temperature Sensor &amp; Transmitter 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Oil Pressure Filter Differential Pressure Transducer 5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scavenge Oil Filter Differential Pressure Transducer 5</td>
<td></td>
</tr>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (electrical)</td>
<td>Engine Controller 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fuel Filter Differential Pressure Transducer 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dedicated Alternator 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ignition System 2</td>
<td></td>
</tr>
</tbody>
</table>

Table 5.6.7 Modules grouped by electrical interface (continued)
<table>
<thead>
<tr>
<th>Module</th>
<th>Component</th>
<th>PEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUEL &amp; AIR CONTROL SYSTEM (electrical)</td>
<td>Instinctive Disconnect</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Push Button</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fuel Flow Sensor &amp; Transmitter</td>
<td>2</td>
</tr>
<tr>
<td>Engine Controls – General (electrical)</td>
<td>Mode Selection Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Timer Module</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Time Delay Relay</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td></td>
<td>Start Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Selector Switch</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Status Relay</td>
<td>2 &amp; 5</td>
</tr>
<tr>
<td>INDICATING (electrical)</td>
<td>Fuel Filter Differential Pressure Transducer</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Fuel Flow Sensor &amp; Transmitter</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>2</td>
</tr>
<tr>
<td>PNEUMATIC STARTER AND VALVE SYSTEM (electrical)</td>
<td>Engine Controller</td>
<td>2</td>
</tr>
<tr>
<td>IGNITION GENERAL (electrical)</td>
<td>Ignition System (&amp; Connection)</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Engine Controller</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 5.6.7 Modules grouped by electrical interface (continued)
5.7 Using the FMEA processed information to develop the Hard System Reliability Risk Views

The FMEA although it defines vulnerable components, does not show those components in a system context, so it is difficult to gain a clear picture of how the components are linked in a system and which parts of that system as a whole are most vulnerable. This may be considered unimportant as long as the system design as a whole addresses these concerns by providing system redundancy or compensation on component failure.

In terms of reliability monitoring systems however, a system level appreciation of component vulnerability is absolutely necessary as designing functionality to monitor these vulnerable components requires this. This systems level approach ensures that the developed functionality is fit for purpose and optimised such that the most is made of the sensors available to provide the detection and corroboration where possible of component failure. This corroborative evidence in particular may be an ‘indirect’ detection of failure. Such a failure is one which is experienced by some other part or function of the system, but which by nature of a shared interface relationship is evidence of a failure of the monitored component.

A simple example would be of an oil debris monitoring system, which could detect bearing failure using a chip detector at the sump but also would corroborate this by means of growth in vibration amplitude at a specific bearing related frequency. Both the bearing and the sump share a physical interface which allows one method of detection, but the corroborative evidence is supplied by a vibration monitor which is in place as a safety feature to detect out of balance forces from a purely safety perspective. The bearing shares an interface with both the fluid system and the vibration system, but each interface is distinct. Such a monitoring arrangement is considered to be the ideal as the failure effect is seen in separate interfaces ensuring that there is no possibility of a false alert because of the independence of each interface.
Before design decisions such as these can be made however, there must be a full understanding of which are the most vulnerable components and whether or not any monitoring functionality can be applied to them in particular. This allows both the designer and the analyst to judge the fitness for purpose of the monitoring strategy for their respective viewpoints, an idea of potential monitoring system coverage and its effectiveness against the associated failure risks as determined by the PEC is therefore required.

A Hard System model has to be developed which not only reflects the FMEA, but also provides a design context in order to establish:

1) Which components are being monitored;
2) Which components are performing a monitoring function;
3) Which components are not monitored.

Only when such a model is developed, can mitigation by means of a monitoring strategy against component and therefore system vulnerability begin.

The first step however is to develop the Hard System model. As the FMEA is being used to develop this model it seems reasonable that this model becomes the Hard System Reliability Risk View when PEC information from the FMEA is applied to it.

With vulnerable components partitioned and grouped according to interface by means of the manipulation of the tabular data from the FMEA, the next step is to produce the Hard System Reliability Risk View from this tabulated data. This model will show the vulnerability of each component identified by the FMEA (and its level of vulnerability as defined by the Performance Effect Criticality number) for each specific interface identified.

The Hard System Reliability Risk View for each interface provides a baseline model of system vulnerability for the components connected by that interface and by implication an appreciation of the risk which each component is considered to exhibit by nature of its function with respect to the system module to which it belongs within that interface.
The advantages in developing the Hard System Reliability Risk View using this information are as follows:

- It presents relevant FMEA information in a more focussed and easily digestible format;

- All components relating to a specific interface are shown at a system level, providing a view from which reliability monitoring strategies can be developed.

In effect, the Hard System Reliability Risk View becomes a graphical version of the FMEA. It should be remembered however that this is only a ‘graphical FMEA’ within the context of the analysis that is developed by this research.

5.7.1 Development and notation of the Hard System Reliability Risk Views using information from the FMEA

The first step toward developing the Hard System Reliability Risk View is to produce a diagram that includes all modules and their respective components which relate to each particular interface type.

In the same way as with previous examples using the modified Role Activity Diagram notation, the module is thought of in the same way as a Hard System platform function and the same notation for Hard System platform function is applied. A reference to the interface type is made in italics before the module name. Therefore each module is identified as a grey boundary box as shown in Figure 5.7.1.

![Figure 5.7.1 Example of notation used to show a module as a Hard System platform function](image)

271
In line with notation used previously, each component is represented by an annotated box, which is in turn placed within the module boundary box to which it is assigned. The component box default colour is black as shown in Figure 5.7.2 below.

![Figure 5.7.2 Notation used for a component belonging to a Hard System platform function – no Performance Effect Criticality assigned](image)

Each component box is then identified in terms of its Performance Effect Criticality which is obtained from the FMEA by colouring the box, relating a colour to a PEC category. If the module belongs to more than one PEC category the component box can be shown with each colour as ‘bands’ as shown as an example in Figure 5.7.3, in the case of a box having more than two Performance Effect Criticality categories, a third can be shown as a ‘shadow’ to the box. If there are more than three Performance Effect Criticality categories associated with the component, the most hazardous Performance Effect Criticality categories will be shown.

In previous examples a similar approach has been taken by colouring the component boxes in order to differentiate between those components which are monitored, those which are not monitored and those which fall into neither category. Although useful in a broad sense, such an approach is not as targeted for reliability considerations as one which identifies level of vulnerability through PEC categories.
A Hard System Reliability Risk View should be produced for each interface type with each view particular to the modules and components that share that particular interface.

Therefore all components belonging to module A with an electrical interface are placed on the electrical interface Hard System Reliability Risk View in a module A boundary box, all components belonging to module A with a mechanical interface are placed on the mechanical interface Hard System Reliability Risk View in a module A boundary box and so on until all components in module A are assigned.

Figure 5.7.4 below shows a Hard System Reliability Risk View for an oil system interface. Note that in this style of representation, the functional design relationship of the components in the modules is not shown. What is recorded in this diagram is the consequence of the FMEA analysis for each of the sensors and components that are related to that particular interface. This information forms a contextual basis for understanding which components are at high risk and which sensors are potentially available to monitor the risk to those components.

It is still important to identify which components belong to which module in the FMEA both because of traceability and because of the need to understand which modules are used where in the mitigation, for example some sensor modules
could be involved in a number of reliability monitoring system mitigation views, which are discussed further in section 5.8 below.

This style of representation differs from that of the Hard System Reliability Risk View shown in Figure 3.4.3 as it does not attempt to provide any design relationship information (which reduces the complexity of the model) but still retains component relationship information layered with detail of the vulnerability of the components within that interface.

A collection of such views which models the complete Hard System provides a concise and graphical, rather than textual, view of Hard System vulnerability information provided by the FMEA.
Figure 5.7.4 A Hard System Reliability Risk View for the oil system interface
5.8 Developing reliability monitoring mitigation cases against the Hard System Reliability Risk View

It is recognised that the FMEA, upon which the Hard System Reliability Risk View is based, will primarily identify system failure cause and safety systems will be in place to mitigate safety issues that develop from these failure causes.

The aim here though is to use the Hard System Reliability Risk View to develop a reliability mitigation strategy by examining vulnerability issues and then by mitigating against failure causes by the use of existing systems and sensors (ideally) to either predict failure or to accurately and speedily diagnose the root cause of any failure that does take place.

System vulnerability in a reliability sense is dependent ultimately on what failure cases can be identified and detected. All techniques used to detect failures are dependent on accurate, appropriate and sufficient data in terms of volume and quality with which to perform either diagnostic or predictive failure detection in addition to an understanding of the physics of failure of the component or components that are in the process of failing.

It follows therefore that the overall detection capability is at least partially dependent upon reliability monitoring system design and that design in turn provides mitigation against Hard System vulnerability. The aim then is to define reliability monitoring system solutions for each known failure mode and for this set of solutions to form the mitigation against system reliability ‘risks’.

It should be noted that what is suggested here is that a particular necessary set of components is defined for a particular detectable failure mode. If a sensor component is used more than once in these solutions, this is acceptable as such an approach will highlight that sensor component regarding its importance to reliability mitigation rather than purely a system function.

This is valuable information for both the system designer and reliability engineer. The safety case should be reassessed based on this information as system design changes may need to be made to support this new position. For
example, more ‘important’ sensors are developed to a higher design specification or are themselves provided with redundant systems to cover for their failure. By the same token, sensor performance and accuracy may also need to be revised to support the reliability effort.

Once each of these reliability monitoring system design solutions is defined they should be laid against the Hard System Reliability Risk View in order to determine which of the system components are monitored and which are not.

5.8.1 Principles behind developing Hard System Reliability Risk Mitigation Views

The Hard System Reliability Risk View for each interface provides a complete view of all the components of the system at risk and their relative risk according to the PEC defined in the FMEA. Whilst this is acceptable in establishing risk, when developing mitigation views, the design of the monitoring system (in terms of which components are involved in monitoring which components) has to be taken into account. In addition the proposed method of monitoring will also have a bearing on the overall system reliability mitigation (which includes the mitigation provided by Soft Systems).

Predictive monitoring for our purposes is characterised as ‘off line’, data intensive and providing a systemic view of system health, whereas Diagnostic monitoring is characterised as ‘on line’ and potentially component specific in it’s monitoring application. Both approaches have their advantages and disadvantages and subsequent demands on system design in terms of cost and effectiveness when reflected against the overall business aims of the system in service. It is important therefore that these distinctions are defined further so that their impact on design and business issues is understood at this mitigation stage. A highly predictive monitoring service could mean increased data transfer and storage costs for example. There are design consequences in this approach for both the Hard System (the data acquisition function will need the capability to transfer high rates of data and store them) and the Soft System (data retention,
cataloguing and administration become issues which may require greater resources).

5.8.2 Predictive and Diagnostic monitoring systems used in mitigation

Before developing these monitoring Hard System Reliability Risk Mitigation Views, a position has to be defined with regard to these types of monitoring system for the purposes of this work.

With regard to the two main types of system monitoring discussed above:

- Predictive monitoring;
- Diagnostic monitoring.

Predictive modelling in this case is defined as the ability to test Hard System behaviour defined by a number of sensors against a model of predicted behaviour of that system or some algorithm that defines failure circumstance. A wide variety of sensors could be used in this type of monitoring which are not necessarily functionally related and are likely to be used in combination to perform non real time trend analysis. For example vibration monitoring sensors and oil debris sensors could be used to detect and confirm bearing failure by separate means.

It can also be reasonably assumed that this type of mitigation may also become useful when the Hard System comes into service if unpredicted failure events emerge as the opportunity to add dedicated monitoring systems to such an emerging failure mode will be limited and costly if implemented.

Diagnostic modelling on the other hand is defined as those systems which perform in situ threshold measurement checks by a sensor or sensors dedicated to determining either a failed condition or to establishing conditions of incipient failure.

Both methods are used as complementary monitoring strategies towards risk mitigation of system availability. When describing the monitoring system
therefore both approaches need to be taken into account and the sensors used in each mitigation approach have to be put in context in terms of:

- Their importance in design terms to the success of the overall solution;
- Their importance in cost terms of the overall solution.

In order to do this, a position has to be developed regarding monitoring system design philosophy.

### 5.8.3 Monitoring system design philosophy (Predictive monitoring)

With predictive monitoring it can be assumed that the components being monitored by this method will be components that are not monitored by a dedicated sensor. This is because this type of monitoring is usually associated with trend monitoring of parametric data over varying periods of time (depending on the component failure mechanism) from control systems sensors that monitor system performance. This is effectively an indirect use of standard component information and as such is predominately a ‘virtual world’ activity.

This ‘virtual world’ activity is a Soft System activity in terms of this research but relies on ‘real world’, Hard System sensors. The ‘virtual world’ techniques and algorithms that interpret that data are not modelled in the Hard System as the techniques potentially require data acquisition over a long period of time in addition to computationally intensive algorithms which are not capable of running in real time. The Soft System component of this activity will not describe such algorithms but instead will relate to the time taken to perform the computation activity and the necessary support required by expert operators in order to examine and make judgements on the collected data. They are therefore considered as an artefact of the operational effectiveness of the Soft System.

This type of monitoring scenario suggests three distinct levels of hierarchy for this type of approach.
They are considered to be:

- Level 1 - monitored components;
- Level 2 – sensor components;
- Level 3 – data collection, storage and manipulation components.

Note that in this case, hierarchy level 3 is likely to be represented by both Hard System artefacts and Soft System artefacts (depending upon the system design) as some form of trend monitoring implies necessary Soft System support.

5.8.4 Monitoring system philosophy (Diagnostic monitoring)

A diagnostic system is seen to apply direct and dedicated measurement of components, in order to detect specific limit exceedances or conditions of failure.

The same level definitions described in section 5.8.3, however, apply to this type of system. The main difference is in the extent of the boundary of level 3.

For a diagnostic system, it is unlikely that the level 3 boundary will be set to extend beyond the Hard System being monitored, as a diagnostic system will act in real time and therefore will need to be self contained in order to provide system failure warnings once set limits or limiting conditions are met by the operation of the system.

As such these systems are predominately ‘real world’ systems with little or no ‘virtual world’ interactions.

5.8.5 Monitoring System hierarchy

In order to provide further differentiation, the components that form the Hard System Reliability Risk View are separated out into the components that are being monitored and those that perform some form of monitoring function.

The method of separation is that of the boundary level hierarchy discussed in section 5.8.3. This is done in order to develop the views of reliability monitoring mitigation that are required. Once separated in this way, a clearer view of which
sensors are available for the mitigation strategy in each particular interface is provided and also a view of which components are to be monitored.

The advantage of delineating the Hard System Reliability Risk View using these conceptual boundaries enables a partitioning of the design and therefore a clearer view of which sensors are involved in the reliability monitoring system, which of those are most important in the mitigation and which components are and are not monitored so that this lack of monitoring can be assessed.

The main issues for Hard System Reliability Risk Mitigation View are:

1) Does the monitoring system provide both predictive and diagnostic coverage appropriate to support the system in the field and therefore provide the expected business case benefit?

2) Does the monitoring system provide reliable and repeatable failure prediction diagnosis for all those components being monitored?

3) What level of data acquisition and manipulation is required to support the above?

Taking into account these issues, we can advocate the following definitions of reliability monitoring system hierarchy level:

**Monitoring System Level 1** – monitored components, provides a view whereby an assessment can be made as to the appropriateness of the monitoring system coverage.

**Monitoring System Level 2** – sensor components, provides a view of what failure modes are capable of being detected and can also lead to an assessment of the level of data transfer that will be required (this would need to be done in conjunction with Monitoring System Level 3 assessment of acquisition rates).

**Monitoring System Level 3** - data collection, storage and manipulation components. An assessment at this level is not possible without the development of monitoring scenarios at Monitoring System Levels 1 and 2 as these levels provide the requirements for this level. It should be noted that the Monitoring System Level 3 boundary extends beyond the immediate system
being monitored boundary to include ‘off line’ predictive systems which are described in part by the Soft System model.

In developing the above Hard System ‘Monitoring System Levels’, the vulnerable components identified by the process listed above are placed in a contextual relationship with each other in a reliability monitoring system sense.

This differentiation is important in assessing the system, as identifying vulnerable components alone is not enough to provide a view of overall system vulnerability. For instance, some ‘vulnerable’ components may be used for monitoring or (if part of a distributed system) may be able to fault diagnose themselves.

5.9 Developing Hard System Reliability Risk Mitigation Views

In order to provide mitigation against Hard System component vulnerability, some kind of reliability monitoring system design solution has to be documented and examined. As discussed in section 1.2, there are a number of methods used to mitigate reliability issues, which can involve scheduled component replacement and other methods.

In this research, the argument is that to increase system availability some form of reliability monitoring system will be needed to provide that mitigation. This requires that subsets of the Hard System Reliability Risk View be transformed to provide an appreciation of the how both predictive and diagnostic monitoring is to be applied to form that mitigation.

5.9.1 Developing diagnostic monitoring Hard System Reliability Risk Mitigation Views

In order to produce this type of view, both of the following are required:

- An understanding of the failure mode (s) to be detected;
- An understanding of System Design.

The system design is implicit in the Hard System Reliability Risk View and so development of this mitigation must be done in conjunction with a system design expert in order to provide an explicit commentary on system design features. If
the Hard System Reliability Risk View has been developed through modelling the
design (as shown in section 3.3) the design implications are more apparent.

The mitigation views show the system components that are, through system
design, intended to monitor particular sub-system or functional components.
Such a view may cover several or all failure modes of the sub-system or
component being monitored. A mitigation view should be produced for each
particular monitoring design approach. Such a mitigation approach could be
considered to already be part of the system specification as diagnostic systems
are usually in place to provide an immediate assessment of the system health
status for safety considerations. These should also be modelled for
completeness.

It is important to be specific regarding the use of this type of mitigation case as
sensors used for these mitigation cases are likely to have a duplicate use for
system control or as an additional source for longer term trending data in
prognostic monitoring cases, in which case they may have conflicting
requirements placed on them. For example, one mitigation case may require the
sensor to have a greater accuracy requirement than another. Recognising and
resolving issues such as this is necessary in providing an optimised and cost
effective design offering.

It is also important to show Monitoring System Levels on these mitigation views.
This is required so that the definitions given earlier can be observed. These
differentiations leave no doubt as to the function of the component in a reliability
sense and provide a relative view of its capability with respect to a reliability
monitoring system capability (for example whether there is a potential for its
function to be used to corroborate failure in another component).

An example of a diagnostic monitoring Hard System Reliability Risk Mitigation
View, involving sensors that will be used to detect engine bearing system failure
is given by Figure 5.9.1.
In Figure 5.9.1, the view is divided up into monitoring levels, this segregates those components being monitored from those components likely to do the component monitoring (principally these are sensors).

It can be seen from Figure 5.9.1 that there are a number of oil washed components that will need to be monitored for reliability issues (Monitoring System Level 1 components). Figure 5.9.1 proposes that each of the identified Monitoring System Level 2 components could be used to diagnostically monitor failures in those specific System Level 1 components.

Note that Figure 5.9.1 is at this stage intended to show all components involved in diagnostic monitoring in general. There is however the option to show specific diagnostic monitoring relationships which target specific failure modes to show how maximum monitoring system coverage is achieved with the components identified. For example a view could be developed which associates the following Monitoring Level 1 components:

- Drive Rod (bearings);
- Peripheral GBox module (bearings);
- External GBox (bearings).

To the following Monitoring Level 2 components:

- Analysers (Vibration Monitoring Unit);
- Analysers (Vibration transducer).

With the aim of suggesting that all bearing failure modes could be diagnosed using the identified Monitoring Level 2 components.

It may be that other Monitoring Level 2 components will either do the same or complement this method of detection, but making a definite statement of aim helps the designer of the reliability monitoring system as well as the designer of the Hard System being monitored.

The key point is that all such suggested diagnostic monitoring strategies should be aggregated into a view as shown by Figure 5.9.1.
Figure 5.9.1 A diagnostic monitoring mitigation view covering engine bearing system failure
5.9.2 Developing a predictive monitoring Hard System Reliability Risk Mitigation View

Where the diagnostic mitigation views are biased toward describing components monitored and monitoring the system directly, on line and in real time, predictive system mitigation views are biased toward off line, long term analysis techniques.

The approach however is the same in that the components being monitored and the respective sensors whose data is used to perform the long term monitoring are recorded on the same diagram for each component or set of components that is likely to be used in detecting a failure. Such views may identify sensors used in diagnostic systems as sensors that are also important in a predictive technique. Identifying such a situation is important for both the system designer and the reliability expert as the importance of the sensor function in terms of the design may need to be re-evaluated as discussed in section 5.9.1.

Again the diagram is drawn in a similar way to that above for diagnostic mitigation views, and again the Monitoring System Level is shown. It should be remembered however that the predictive system (given the definition of such systems earlier) also relies on attributes of the Soft System model in order to be truly effective and therefore although this figure is a representation of the Hard System aspects of the predictive monitoring approach required it does not describe the predictive monitoring system in total.

Note that no Monitoring System Level 3 components are shown here in this example. If there were any Level 3 components, they would in this case be in the form of a data acquisition and transmission unit. If that unit were to be directly connected to these sensors, then it would be shown as a module on this view as it has a failure rate attributable to it as a piece of hardware in its own right. It would also appear on any Soft System Reliability Risk Mitigation View as the start point of that mitigation (assuming that it transmits data to the Soft System to begin Analysis). In the Soft System, the hardware failure rate is not taken into account, but there is a reliability component in terms of processing and
data transmission time that will form part of the Soft System analysis. The level 3 components will always exhibit this duality in this assessment technique.

An example of a predictive monitoring Hard System Reliability Risk Mitigation View which could be used to predict high engine vibration is given in Figure 5.9.2. The approach here is much the same as that adopted in developing a diagnostic monitoring Hard System Reliability Risk Mitigation View. The intention is to show which components are involved in a predictive monitoring context for the particular components under review. In this case, the same components in Figure 5.9.2 are chosen as used in the example discussed in section 5.9.1. In this case however the intention is that these components are used to trend monitor recorded values in order to warn of impending failure rather than diagnose the specific cause.
Figure 5.9.2 A predictive monitoring Hard System Reliability Risk Mitigation View used to predict high engine vibration
5.9.3 Developing a Summary Hard System Reliability Risk Mitigation View

In order to provide a holistic view of the vulnerability of the Hard System to reliability issues it is important to gather together the reliability monitoring system mitigations and project them onto a single view in order to provide a Summary Hard System Reliability Risk Mitigation View to ensure that all of the various mitigation views are captured and their respective effects are understood.

In this view, the Hard System Reliability Risk View for each interface type is taken and then the components from each mitigation view are recorded on the Hard System Reliability Risk View by colouring or shading the component box involved in the mitigation. If there is no mitigation against a particular component, then that component retains its status on the Hard System Reliability Risk View. That is, the FMEA colour coding still applies.

Components involved in the reliability monitoring system retain their status from the Hard System Reliability Risk View also.

This Summary Hard System Reliability Risk Mitigation View then shows which components are monitored and those that are not, meaning that they are still at risk.

Those specific components that are identified as vulnerable and which do not have appropriate sensors available to provide a mitigating failure detection capability, leave various options open to the designer and the analyst, they are:

- If the component has a long MTTR, it may not be necessary to develop a monitoring capability;
- If the component has a short MTTR, some form of monitoring strategy is required;
- If it is not possible that a monitoring strategy can be put in place for a short MTTR component then a redesign is required or some form of change to the business case will need to be negotiated.
For those components where a monitoring strategy is possible, for any particular Hard System platform function or Hard System interface as a whole there may be a number of complementary design solutions, which form part of this mitigation. If these are identified in the early stages of system development they can strengthen the business case by:

- Optimisation and integration where possible of the complementary methods;
- Providing corroboration of a system failure in order to avoid false alerting to failure conditions;
- Using different systems to detect the progression and confirmation of a particular failure as non electrical failure mechanisms usually occur over a time period which enables precursor failure events or symptoms to be detected.

The resulting view is shown in Figure 5.9.3 and provides a summary view of all components involved in mitigation of reliability issues for that Hard System interface type considering only the diagnostic and predictive views developed earlier. Importantly, this view also shows the vulnerability of those components in terms provided by the FMEA. It should be remembered however when referring to this view that even though a component or sub-system is monitored, not all of its failure modes may be monitored by the monitoring system.

The following information can be drawn from this view:

1. There are a number of Monitoring Level System 1 components that either cannot be monitored for reliability or are not as yet covered by the monitoring system;
   a. These should be noted and examined further to see whether their failure could be monitored.
2. Those that for whatever reason cannot be monitored will need to be further assessed;
   a. To determine the likely disruption to availability should they fail in service;
   b. To determine an appropriate maintenance strategy best suited to avoid availability disruption using current techniques such as those listed in section 1.2.

3. Those that can be monitored must be further assessed in order to understand whether the monitoring system can detect all of their identified failure modes;
   a. if not, then an investigation should be carried out to understand the effect of this undetectable failure mode on availability. If this is a serious effect, then an appropriate maintenance strategy should be determined as suggested in 2b above.

4. A record should be made of the monitoring system components along with the monitoring strategies that they are used in, in order to inform the Hard System designer of their uses so that if required the specification for these components can better reflect their use or if necessary, be duplicated to ensure redundancy;

5. Finally a clear view of the capability of the monitoring system should be provided to those developing the contractual arrangements of the Servitisation business case to ensure that they are aware of the risks to availability by lack of monitoring coverage or any other pertinent issues highlighted by the points in this section.
Figure 5.9.3 A Summary Hard System Reliability Risk Mitigation View of all components involved in mitigation of reliability issues for the oil system interface type
5.10 Hard systems modelling technique summary

The purpose of Hard Systems modelling is to develop a view to determine:

1) Which components provide reliability risks to the system;

2) What mitigation there is against each components reliability risk by use of a monitoring function;

3) What sensors, techniques and functions are used to monitor particular components;

4) What is the importance of the system sensors used in the monitoring function (do the sensors have multiple uses).

These aims are met by using the FMEA as source documentation from which to model the Hard System. Design documentation could also be used. Using the FMEA, in addition to providing a design and function context, also defines components susceptibility to failure and the consequences of that failure to produce a view of system risk to reliability issues.

The FMEA information must be partitioned such that components are assigned to their respective system interfaces, system platform functions (the modules or subsystems relating to that interface) and the components in those modules that relate to that interface.

The Hard System Reliability Risk View is produced for each Hard System interface which groups these modules together. The system interfaces can be either:

1) Electrical (and software);

2) Oil;

3) Pneumatic;

4) Mechanical;

5) Hydraulic;

6) Any other type of fluid interface particular to the system (such as fuel).
If a module component belongs to more than one interface type (such as a mass flow meter), the component must appear both in the electrical system interface (if the mass flow meter count is made electrically) and the fluid system interface (the fluid that the mass flow meter is measuring).

The Hard System Reliability Risk View also shows the Performance Effect Criticality obtained from the FMEA for each component in the system interface. Such a view provides a graphical interpretation of FMEA findings. It also provides a view of the system interface components against which reliability monitoring mitigation strategies have to be designed and of the sensors that are available to each system interface to perform that monitoring activity.

Mitigation strategies are formed using Hard System Reliability Risk Mitigation Views which are similar to and based on the Hard Systems Reliability Risk View for each system interface. These mitigation strategies are a number of designs which show how each component or module is to be monitored and how that monitoring is likely to take place (diagnostic or predictive). These diagrams include the sensors that are to be used to perform the monitoring.

These Hard System Reliability Risk Mitigation Views are then summated and a Summary Hard System Reliability Risk Mitigation View is produced. This also shows the defined Monitoring System Levels in order to provide an indication of which components in the system interface are:

- Involved in monitoring;
- Being monitored;
- Not monitored.

Such a view provides the system designer and reliability expert with an appreciation of which monitoring sensors are important (and whether they are used in a number of monitoring functions) and which components (some of which may be critical to system reliability) have no monitoring strategy against them.

Such a view will also inform the debate regarding whether or not either a system redesign is required or, if there is no suitable system design solution mitigation, a
modification of the business case and proposal for service support provision of the equipment in operation is required.

It should also be noted that if the FMEA is used as the main source document for this modelling approach, only vulnerable components identified by that document will be considered. It should also be noted that even those components to which a monitoring strategy is applied may not have all of their failure modes identified for detection and that failure detection will depend on the appropriate data being available to the monitoring strategy that is being applied.
Chapter 6 - Application to more complex Soft Systems

6.1 Introduction

In this chapter the ideas discussed in earlier chapters are extended to show how they would be applied to Soft Systems which are complex in nature, larger in scale than those previously considered or existing systems that have been developed and established over a long period of time and now need to change to support a new Servitisation business model.

These systems are likely to:

- Have evolved to fulfil a specific role adjacent to what is required but are at present disparate and require to be integrated more effectively;
- Be systems which need to be designed without preconception to meet a change in business operation;
- Be a combination of these factors.

For the Soft System, the expanded modelling approach identified in Chapter 4.4.4 will be used. The basis of these models is the quality system documentation which is normally only audited for compliance to the required quality standards such as ISO 9001:2008 [46]. These are the building blocks of the complex Soft System and as such may not have been developed as an integrated set of procedures. These circumstances may arise because of departmental boundaries or even differences in approach between those who draft individual procedures.

The modelling approach proposed allows these procedures to be viewed holistically. Once this holistic view is established, the Soft System is open to a variety of performance related measurement techniques for analysis such as CMMI and other techniques as listed in Section 1.7.2.
With respect to analysis necessary for this research the primary aim is to establish a response time. Soft System performance will affect the response time but in the first instance the aim is to understand all the systemic factors which contribute to the established response time without thought of improvement but merely to provide a baseline understanding of Soft System capability.

There is a key difference between Hard and Soft Systems as the Hard System is often designed with key, measurable objectives which form the basis for its existence but Soft Systems intended to support Hard Systems may have been developed to support a variety of legitimate business aims, many of which may be intangible or at least not easily expressed as a performance objective. Defining a specific and measurable aim for the Soft System links this aim and the future development of the system with the Hard System it is meant to support. It is the effectiveness of this type of Soft System that this research is aimed at identifying.

The Soft Systems modelling approach requires flexibility in the interpretation of ideas developed previously to match with the constraints of the application provided.

### 6.2 Considerations in developing support mechanisms for Monitoring Systems for established or evolving complex designs

With a monitoring system that provides a commercial service, the reliability and efficiency of the physical equipment or Hard System that performs the monitoring function is not the sole consideration. When dealing both with diagnostic systems, which indicate when a system function has failed and with predictive systems which require non real time analysis techniques to be applied, Soft System support mechanisms are invoked in response to functional failures detected by these systems. These support mechanisms enable the acquisition of reliability monitoring system data, analysis and storage of that data and advice
on how to act upon the analysis findings to maintain availability of the Hard System being monitored.

It is possible that such Soft System support mechanisms (unlike the one proposed in the simple system example discussed in section 3.5) will require numerous organisational interactions to perform these functions. For instance many monitoring algorithms (especially those which trend operational data) will be Soft System centric as the amount of data and the frequency at which that data is analysed will be by definition at a slower rate than monitoring systems which are intended to detect real time events. The reliability of this data intensive and human interactive support system of the monitoring function is as important as the reliability of the Hard System that is being monitored.

System reliability for both Soft and Hard Systems has the same meaning in that both systems must perform reliably. It is acknowledged however that there are specific differences in the ways that each system might fail, but the aim of the assessment is not to determine these individual factors and their causes at such an early stage, that is the function of later analysis.

It should be noted in addition that some aspects of the function of the Soft System which may affect the reliability of that function are independent of the design and assessment of the system and as such may only arise when the system is in operation. These specific issues relating to Soft System reliability are outlined as follows:

1) Electronic sensing equipment may produce ‘false’ or incomplete data which will need to be reported to the operator and potentially ‘reconstructed’ by analysts;

2) Mechanical systems fail predictably in the main but the rate of that failure is due to the duty cycle of that equipment, making failure trend analysis on a daily, weekly and monthly basis rather than in ‘real time’ an important factor in determining reliability and availability;
3) The system as a whole may fail in a way not experienced previously due to unique system integration or unexpected operational issues, requiring Soft System support mechanism expertise to be available to analyse and bring engineering judgement to such instances;

4) The Soft System may not fail in physical terms but may fail to meet its commercial guarantees. Such an instance can jeopardise the commercial business case when implemented;

5) The quality and availability of data necessary to analyse the condition of the system may not be sufficient to make either a full prognosis or diagnosis. Expert knowledge both in failure behaviour from past systems and from when the system was in development will need to be captured and included into the Soft System support mechanism;

6) In the initial design phases a monitoring system may be under development which cannot provide a fully supported response (including a scheduled maintenance action) to the event it is supposed to detect in time to ensure the availability of the equipment being monitored;

7) A dialogue between the operator and the service provider will be required in order to help with reliability diagnosis (unless the service provider is fully supporting the operator in the field). The efficiency of this process will have a direct impact on providing a rapid solution to problems and will indirectly help to develop the reliability of the Soft System as experience in failure mechanisms grows.

Each of the issues raised above requires full or partial human interaction to be fully resolved. Understanding the level of human involvement and the process of that involvement is crucial. Without this understanding (and subsequent analysis of the effect of that involvement) an expensively designed and developed reliability monitoring system can be limited in its potential and could consequently lead to failure to meet its contractual obligations.
To mitigate these issues the organisation that provides the support mechanism to the reliability monitoring function has to be optimised in order to fulfil the commercial service obligations of the system as a whole. This organisational optimisation is as critical to the business case as the bespoke reliability monitoring system. To assume that it is not, is adding considerably to the risk of the business case in fulfilling its potential. Therefore a ‘Soft System’ model of all aspects of the support mechanism has to be developed and analysed.

Issues relating to data are also referenced in the 7 factors discussed earlier in this section, but have not been expanded on. This is because organisational and data issues are considered here to be inextricably linked. In order to enable an efficient information flow around an organisation, the organisation should be structured to allow that flow regardless of any arbitrary control mechanisms that may be placed upon either the information or the organisation as speed or response to that data is of the essence.

There are however, specific attributes of data that need to be addressed as they have a direct impact on the reliability and efficiency of the system performance as a whole and they are:

- **Data availability;**
  - In terms of quantity.
  - In terms of speed of transmission.

- **Data quality;**
  - In terms of appropriateness (to perform the required function).
  - In terms of correctness.

- **Data type;**
  - Electronic transmission (rate of transmission, mode of transmission).
  - Manual collection/interpretation of data or input of data.
These aspects should be considered to be subordinate to the organisational structure as this structure provides the framework on which the basis of this analysis can take place. In the majority of cases an analysis of the Soft System will by definition also take account of these factors as the availability, quality and means of transmission will have a quantifiable impact on the robustness and efficiency of the Soft System.

6.3 Source documentation

In all modern large companies, the quality system is a key part of documenting the processes of that business. These processes document how employees are supposed to behave in order to fulfil their purpose in the organisation.

It is suggested that such documentation, if correctly modelled, will effectively represent the organisational system response to inputs from monitoring functions in the field. It is acknowledged that this documentation may already exist but will be used for audit purposes of the quality system only and may not be in a form that lends itself to modelling both intra and inter departmental functions (an example of this would be a flowchart of a process that described the steps required to fulfil a function in a department). Such a ‘model’ has the potential to be of a great size for only a relatively small function, and the opportunity to cross link such models in an integrated manner is therefore minimal.

Indeed, as these documents are often used for audit purposes, it is suggested that sometimes it could be beneficial to provide as abstract a view as possible of the functional description to provide greater leeway in passing an audit. Such an approach, though understandable, is not acceptable for modelling purposes where greater detail (to a qualified level) is required. This in turn should provide greater understanding of the actual tasks that must be performed to fulfil the function. This greater understanding, will lead to a more detailed and accurate analysis of the process.
If this type of documentation is not available as a modelling source, any Gantt chart which describes the support function of the equipment in the field could also be used as a modelling source in the same manner as that in section 3.5.

6.4 Additional Soft System support mechanism design features

Any optimisation or additions to the support mechanism is likely to be derived mainly through the development of extra and inter departmental interfaces or organisational remodelling following an analysis of the capabilities and limitations that such interfaces impose on the structure as a whole. A developed model of the support mechanism should be used as the basis for analysis such that it becomes a means of system integration. System improvements can also be developed from it in much the same way as a reliability assessment of a Hard System may drive re-evaluation of Hard System design.

6.5 Soft System support mechanism risk mitigation

Mitigation against the Soft System support mechanism model is substantially different to the mitigation proposed for the reliability monitoring Hard System model.

With Hard System mitigation the aim is to assess the exposure of the system as a whole and to develop new or complementary strategies to offset this exposure, and to gain an understanding of system vulnerability to unmonitored failure modes. This mitigation is relatively easy to quantify in both design cost and commercial terms.

In the case of Soft System mitigation however, the situation is more complex as organisational change has factors which relate to human behaviour in addition to task performance. Leaving these cultural issues aside, the purpose of this mitigation should be to develop an idealised organisational framework which would deliver the required response reliably enough to support the business case.
Mitigation of Soft System risks will therefore require an understanding of organisational complexity and the character of communication interfaces that support that structure and a consequent reduction of risk associated with these factors based on that understanding.

For these risks to be determined, the Soft System support mechanism must be modelled in such a way that these issues are both made apparent and are shown in context within the current organisational structure. The reliability of the Soft System will be based on the time in which a response to an external event from the Hard System reliability monitoring system takes to enact based on:

1) Time taken to complete process steps or tasks;
2) Time taken to enact functional interfaces;
3) The nature of the functional interfaces.

Although the first point is self explanatory with regard to an understanding of the process, the other points require further explanation.

A functional interface here is taken to mean a function to function interaction (such as passing data from a data source to a data analyst). The nature of the interface is the method of communication (for example is the data sent over a data communications network or is it sent on a disk through the post from source to destination). The nature of the interface is also an aspect of Hard System design, which must be considered in tandem.

As with Hard Systems the Soft Systems mitigation could ultimately take the form of a Soft System redesign. Equally it could force design changes to either the Hard System or Soft System or both in order to meet a holistic requirement of performance for the business case to be successful overall.

6.6 Summary approach required

The Soft System support mechanism acts in response to the reliability monitoring system detection of Hard System failures. It is the purpose of the Soft System to anticipate and structure appropriate maintenance activity in order to reduce
operational disruption in response to these detected failures. Modelling the Soft Systems that collectively achieve this aim therefore requires:

- An understanding of the organisational framework that supports this activity;
- An understanding of the way the framework is enacted to support equipment in the field;
- An understanding of the factors inherent in that framework that may lead to disruption.

Developing this understanding requires that the Soft System business processes are modelled. There are many techniques that are used to model business processes and all have their particular advantages and disadvantages. Such modelling however is unlikely to completely model the Soft System as the processes are enacted by humans, so factors such as cultural values, motivational and management issues all impinge on such a model, but cannot be easily modelled, nor can their effect be accurately quantified because of these indirect and more intangible effects.

It is impossible however to fully appreciate the potential benefit of a reliability monitoring system business case without developing at least a rudimentary appreciation of the Soft System support mechanism processes. Any model of the Soft System must therefore not be considered to be complete and accurate because of these factors.

It is suggested however that the primary purpose of this modelling and subsequent analysis of that model will be to:

1) Provide an overview of the Soft System and its points of vulnerability;
2) Provide the response time to reliability monitoring system inputs;
3) Provide an estimation of the repeatability or consistency of that response time.
The Soft System model developed, despite the constraints listed above, should ensure a model of system operation capable of providing an approximate answer to these questions.

6.7 Soft Systems modelling example

It is reasonable to ask the question why Soft Systems modelling should take place in the first place as, similar to the Hard System modelling approach, the information required to make an assessment of the system already exists. As with the Hard System however, the information is structured in such a way that makes it hard to use to initially make this assessment because the view the information provides primarily supports other objectives. It is also highly probable that the information is not integrated between various company departments and some form of integration (or in some cases where duplication of effort is described) rationalisation exercise will need to be carried out before a complete view of the Soft System from existing documentation can be obtained.

There is also a question of granularity when describing the process. Too detailed a level of granularity and the Soft System model becomes very proscriptive, too abstract and the process is difficult to understand and accountability for process delivery is often vague. The technique used to model the Soft System must therefore highlight and be capable of dealing with such problems.

6.7.1 Process flow charts review

The process flow chart (Figure 6.7.1) used as an example to describe a part of the Soft System that is to be modelled takes the form of a computer flow chart with the added aspects of ‘who’ performs the task described in the corresponding flow chart box and ‘how’ that task should be performed.

Such an approach describes linear processes, which break occasionally on a condition being met to either ‘go forward’ to a point in the task whereby, for example, either the process stops or previous activities are repeated. Such process descriptions are self-contained and tend to be lengthy on occasion. It
is also difficult to integrate this type of process description with other processes described in the same manner.

The overall effect of these constraints is that it is difficult to assess the process needed to support equipment reliability alerts produced by the equipment monitoring system, especially if the process (as it is most likely to do in this application case) is carried across departmental and organisational boundaries.

6.7.2 Description of the example process

The flow chart describes what should happen when a Hard System failure is detected in service. The process shows three sets of analysis procedures, each requiring more specialism to be applied to the data analysis if the previous level cannot provide fault condition confirmation. This increased examination effort may be required because of a lack of data, because such a fault condition has not been seen before, or if a fault condition is inadequately defined.

If the fault condition is determined at any stage during this process, the result is communicated to those who need the information and that part of the process then stops as control is handed over to another process which deals with how to use that information.

The process described therefore can ‘complete’ in a number of ways each with an associated ‘time to completion’. Each of these processes and its ‘time to completion is valid’ and so each has to be modelled for assessment purposes.

Clearly, the flow chart does not describe all of the processes required to get data from the equipment being monitored, assess that data appropriately and develop a maintenance schedule with the operator, it is just one part of the Soft System support mechanism.
6.7.3 Process integration and rationalisation

It is possible that all of the processes that relate to the one shown in Figure 6.7.1 exist in other departmental documentation (as an example Figure 6.7.1 indicates...
a route to ‘process 2.3.3’ which is applied when the relevant company expert cannot explain the event to an adequate level. Note process 2.3.3 is referenced but not detailed further) but it is equally likely that they do not, or that they appear as processes written at another level of abstraction. It is unlikely that they are integrated. This is because such a process typically involves not only other departments within the Servitisation company (which have their own discrete processes) but also the equipment operator, which will also have dedicated company processes.

In such cases an ‘idealised’ integrated Soft System process has to be developed in order for an analysis to be performed using if necessary, a ‘black box’ approach in cases where the details of the processes within the ‘black box’ are unknown. Such an approach is very likely in an actual application of this technique as company processes are commercial in confidence, and so it is unlikely that the operator would give the Servitisation company access to their processes, especially if that information were to be used in order to develop a business case for the service to be provided.

It is also unlikely that an integrated Soft System process will be developed without management or process specialist involvement as an understanding of organisational and process issues will be of importance in developing the integrated view. The aim regardless of organisational and other such barriers should be to describe the Soft System support mechanisms fully and that the flow of the process of that Soft System should operate in the most efficient way in terms of time taken to accomplish the tasks that comprise the Soft System.

An example of the kind of rationalisation activity that may be required to take place is as follows. Taking Figure 6.7.1 as an example, it can be seen that this departmental process mitigates against both levels of operator expertise (if the data, and the failure event that the data describes is well known, the consequences can be defined and understood by staff with a relatively low level of expertise. If the problem is more complex, then the process passes the problem to more highly skilled staff.
This is a sensible approach which is easily understood by all and will deliver an appropriate resolution to the problem set. There are however, a number of problems with such an approach with regard to developing an understanding of how the Soft System support mechanism operates. Failing to understand such implications damages the business case not only for the proposed service but also for the reliability monitoring system that delivers the data.

Firstly, there is no idea of what level of data is required to provide a full understanding of the problem. Secondly, there are no timings involved in this process to suggest when the problem should be escalated. Thirdly, there is no idea of how the operation of the people in this process impacts on other people in the process who are helping to provide a service to the customer. According to this process, the customer can be ignorant of any problem solution until the relevant company expert has delivered a verdict after an indeterminate analysis time period.

Such a piecemeal approach, whilst addressing a specific engineering activity problem ably and appropriately (that is, how do we determine a reliability condition of concern), does not address the real issue which is, as a business, if we have a reliability condition how do we schedule a maintenance action with the customer such that there is no reduction in operator equipment unavailability.

Figure 6.7.2 shows how the flowchart shown in figure 6.7.1 (which describes primarily the engineering reaction) could be developed to show a view which is integrated with a business response activity to deliver an idealised business response to a detected failure. It is this response that must be assessed to see whether it meets the business case, not simply an engineering response to a failure.

The steps required to perform this development from the flowchart shown in figure 6.7.1 to the modified Role Activity Diagram are as follows:

First, as outlined in section 3.6.1, define the Soft System Boundaries. These are provided by the flowchart as those referenced in the 'who' column. In this example:
• The ‘service provider first line analysts’ in the flow chart become the ‘service provider operations management’;
• The ‘specialist team’ become ‘service provider Health Monitoring Function’;
• The ‘relevant company expert’ becomes ‘service provider specialist function’

Then, as outlined in section 3.6.2, tasks are associated with each system boundary. In this particular case, figure 6.7.1 shows 3 main tasks all of which are ‘data is analysed’. Two of these tasks are interpreted in Figure 6.7.2 as Task T221 and T222. The first data analysis point is assumed to have stimulated comms path C221.

Finally as in section 3.6.3 the system boundaries are connected. An attempt is made to show the process flow through the system boundaries in order of precedence, starting with first line analysis and ending with the response from the specialist function.

The ‘first line analysis’ is carried out by comms. paths C221 and C223 with Task T221. The specialist team analysis is carried out using comms. paths C221, C222 and C224 and Tasks T221 and T222. These elements are the only ones that bear any relationship with the flowchart process. Merely by plotting the flowchart using this method it is seen that not all of the tasks that are being carried out are necessarily reflected by the flowchart and other tasks and communication paths are added in where necessary to provide a fuller picture of the actual function. These other elements are brought in to suggest tasks that must be accomplished by the business to provide:

• Response management (via the Service provider operations management Soft System boundary function);
• Ensure comparison of existing and new failure characteristics is carried out for knowledge capture (T244 analysis against the equipment fleet);
• Ensure knowledge capture of each event is made through the Service provider operations data centre Soft System boundary function.
This is by no means an exhaustive list and is provided only as an example of what might need to be done to develop an understanding of the functions that need to be carried out to support the business case response.

Note that the idealised Soft System support mechanism shown in Figure 6.7.2 includes management of the event issue. Often management type activities are not included in quality procedures as they are not the type of activity that can be easily characterised. Information on how this aspect of the process is managed however is of vital importance as understanding the level of information that is needed for managers to be able to make fast and accurate decisions is a vital aspect in improving the response time to the customer.
Figure 6.7.2 Developing the flow chart into a modified Role Activity Diagram format
6.7.4 Soft System model overview and development

6.7.4.1 Overview of the current model

It can be seen from the example given that there is a great deal of interpretation required when developing Soft System models of support mechanisms. Process flow charts can provide guidance regarding the functioning of the support mechanism, but are often not developed with any thought to integration beyond the specific area of expertise of the function that uses them.

Interpreting piecemeal flow charts such as these is acceptable, but it follows that this is very much a ‘bottom up’ approach to understanding what needs to be analysed (or even developed) to support a business case. Therefore it is important to have an overview of the processes that are required to perform the support function and set in context the roles of those performing the function to aid the Soft System modelling process.

As can be seen from the example given in Figure 6.7.2, there are a number of Soft System boundary functions in any process (these are denoted by the grey boxes). Each of these Soft System boundary functions is owned by a particular Soft System boundary (where a Soft System boundary can be defined either as a person or organisational role, preferably the latter).

It is with these Soft System boundaries that the contextual overview is developed. The Soft System boundaries are first shown on a diagram then their functions are placed in an ‘orbit’ about them. With the diagram in this state, the viewer can see who is involved in the process as a whole and what their particular relationship is. This is important as it allows not only a judgement of who should be involved to be made but also the importance of the contribution of each Soft System boundary to be made qualitatively.

It is important that this is the case as in the first stages of Soft System modelling, there may be interested parties who are not shown on the original diagram and this provides an opportunity for the interested party to challenge this state. It is also true that the importance of each Soft System boundary can be judged by the
number of functions that it has assigned to it. This is useful as a guide for management to ensure that such functions are sufficiently resourced and such a level of recognition is useful at this stage of modelling.

Finally, as with the process shown in Figure 6.7.2, it can be seen that not all Soft System boundaries have a part to play in all of the processes that take place in the Soft System model. It is important therefore to show those Soft System boundaries which are required to communicate with each other. As the method of communication and the speed and control of that communication (in the case of supplier to customer) should be recognised. Such factors may not have an effect on the analysis of the model, but they may affect how the model is designed to ensure that the most efficient communication paths and methods are utilised.

6.7.4.2 Development and expansion of the current model

Figure 6.7.2 documents a process driven by a detected Hard System fault and derived from Figure 6.7.1; it has its Soft System boundaries defined. An expansion of the Soft System model as a whole would have to include inputs to and outputs from the decision process whose boundaries would need to be reconciled with those that already exist. The input to the decision process takes the form of data sent by e-mail (referenced in Figure 6.7.1) and the output from the decision process could for example be considered to be shown by the example flowchart Figure 6.7.3.
Figure 6.7.3 A flowchart showing the response to operator feedback following a fault detection warning delivered by the service provider

Note that the Figure 6.7.3 begins with a response from the operator following a recommendation made by the decision process described by Figure 6.7.1.

From Figure 6.7.2, the decision process defines four Soft System boundaries:

1) Service Provider operations management;
2) Service Provider health monitoring function;
3) Service Provider specialist function;
4) Service Provider operations data centre.
As regards an input to the process defined by Figure 6.7.1 an e-mail from a service operator in the field is referenced. However in this instance and to demonstrate the duality of the interface between Hard Systems and Soft Systems, the data will be assumed to be automatically downloaded from an aircraft engine via the airframe to an operator data centre and from there to a service provider data centre. The new Soft System boundaries for the input to the decision process are therefore:

1) Aircraft engine;
2) Airframe;
3) Operator data centre;
4) Service provider data centre.

The output from the decision process is defined initially by Figure 6.7.3. Here the Soft System boundaries are set to:

1) Operator data centre;
2) Service provider specialist function.

Again in Figure 6.7.3, there is no mention of management of the response from the operator. This should be redressed by adding a further Soft System boundary of ‘Service Provider operations management’. Equally, there is no mention of the Service provider data centre, so this should also be added as we have assumed such a boundary for data input.

The Soft Systems Context Chart can now be constructed using this new information this is shown in Figure 6.7.4 with only the Soft System boundaries defined.
Figure 6.7.4 Soft System Context Chart showing derived Soft System boundaries
6.7.5 Temporal partitioning

As stated in the previous section, analysis of the period that it takes to perform tasks in the process chain is essential to the business case. For example if the reliability monitoring system is developed to determine reliability failures, but if the Soft System support mechanism is not able to confirm the reliability failure and inform the operator in a timely fashion, the system as a whole will have failed.

It is recognised however that timing issues are not easy to resolve especially in large and complex Soft System support mechanisms. It is an aim of this research that the overall process should also be partitioned appropriately in terms not only of Soft System responsibility and tasks but also in terms of Soft System sequencing (the flow of task sequences with time). This partitioning should be made (where possible) where significant breaks in the process occur or where there are specific review ‘gates’ through which the process must pass to continue.

The overall idea of this temporal partitioning however is not only to reduce the Soft System model (and therefore timing analysis) to a more manageable size but also to develop a more structured overall understanding of the purpose of each party engaged at that particular stage of the Soft System model operation.

This approach means determining the order in which the tasks in the Soft System model are performed and bounding them by some means such that only the Soft System boundaries and their functions relating to that particular stage of the model are shown on the Soft System Context Chart.

In this particular case example the major temporal partitions are deemed to be:

1) The data acquisition phase;
2) The data analysis phase;
3) The resolution phase (operator response).
These could be partitioned further depending upon the level of detail and complexity of interfaces between the functions that will deliver the service. The boundary for each phase, however partitioned, should have some definite logical or commercial aspect.

6.7.6 Developing the Soft Systems Context Chart with boundary functions

Once all of the above considerations are taken into account the actual modelling of the Soft System support mechanism can take place. It should be remembered that models and overviews are subject to iterative improvement both in the Soft System model design phase and after analysis has taken place.

6.7.6.1 The data acquisition phase

In the acquisition phase a modelling of the data capture process is made. This involves data transfer from the equipment being monitored (in this case an aircraft engine) to the service provider’s operations data centre, where notionally alerts are handled before being passed on to the service providers operations management team which handles the operation of the aircraft engine fleet for maintenance and other issues relating to the service contract.

6.7.6.2 The data analysis phase

This phase relates mainly to the operations carried out on the data in response to alerts by the service provider. It requires a scheduled data acquisition phase to be completed before it can begin. This phase includes the establishment of processes for specialist functions who will look at data particular to their expertise when it is not easily interpreted as a known failure (as shown in the flow chart in Figure 6.7.1).

We also have to make the following assumptions regarding this phase:

- The reliability monitoring system will produce data relating to known failure diagnostic methods (in a manner defined by Hard System mitigation);
The reliability monitoring system will produce data relating to known failure prediction patterns (in a manner defined by Hard System mitigation);

The reliability monitoring system is likely to produce data which may or may not indicate incipient failures which can only be rationalised by specialist functions.

A Soft System model is therefore required for each of these eventualities. It is likely that the same Soft System boundaries and their functions will play a part in each of these scenarios, but the way in which they act in each will differ according to the scenario type.

6.7.6.3 The resolution phase

The actions in this phase relate the service provider formally with the operator. Depending upon the outcome of the data analysis phase, some form of resolution of how to act upon the alert must be made.

It should be recognised here that in some instances the alert may be of an unknown quantity in reliability terms and so the operator and service provider will have to negotiate what the best approach is. In some instances, this may include the operator ignoring the advice of the service provider and continuing to operate the equipment ‘at risk’ as described in Figure 6.7.3.

Therefore the processes considered in this phase are as follows:

1) Known event response;

2) Unknown event monitor progress;

3) Unknown event ignore service provider advice.

Again a similar suite of Soft System models is required for each of these eventualities as suggested in the previous section.
6.7.6.4 Adding functions to Soft System boundaries

Before further detailed modelling can take place, functions will need to be assigned to each Soft System boundary identified. Initial function placements can be made and then debated and resolved amongst all of those who control Soft System boundaries.

Some functions from the information given so far are simple to determine. It is given that there is an interface between the airframe, Operator data centre and the Service provider data centre for data collection for instance.

Similar deductions can be made for the other Soft System boundaries to determine a Soft System Context Chart such as that shown in Figure 6.7.5 which builds on Figure 6.7.4 and includes interface lines based on the initial belief of shared interfaces. Figure 6.7.5 demonstrates that all those involved in quite complex processes can be shown in context on a single sheet of paper, the detail of these interactions is yet to be defined, but the main participants in that process and their proposed relationship is shown. Temporal partitioning however could be used to simplify the overview further by providing a Soft System Context chart for data acquisition, one for data analysis and one for data resolution.

The rudimentary guidelines for establishing such functions are heavily dependant on the existing organisation and its existing processes.

Departments that have an existing relationship may not have that relationship fully documented, but will know the aim of their joint function and it is this that should be represented in the Soft System boundary in the first instance.

There is also the case to add a function which is desired, but not yet defined. Establishing temporal partitioning is also dependant on existing processes but here guidance is gained from points where there is a natural break in the process, where a management function for example would have to reflect on information received to decide on the next most appropriate course of action.

Another method of temporal partitioning could be based on the level of authority needed to make decisions on the way forward, so the process could be reflected
at those involved at team leader level, those involved at group leader level and those involved at management level.
Figure 6.7.5 Soft Systems Context Chart for a fault detection support mechanism
6.7.7 Soft System Reliability Risk View

Once the Soft System Context Chart is defined, detailed Soft System modelling and the Soft System Reliability Risk View is developed. In the case of the Soft System, primary system reliability risks are suggested to be those of failure of interface between Soft System boundary functions as the tasks that make up the Soft System (like the Hard System components) have identifiable ‘failure’ values (defined by an exceedance of the target time taken to complete) that are more easily quantified and therefore their risks are more easily understood. Unlike Hard Systems however, their interfaces reliability is less easily quantified as it:

- May be carried out in a variety of ways;
- Is not as easily identifiable if it fails – or part fails (as opposed to a Hard System interface failure on an oil system, which could be identified);
- May be accomplished by a number of concurrent tasks.

These interfaces are effectively communication paths between functions and the nature of these communication paths is therefore key to both the efficient and speedy performance of tasks in the respective functions and the underlying reliability of the process as a whole in that if these links break down or provide inaccurate or incomplete information, then the effectiveness of the process as a whole is at risk.

They are also vulnerable if the communication links are ones which are made beyond the boundary of the company organisation. Here, it is possible that there is little control of necessary reliability data transfer from equipment to the service provider, which puts the service offered at risk.

There are two types of interface vulnerability that shall be considered for this research, these are based on whether the interface is an ‘automatic’ or a ‘manual’ interface.

The automatic interface is one whereby data or process control is carried on automatically through a defined electronic transmission of data or control. No
decision has to be made whether or not to send the data or what form the data or control message should take.

Manual interfaces are all those other types of interface, where some form of human interaction is required to take place before data or process control is passed on through the process.

These interface types are chosen as it is accepted that the system is less vulnerable to failure if the linkages and the data and tasks that supply those linkages can be classified and made more automatic in operation. A demonstration of the application of the interface vulnerability and hence the Soft System Reliability Risk View for that specific process is shown in Figure 6.7.6 below which takes as an example application the process shown in Figure 6.7.2.
Provide unknown anomaly data C221

Review data to decide upon specialist function to deal with anomaly type T221

Provide data C222

Analyse data T222

Confirm Analysis against fleet T224

Provide findings for close out C226

Figure 6.7.6 Developing the Soft System Reliability Risk View using the example process given in Figure 6.7.2
Note that such a graphical indication of reliability ‘risk’ acts in the same way as the Hard System representation in identifying areas of the support mechanism that are considered vulnerable in a simplistic manner. No judgement is or should be made regarding the appropriateness of the reliability ‘risk’ here. It may be perfectly acceptable (or unavoidable) to have a part of the system which is vulnerable in service. The object is merely to highlight that fact and if required, provide mitigation against it.

Note also that the tasks in the Soft System Reliability Risk Views are not examined for vulnerability at this stage (but are taken into account eventually in the assessment phase – see Section 7.9) as it is accepted that the tasks need to be performed and can be performed in any way deemed acceptable by the organisation without impact on system vulnerability but not necessarily without impact to system efficiency as the time which each task takes to complete will contribute to the target for response to maintenance events when added to the timings associated with the interfaces. The response to maintenance events as a whole is an important factor to be considered when developing a business case to support Servitisation. Although, such considerations do not provide an obvious and immediate link to the traditional financial cost and benefit analysis of a business case, they do play an important part in deciding whether or not the business model for Servitisation is sustainable. If a serious failure of the Hard System occurs but the response of the Soft System is either not capable of analysing this failure in time for the maintenance response to avoid loss of availability or can only provide a response time which is outside that guaranteed by contractual agreement for response in the business case then the business case and business model as a whole is at risk.

An early indication and thorough understanding of such potential risks to the business case by an understanding of the actual capability of the reliability monitoring system response is therefore an important factor in both supporting and developing the future business case.
It should be remembered that the contractual agreements for response time to maintenance events are driven by the customer in order to ensure the maximum availability of the Hard System for their use. Provided that these customer desires do not infringe certified safety constraints there is an onus on the Servitisation provider to reduce the response time to the minimum. Before Servitisation, this response time would be driven primarily by the maintenance action time. With the advent of Servitisation, there is an opportunity to tailor maintenance actions to suit all parties but an understanding of the time taken to establish the maintenance action required in response to a failure event becomes an important factor. The Soft System Reliability Risk View establishes that minimum response time.

6.7.8 Development of the detailed Soft System model with associated Soft System Reliability Risk Views for the example Soft System support mechanism

6.7.8.1 The data collection phase

This aspect of the Soft System is defined by the Soft System boundaries and functions highlighted with broader, emboldened interaction lines in the Soft System Context Chart Figure 6.7.7. These boundaries are further defined with the function ‘Data collection’ which indicates that there is an understanding that each has a part to play in the Data Collection activity.

Detailed modelling of this function is then performed using the modified Role Activity Diagram notation. Each of the Soft Systems boundary functions is represented by a grey box. The tasks which each Soft System boundary function must carry out are then added. These tasks are time related as the intention of this model is to determine a time for the overall process.

The overall aim of this phase of the process is to download fault detection data from the aircraft engine to the Service provider’s data centre and to pass on any alert information identified.
The initial Soft System boundaries are unusual as there is no human interaction within these boundaries. They are however important as the method of capturing and transmitting data is very important as particularly in this case the asset being monitored is peripatetic. This means that unless the link between the aircraft and the operator's data centre can be bounded, the business case is at risk, mainly due to processes outside the control of the Service provider. This is a business issue as well as an engineering one, but a resolution to the satisfaction of all parties has to be made. The other boundaries are links between the operator and the Service provider, which we can assume already exist in some form, so should be well established.

Figure 6.7.8 shows the passing of the data between the Soft System boundaries. This is also a Soft System Reliability Risk View as the interaction line connector boxes are not white, but coloured green. The interactions represented in this way are automatic in terms of data transfer. That is not to say that there is no associated time delay with each data transfer operation, just that when the data is passed, it will be passed when available and in a defined format.
Figure 6.7.7 Soft System Context Chart for data collection phase
Airframe Data Collection
- Receive sensor input T11
- Buffer, analyse data T12
- Transmit to main airframe data store C11

Airframe Data Collection
- Buffer, analyse data T13
- Provide data to analysts C12

Operator Data Centre Data Collection

Service provider Operations Data Centre
- Process data check for alerts T14
- Provide alert information to service provider operations Management C13

Operator Data Centre Data Collection

Service provider Operations management

Figure 6.7.8 Data collection phase (detailed process)
6.7.8.2 The data analysis phase

There are notionally three separate processes that will need to be defined to deal with data analysis. These all exist at the same ‘temporal’ level and which process is enacted will depend upon the outcome from the data acquisition phase. They are:

1) Known anomaly diagnostic process;
2) Known anomaly predictive process;
3) Unknown anomaly predictive process.

The known anomaly diagnostic process is one where the reliability monitoring system has detected and isolated a known fault. The fault will need to be confirmed and action taken on that confirmation (a maintenance action either immediately or at some defined point in the future – possibly with the co-operation of the operator).

Figure 6.7.9 identifies the Soft System boundary functions that are likely to play a part in this process and Figure 6.7.10 shows the tasks that are required to fulfil that process. Note that there is a management function included here as the effect on the fleet and business has to be considered. Note that other functions such as collection of failure statistics may also be expected to form part of this process. These are largely ignored here as they do not have a direct impact on the service delivery itself (though it is acknowledged that they may have an impact on management time and efficiency).

As with the data collection phase Figure 6.7.10 shows the Soft System Reliability Risk View, again data transfer across boundaries is shown as automatic as the fault condition is a known one and so data regarding this failure condition should be proscribed, reducing confusion in the way in which the data should be interpreted and used.

The known anomaly predictive process is similar to that of the known anomaly diagnostic process in that the failure case and progression of that failure is known and documented. The only difference is that the failure will need to be monitored
to confirm that the failure is progressing as expected in order to determine when a repair action is necessary. The Soft System boundaries that are expected to play a part in this process are highlighted in Figure 6.7.11 and the tasks and detailed interactions to carry out the process are defined in Figure 6.7.12.

Note that there is an interaction here between the operations management boundary and the health monitoring function that confirms the fault before continuing to monitor the progression. The management function may at this point decide to notify the operator of the problem and negotiate a maintenance procedure that optimises availability for the operator. Again, this is not accounted for in the process as this would be a ‘best case’ outcome and would short circuit the monitoring task. The aim of these models is to determine a ‘worst case’ time.

The final process in the data analysis phase is that of the unknown anomaly predictive process. Here a data anomaly has become apparent and the process has to be enacted in order to manage this unknown event. It is assumed in this case that a failure has not occurred, but it is possible that a similar action would take place if a failure had occurred and an investigation was ordered. The only difference would be that in the case of an investigation there would be a different input process route providing actual component failure data and failure findings to appropriate specialists.

The unknown anomaly predictive process Soft System Context chart is shown in Figure 6.7.13 and involves other Soft System boundaries than those previously described. Here there is a need to involve specialists to analyse the data and the management of that task is devolved to the health monitoring function that will need to determine whether or not a fault identification solution can be implemented and lodged as part of the close out of the investigation with the data centre.

Figure 6.7.14 shows the detailed tasks and interactions of the process. Here the Soft System Reliability Risk View shows a number of manual interactions represented by red interaction boxes. There is no defined set of information that has to be passed at these interaction points and the time taken in each case may
vary a great deal. An understanding of the number of these ‘uncontrolled’
interactions and the time that they may take is an important aspect in deciding
the upper most limit of the reaction time to a specific event. That said, the data
anomaly in this case (no failure) is not a critical to the business case as an event
that should have been detected, but was not anticipated in the first instance or
one which is a known failure, but manifests that failure using a never previously
seen before mechanism.
Figure 6.7.9 Soft System Context Chart for known anomaly diagnostic phase
Figure 6.7.10 Soft System Reliability Risk View of the known anomaly Diagnostic Process
Figure 6.7.11 Soft System Context Chart for known anomaly predictive phase
Figure 6.7.12 Soft System Reliability Risk View of the known anomaly predictive process
Figure 6.7.13 Soft System Context Chart for an unknown anomaly predictive phase
Figure 6.7.14 Soft System Reliability Risk View of the unknown anomaly predictive process
6.7.8.3 The resolution phase

There are three resolution phase processes which follow on from the data analysis phase. These are all at the same ‘temporal’ level and are initiated based upon the output from the data analysis phase. They are:

1) A known event response process;

2) An unknown event where the operator ignores maintenance advice process;

3) An unknown event where the operator requests event monitoring process.

Once the data analysis has been completed, the operator must be advised of the findings and outcome. This may be a simple process if there is a known event or more complex when the anomaly event cause is unknown. Here the operator may decide either to ignore maintenance advice, or to take notice of the advice, but not perform any maintenance action until the failure progression and potentially its cause become more obvious.

The Soft Systems Context Chart for a known event response process is shown by Figure 6.7.15 with all the Soft System boundaries that are likely to form that process highlighted by the emboldened interaction lines. These are primarily management functions and cross company boundaries. There should not be any negotiation activity here as the event is a known one and the response by the operator should be defined.

This however is an occasion where the Service provider cannot control events within the operator process and a notional set of tasks is defined for this process in Figure 6.7.16. The Soft System Reliability Risk View shows the interactions between the operator and the Service Provider as 'manual' represented by red interaction boxes. This is because the response from the operator may be that operational circumstances do not allow for repair in the timescale defined. This is a worst case approach and should be adopted initially. The idealised solution would be one where all of the interactions are automatic and represented as green, but until evidence of operator behaviour to support this view can be built.
the interactions should be set to ’manual’ at least for initial assessment purposes as this raises awareness of the issue to management when considering the input of the assessment with regard to maintenance response time to the business case.

The Soft Systems Context Chart for an unknown event where the operator ignores the maintenance advice provided by the Service provider is shown in Figure 6.7.17. Again this is a management function between the service provider and the operator, but in this instance it is also in the interest of the Service provider to monitor the progression of failure to ensure their position is covered if contractual issues are breached. Figure 6.7.18 shows the Soft Systems Reliability View of this process, again all of the interactions are shown as ‘manual’ as the data transfer between the various function is unknown. There is a cue here however for the process to define a set amount of time to continue monitoring in this contingency. This set time could be dependent on the maximum failure progression rate that is known or could be an arbitrary figure based on the view of a specialist. Determining that such a figure is required however helps in understanding the boundaries of the defined business case.

The final process, that of an unknown event where the operator requests event monitoring process may come about as a request from the operator either because they are unsure of the quality of the data that has discovered the anomaly or due to operational considerations. The Soft Systems Context Chart for this eventuality is shown by Figure 6.7.19 and is ostensibly the same as that for the previous process, the only deviation being in the way the various Soft System boundary functions relate to each other. This relationship is shown in Figure 6.7.20 which again shows the Soft System Reliability Risk View. This view is essentially the same as that shown in Figure 6.7.18 except the data is monitored by a specialist on a defined basis to ensure that the rate of failure progression does not rapidly increase. If it does or an analysis is forthcoming then that is passed to the operator via Service operator management. All of the interactions here again are described as ’manual’ apart from the ’automatic’ data update to the specialist.
In this particular case, where there are two specifically different relationships of the Soft System boundary function (and hence two different Soft System Reliability Risk Views) stemming from what is effectively the same Soft System Context Chart, there are two approaches that can be taken. Either the Soft System Context Chart is left as it is and the situation is accepted as such, as the Soft System Context Chart has done its job and provided guidance for process definition or a new Soft System boundary function could be added to the Soft System Context Chart to account for this difference in approach and thus differentiate these functions at Soft Systems Context Chart and Soft System Reliability Risk View process levels. This approach is valid if a number of similar response types are expected which align with this new function which will then enable a more integrated process to be developed. The former approach is adopted here and will not affect the resulting analysis as analysis is only ever performed at the Soft System Reliability Risk View level of abstraction.
Figure 6.7.15 Soft System Context Chart for a known event response phase
Figure 6.7.16 Soft System Reliability Risk View for a known event response process
Figure 6.7.17 Soft System Context Chart for an unknown event where the operator ignores maintenance advice
Figure 6.7.18 Soft System Reliability Risk View for an unknown event operator ignores service provider advice
Figure 6.7.19 Soft System Context Chart for an unknown event, operator requests event monitoring
Figure 6.7.20 Soft System Reliability Risk View for an unknown event operator requests monitor progress of event
The detailed models determined above using modified Role Activity Diagram notation provide a qualitative assessment of the overall response based on the Soft System model processes and elements identified. The Soft System Context views show who is involved with which particular function and provides a configuration basis that all functions identified are also modelled (as determined by the emboldened interaction lines shown for each process).

When reviewing the risk to reliability of each aspect of the Soft System model, it can be seen that the data acquisition phase can be viewed as a primarily automatic function involving the operator and service provider and its partners, the data analysis phase as being an automatic/semi automatic function which involves primarily the service provider analysing the effect of reliability issues on his equipment and the resolution phase as being a primarily ‘manual’ function involving the operator and service provider making business decisions based on the state of the condition of that equipment.

Each of these phases will have a contribution to make toward the speed of resolution of equipment issues, and whilst it is recognised that the quantification of these effects is not easy (especially when dealing with business decisions) ignoring that effect entirely is detrimental to the business case for the reliability monitoring system as a whole.

### 6.8 Soft System modelling technique summary

The aim of Soft System modelling is to develop a view of the Soft System support mechanism that is put in place for the service provider to respond to a failure alert originating from the Hard System. This alert may be either diagnostic or predictive, in the latter case there is an additional failure progression time from the point of alert of incipient failure. The failure progression time from that point may vary according to operational conditions and so will correspondingly vary the time of execution of the Soft System support mechanism response, but will have no effect on the process described by the Soft System model.
Without an understanding of the capability of the Soft System, it is not possible to understand whether or not the Hard System is either appropriate or fit for purpose. Without a full understanding, consequent integration and optimisation of both systems, it is unlikely that the business case requirements will be fulfilled.

When developing a Soft System model, source documentation can take the form of flow charts as shown in figure 6.7.1 or Gantt charts which show how the response to a failure alert is made. Typically such Soft System representations are not integrated especially when they cross company boundaries and an effort should be made (as shown in the example application) to develop a process model which is integrated and complete. Developing an overview of each of the processes that are known is a useful method if developing an integrated view.

The timing of the process is also an issue and constructing a process which breaks at appropriate points in time helps not only to simplify the analysis but also helps in modularising the process such that different paths in the process can be defined to suit specific alert and response conditions. Once an integrated view of the process has been developed the analysis can begin with an assessment of the risk to the reliability of the communication links. Reliability risk in this sense is not to the loss of the communication links themselves but to the risk in being able to deliver the service. In this sense, ‘manual’ links offer more risk than ‘automatic’ links and this differentiation is shown on the process diagrams.

The developed Soft System model forms a basis for negotiation and subsequent redesign if it fails to meet the needs of any of those who responsible for the Soft System boundaries, it also provides a qualitative view of how the process will enact by the number of interactions and their nature and forms the basis for more detailed numerical analysis techniques.

Unlike the Hard System model, no ‘summary’ view of reliability risk is required. This is because, unlike the Hard System which may have a number of reliability monitoring designs to act as mitigation against reliability risk, the Soft System
mitigation will take the form of process redesign, interaction improvement or process improvement.
Chapter 7 - Proposed reliability assessment method

7.1 Introduction

Any reliability assessment of large complex systems (whether they are Hard or Soft Systems) is prone to error due to many reasons, some of which include:

- The scale of the system being assessed;
- Unexpected events which occur during system operation;
- Emergent system features caused by system integration;
- Environmental factors that are poorly understood at the design phase;
- Lack of or poor configuration control during the design process;
- Lack of failure data quality and quantity.

Many techniques, some of which are listed in Section 1.8.2, have been developed to contend with the factors itemised above.

It is important then to recognise that any assessment technique proposed for this research should be complementary to such techniques and as such will not require similar assessments to be made on the system models produced thus increasing assessment activity where it is not necessary to do so.

A reliability assessment of Servitised systems is not one that can be carried out for purely physical products. The involvement and role of human factors in the Servitised system suggests more a parallel with risk management techniques employed in risk management of projects.

The approach favoured by Chapman and Ward [51] is to include both quantitative and qualitative approaches to analysis of projects with the emphasis on a qualitative approach in early stage development and the emphasis on quantitative approaches in the latter stages of development. They take this view because ‘the effectiveness and efficiency of quantitative analysis is driven to an
important extent by the quality of the qualitative analysis and the joint interpretation of both.'

The aim of this research is primarily to provide a holistic and manageable view of complex Servitised systems with a reliability perspective. To do this a qualitative perspective is of prime importance and certainly in early stage analysis could be the only method used. This of course depends upon the amount of, and level of confidence in, data available for quantitative analysis.

7.2 Assessment aims for Hard Systems

The general assessment of Hard Systems will vary from those of Soft Systems as there is an inherent design aim to develop a safe and reliable Hard System in the first instance. Due to this remit, the system design will be assessed in its own right and have some consequent form of mitigation for safety purposes. If however a Hard System is to be developed for a Servitisation business case and that case requires reliability monitoring in service to support it, reliability issues which drive system availability become important and the Hard System must be reassessed in those terms.

Of particular importance in this case are the following:

- Which components are vulnerable;
- What is the level of reliability monitoring system coverage to provide mitigation against that vulnerability;
- Are design changes required in order for the business case to be upheld?

The primary aim of the Hard System assessment is to establish Hard System reliability weakness and the extent of the mitigation against that weakness in terms of system features. This in turn promotes the availability of that system. In order to do this the Hard System assessment must inform the design decisions relating to these concerns.
7.3 Assessment aims for Soft Systems

The assessment aims of the Soft System are perhaps more extensive than those of Hard Systems as there is a relative lack of inherent reliability in such systems as the design of these systems are not required to be certified to support system reliability. A certified Hard System by contrast would have to meet a rigorous testing regime usually defined by a regulatory body to ensure its fitness to operate safely and reliably.

The assessment aims are however also limited in their scope as there are a number of influences on the Soft System that cannot be measured directly as they are aspects of human behaviour which are not easy to quantify except in their effect on the system as it operates.

The Soft System assessment aim is to establish boundaries around the delivery of the maintenance response to an alert and then perform an analysis of the timeframe required from detection of an event and isolation of a failure to a proposed maintenance action. It should also establish the confidence in repeatability of that timeframe. It should be noted that other factors relating to process performance could be measured, such as training and skill levels of operatives. It is important however in the first instance to establish a baseline from which decisions on the potential of the Soft System to meet contractual response times and performance improvement can be made. The contention is that time assessment is an absolute measure which can be influenced by many factors but these factors will vary according to situation and so should be reduced by a more focused effort of more detailed analysis at a later time.

The primary aim is to determine the potential to deliver a service in a timeframe that fits both with customer operations and with failure patterns of vulnerable components.

There are other secondary objectives of such an assessment, these are:

- What is the exposure of the system as a whole to an undiagnosed or a previously unseen event?
7.4 The effect of the holistic nature of reliability monitoring system and also its ‘duality’ on assessment aims

The reliability monitoring system is considered to exist in both Hard System terms and Soft System terms as a whole. Changes in the Hard System will affect the Soft System and vice versa. This is particularly pertinent to the Soft System Secondary aims. These aims are not wholly addressed by this research but the simple modelling and assessment method proposed will identify these aspects of the system design as a whole. It is important to realise that both of these secondary objectives are as much influenced by the Hard System design as the Soft System design.

That is, if sufficient infrastructure is not in place to transfer data for analysis from the Hard System to the Soft System or to transfer the analysed data around the Soft System, then the system as a whole is at risk. In the same way, if not enough is understood about the exposure of the Hard System to failure (failure event mechanisms are not understood or there are no monitoring systems able to monitor to sufficient detail to detect the failure occurring) then the system as a whole is at risk. How much risk depends on the mitigation coverage of the reliability monitoring system and how vital the unmonitored components are to the availability of the Hard System.

Both of the secondary Soft System aims will be very specific to each particular assessment case, which is why no explicit assessment is proposed for them, but the methods described here could be used to determine an appropriate response in these cases.

In addition to these holistic effects the reliability monitoring system has a ‘dual quality’ at the interface between the Hard and Soft System. This interface is that of the data transfer from the Hard System being monitored to the data processing function of the Soft System.
This dual quality is shown in the Soft System modelled example. The data gathering and transfer apparatus of the engine and airframe has a Soft System component regarding transfer time. It also exists in the Hard System sense as physical components with associated reliability failure rates. There is no conflict of interest here. In this case the components are considered only to have the attributes pertinent to the system for which the assessment is being made.

7.5 Hard System assessment

There are two ways in which Hard Systems assessments have been made in this research. Which method should be used depends upon the information required from the assessment as much as upon the nature of the system.

These methods are discussed further in Sections 7.5.1 and 7.5.2.

7.5.1 System assessment for Hard Systems of a simple nature or those without a safety assessment made against them

This type of assessment is typically used with either very simple systems or when a summary appreciation is required of the monitoring system coverage rather than capability. Its information source is primarily that of the system design documents and not any system safety assessment information such as an FMEA.

The modelling notation used to produce the Hard System models described earlier, is developed ultimately to provide a basis for identifying areas of reliability vulnerability in Hard Systems. These models form the basis of a qualitative assessment of a reliability monitoring system.

This model of the system design under assessment is annotated to show:

- Which components are monitored;
- Which components perform a monitoring function;
- Which components are either not monitored or are only partially monitored.
The aims of this type of assessment are very simplistic:

- What is the extent of monitoring system coverage of the hard system? (a measure of mitigation against loss of availability);

- Are there any Hard System components that are not covered by the monitoring system but may raise reliability issues (because of difficulty of component replacement, availability of replacements, etc).

The assessment method itself only provides a basis upon which these issues can be debated further by bringing them into relief.

Providing (for instance) a percentage figure of component coverage by the monitoring system may be of use to give some kind of confidence measure, but this would have to be balanced against the cost of achieving a higher level of monitoring system coverage and that some of the components considered may not themselves be vulnerable to reliability issues. These aspects of the system are important, but unless they are being used to compare one design solution against another at an early stage in system design are not seen to be practically relevant.

The benefit of performing an assessment such as this is to use it to establish the risks inherent in delivering system availability, but that risk is highly dependent upon particular assumptions that will have to be made when a formal safety assessment of the design solution is not available. The assessment therefore should be used as a guide to aid design decisions at an early stage.

An example assessment based on Figure 3.4.3 which shows the Hard System Reliability Risk View developed for the example tank system would provide the following information to the system engineer by inspection:

1) The destination flow from Valves V2 & V3 will need to be established to determine whether or not ‘down stream’ effects could impact negatively on the tank system reliability or availability. (no destination flow is defined in the information supplied);
2) The spill tray detector switch (SP1) and the Flow meters (VF1, VF2 and VF3) control and monitoring mechanisms should be defined and reappraised on that definition as the design intention is not confirmed with the information provided (all of these components are identified in the model as 'red' boxes);

3) Valve V2 is manually operated, the assumption is that the failure of the valve is not automatically detected but this will have little effect on availability as the valve will either have a specific maintenance routine applied to it or will not cause a serious risk to availability and is highlighted as such. This may be a cheap and appropriate design solution, the model merely highlights it as such;

4) From the diagram two components are monitored, (V1 & V3) some need further definition (SP1, VF1, VF2 and VF3) and others are assumed acceptable (V2) in their monitoring strategy.

Some quantitative assessment can be made but this information is of secondary and minor importance to the main qualitative assessment described above.

An example would be for monitoring system coverage for this example. There are 2 components monitored from a potential 7 components that could be monitored, therefore the monitoring system coverage is:

\[
\frac{2}{7} \times 100 = 28.5\% \\
\text{(7.5.1)}
\]
7.5.2 System assessment for Hard Systems of a complex nature or those with a safety assessment made against them

The second method is more applicable for highly integrated complex systems, typically those systems that have had a safety assessment or FMEA made against them or where the capability of the monitoring system design is under review. The system is partitioned to provide a view of which modules and components provide a reliability risk for each interface type based on information from the FMEA. This partitioning is necessary as it provides a system wide context to the information contained in the FMEA and allows the analyst an opportunity to use this context to:

- Determine which vulnerable components are unlikely to be covered by a reliability monitoring system;
- Assess opportunities for reliability monitoring system integration both within and across interface boundaries;
- Develop a view of the capability of the proposed reliability monitoring system;
- Specify which components will be used as monitoring sensors and relate them to the components in the interface that they will monitor;
- Gain a perspective regarding what proportion of the monitoring system will be prognostic and what proportion will be diagnostic.

This system wide contextual view is important as sensors which may not have been used in the past for particular fault detection activities may be developed to be included in any reliability monitoring system mitigation against the identified reliability risk as an extended use of that sensor or may be used either to supplement other sensors or to play a part in identifying stages of failure mechanisms ‘witnesses’ to threshold levels of failure. It also confirms the status of specific sensors in the reliability mitigation which may be called on in a number of ways in the mitigation such that the reliability of such sensors can be recognised.
This method was demonstrated in chapter 5. In the example given, the FMEA was used to develop a Hard Systems Reliability Risk View of the oil system interface (Figure 5.7.4) to illustrate this. All of the components that relate to the oil system interface are included in this view.

In addition to establishing the components that relate to each system interface, Figure 5.7.4 also establishes which module the components belong to. This is important information if one of the modules is supplied by a third party. In this case, monitoring of this module may be carried out by the third party or there may be no monitoring of the module which may require the system integrator to develop a monitoring capability.

Figure 5.7.4 also categorises each component with a particular colour which relates to a vulnerability category defined by the FMEA. This means that Figure 5.7.4 provides qualitative and quantitative information on the system. The qualitative information in the form of the component and its context (a particular failure mode has been identified for that component) and the quantitative information in the form of a category of vulnerability supplied by the FMEA.

This is not the end of the assessment however, as discussed in chapter 5, the Hard System Reliability Risk View is used to develop monitoring strategies which provide mitigation against the reliability risks posed by these components. When these strategies are developed, be they predictive or diagnostic, they are summarised and played onto the Hard System Reliability Risk View as shown in Figure 5.9.3.

Figure 5.9.3 provides a qualitative assessment of the vulnerability of the particular components in the oil system interface and provides a summary indication of the monitoring system coverage by replacing the colour in the component boxes to show the component is either being monitored (and how it is monitored) or is part of the monitoring system. All those that do not have some form of monitoring laid against them will either have to be reviewed to ensure whether redesign is required, have monitoring capability developed for them or will require a maintenance programme to be designed to ensure they meet
availability targets. They do not require further quantitative assessment as that is already apparent from the colour of the component boxes that are not monitored. If redesign is necessary, any resulting changes made to the design and subsequently the FMEA will need to also be reflected in these Hard System Reliability Risk Views.

Those components which are identified by the Hard System Reliability Risk View to be at risk and have no monitoring capability directed at them would at first appear to be a minor benefit. This because the FMEA will be expected to have identified these components as such, but the modelling technique and subsequent assessment highlights and isolates these cases graphically. This is not a primary function of the FMEA and it is certainly not a function of the FMEA to identify these specific components in relation to the coverage and capability of the reliability monitoring system.

As with the simple system various factors on component monitoring coverage, level of prognostic to diagnostic monitoring can be developed. Such statistics are useful especially when providing comparison between candidate systems but these are secondary considerations when compared to the prime purpose of the assessment discussed in Section 7.5.2.

7.6 Hard System reliability assessment summary

As far as Hard System reliability assessment for complex systems is concerned, it must always be remembered that any formal quantitative assessment will always be made using the FMEA mainly because this is a certification document but also because there is no real need to generate extra work in performing a separate analysis. There is no need therefore for specific calculations to be made in the assessment discussed as part of this research for Hard Systems as it is meant to complement the FMEA.

This complementary reliability assessment for complex Hard Systems is intended to act in the same way as the FMEA however, as a driver for Hard System redesign or reassessment in light of its findings. It focuses primarily on
documenting the coverage and capability that can be provided by the reliability monitoring system on Hard System vulnerable components and identifies through the Summary Hard System Reliability Risk Mitigation View:

- Which vulnerable components are not covered by the reliability monitoring system and as a consequence whether a system redesign is required;
- What design issues are outstanding to develop the reliability monitoring system in terms of required new sensors, improvement in sensor design or rationalisation of sensors;
- Develop a view of the capability of the proposed reliability monitoring system and highlight where improvements are required;
- Gain an understanding of the level of predictive monitoring that is required in order to provide requirements to the Soft System for data acquisition and transfer.

All of these findings are subjective and will depend on the Hard System under assessment, but can be derived from the assessment technique described above.

### 7.7 Soft System assessment

When developing a Soft System assessment of the maintenance response to a detected fault or failure condition, it must be recognised that the analysis will not be as comprehensive as one that is made for a Hard System. This is due to the multifaceted nature of the Soft System both in organisational terms and purpose of the system which is varied and varying against the defined, specific purpose of the Hard System.

That said, there is still benefit in developing an assessment technique that can be used as a metric for fitness for purpose of the proposed Soft System and as a metric for future improvement and development of the Soft System. The prime purpose of the assessment should be to give some view of the fastest possible maintenance response time when prompted by data indicating a potential
reliability event. This provides guidance for the business case if, for instance, the maximum loss of availability period required is less than the best case response time, the contract will have to be renegotiated.

Such a metric also provides a basis for developing maintenance strategies to best cope with particular failure mechanisms or operational and repair limitations. For example, if the maintenance response mechanism cannot identify a reliability event from data provided in a timescale that will allow for a repair action to take place to avoid a service failure, then the options would be to either develop a more effective prognostic monitoring technique, develop a faster analysis or trending technique or failing that develop a regular maintenance programme for that particular failure mechanism.

In itself however, as suggested earlier, such a maintenance response time can only be used as a guideline as it is subject to operational and other factors, therefore some confirmation of the consistency of this ‘best case’ figure will be required. There are a number of methods that can be used to develop the statistical bounds of this figure, such as Monte Carlo analysis [52].

### 7.7.1 Developing Soft System models for reliability assessment

Developing Soft System models which represent the maintenance support mechanism to this level is in itself a not inconsiderable task. To interpret that model to support the business case objectives (and to convince senior managers that any organisational changes/compromises required to support that model are necessary) some form of an analysis is required both regarding the appropriateness of the modelled systems function and the efficiency of this function.

One such efficiency measure is the time it takes to enact the Soft System that is modelled. It is recognised that this time will (due to the multifaceted nature of Soft Systems) also include ‘intangible’ effects such as human factors and other constraints. These effects are not easy to filter out from the overall time taken to enact the Soft System and it may be that variation in timings due to these effects
is minimal. It is not seen as a difficult task though to develop statistical bounds around these times as a means of judging just how significant these effects are in order to develop a 'best case' response time.

It has been stated before that Gantt charts and the modified Role Activity diagrams proposed by this research can exhibit commonality in that Gantt chart bars can be related to tasks and deliverables can be related to interactions on the modified Role Activity Diagram. The important aspect of that commonality is the information provided by Gantt charts in terms of resource commitment, or how many hours it takes to perform specific tasks that is detailed by the Gantt chart bars. Note that the number of hours to complete tasks are usually fixed and can only be completed more quickly if more than one person can be assigned as resource to complete the task. Such an arrangement would if say, two people were assigned to a 100 person hour task, mean the task would be completed in 50 person hours. Such an arrangement presupposes that two people can work on the task simultaneously and that they would work with the same level of efficiency.

This means that information on task timings is (given the Soft System model is properly planned and the plans follow the model) freely available, the assumption made in developing the 'best case' service delivery time is that the task timings are set and are not affected by resource loading issues (that is introducing two people to perform the task in half the time). Such issues can be regarded as ‘fine tuning’ of the Soft System model by management.

In terms of interface between system boundaries, which in effect are data exchange timings, these are not usually represented by either Gantt charts or other types of Soft System representation and are therefore more problematic to quantify in the first instance. By the same token these interfaces can be diverse in nature and can rely as much on human centred forms of communication activity (fax, e-mail, etc) as through automatic data exchange over a computer network. The interfaces have to be treated in a similar way in assessment terms however despite this. The approach with interfaces then is the same as with
tasks and is time based measured from the time the data leaves a task to the time received at the subsequent task. As with tasks, this timing will vary and although automatic linkages can be assumed to be instantaneous in the first instance, these should be monitored and if outages in the linkage occur regularly during a set period of time, these should be accounted for in the ‘best case’ service delivery time.

It is assumed that the ‘best case’ service delivery time will be initiated from such an automatic link and care should be taken that the time taken from the point of detection to the beginning of the first task in the Soft System is accounted for. In many cases, the fault detection and communication to the Soft System will be instantaneous, but it is also possible, especially for predictive mitigation cases (where there is no definite indication of a fault until the input data is processed) that there is a considerable delay between the data required to start the process being available to the Soft System. This could be due to the requirement to transfer large amounts of data or it could be that the chosen data transfer medium is not performed over wires but via a Compact Disk through the postal service.

Human based interfaces are expected to vary with time, competence and the decision level that is associated with what to transmit. Such an approach by definition requires some form of probability assessment, which it is acknowledged, will be difficult to provide accurately in the early stages.

7.7.1.1 Relating timings and process flow conditions to the Soft System model

As can be seen in Figure 7.7.1, both tasks and communication paths are labelled. Soft System model tasks are labelled T<identifier number> and Communication paths are labelled C<identifier number> each number is specific to that particular task.
Figure 7.7.1 Soft System Reliability Risk View of the data collection phase (detailed process)
Each task or communication path has a particular time to complete associated with it. To calculate the time taken to complete each section of the system modelled the following rules are observed.

If the model is defined by purely sequential processes where the process flows without interruption through system boundaries and between system boundaries (such as those described in Figure 7.7.1) each time associated with each task or communication path is simply added together from the start of the process to process completion.

It should also be recognised that there may be special conditions relating to each communication path. For instance, electronic and software links are assumed to be instantaneous. It may be though that such links are subject to ‘data polling’ whereby a data source is accessed regularly or ‘polled’ every hour. If this is the case then this delay should be included at its maximum time period. Thus the worst case assumption is made in terms of time delay, an example illustration of this worst case approach is that an alert occurs just after the data from that point has been polled.

There are two other cases which do not exhibit this sequential process flow. One such is where the Soft System model is seen as a ‘master process’ where a system boundary drives the process to other system boundaries and the other as a ‘slave process’ whereby the process is halted until data or commands are required to be received from a System boundary before the process flow can recommence.

An example of a ‘master’ process is that given in Figure 7.7.2. Here, in the case of the Soft System boundary function ‘health monitoring function – analysis co-ordination’ there are a series of communication paths that originate from this Soft System boundary function and in this case the process is allowed to continue once the communication or control is made from these points.
Figure 7.7.2 Soft System Reliability Risk View of the unknown anomaly prognostic process
An example of the ‘slave’ process is that given in Figure 7.7.3. In this case, the same basic principles for process flow apply for task and communication path timings but here when a communication path is arrived at data or control from the ‘sending’ system boundary must be available for the process to continue.

Taking as an example the Soft System boundary functions in this figure ‘service provider operations management – maintenance notice’ and ‘operator data centre, operations’, the task T324 (determine business effect of analysis/operator response and operator response times) cannot be started before the communication C322 (inform service provider of decision) has completed.

The implication is that specific ‘blocks’ of the process have to be resolved before continuing with the sequential addition of following tasks. This rule is of greater importance when establishing process ‘reliability’ as discussed below as it depends upon not just the type of issue being decided by the operations function in the operator data centre but also the efficiency of the staff performing that function as to how reliable or ‘repeatable’ such a response time is.

Determining the statistical bounds of operator response may have a significant effect not only on the design of the process but also the contractual limits to which the operator has a right to influence equipment operation (for example, if the operator wishes to operate the equipment against the advice of the service provider this could be accommodated but at a higher contractual risk premium).

Both aspects are important factors in determining the business case of the overall service.
Figure 7.7.3 Soft System Reliability Risk View of the unknown event operator requests monitor progress of event
7.8 An example Soft System timing analysis

An example process would be that of a 'known' alert which has been identified and is diagnosed by a known failure pattern and for which a particular response has been developed (due to the familiarity of the failure mechanism). The following models and analysis define the 'best case' time that can be expected for this process to execute. Statistical boundaries are not placed about this time, but such an analysis extension could be performed using Monte Carlo methods.

Such a scenario can be developed using the example models produced so far. These will be: from stage 1 (data acquisition phase) the data capture process shown in Figure 7.7.1; from stage 2 (the data analysis phase) the known anomaly diagnostic process shown in Figure 7.8.2 and finally stage 3 (the resolution phase) will use the known event response process shown in Figure 7.8.4.

We know from the Soft System Reliability Risk Views defined by Figures 7.6.1, and 7.6.5 that the data acquisition and data analysis phase communication paths are primarily automated and therefore, these phases should return a fast execution time. The event response defined by Soft System Reliability Risk View shown in Figure 7.8.4 however has primarily manual communication paths because of the need to deal with the operator and the predominantly 'unpredictable' operator response Soft System model.

The total execution time for the process is developed by taking each Soft System model figure in turn and developing an execution time for each phase and then adding these times together. This modular approach allows a number of phase combinations to be brought together in any combination required to develop an overall time relating to a variety of Soft System model combinations.

7.8.1 Data capture process shown in Figure 7.7.1

Using the defined example the process flow for data acquisition can be directly interpreted from Figure 7.7.1 but in order to illustrate further the process is shown in block diagram form in Figure 7.8.1. Note that timings are added to the block
diagram, these would naturally form part of the process defined by Figure 7.7.1 but are not shown as such in Figure 7.7.1.

Also note that the data acquisition phase process could apply to both diagnostic and predictive events. Failure progression time for a predictive event is not considered specifically here as such a timing will relate to a specific component and monitoring strategy that will need to be defined by the Hard System and may also require a specific Soft System mitigation strategy that differs from the generic Soft System approach demonstrated by this example.
Figure 7.8.1 Block diagram of the data acquisition phase defined by Figure 7.7.1
This Soft System model is a sequential process type which flows without interruption from Soft System boundary function to Soft System boundary function. It is therefore a matter of adding each task time and communication path time (taking into account any particular timing attributes of that communication path).

In this process, we have the following tasks and task timings:

*From Soft System boundary function ‘Engine’*

Task T11 – receive sensor input – 20 milli seconds.

Task T12 – Buffer, analyse data – 5 seconds.

Comms path C11 – transmit to main airframe data store – 1 minute.

*Sub Total* 1 minute 5.02 seconds.

*From Soft System boundary function ‘Airframe’*

Task T13 – buffer, analyse data – 100 milli seconds.

Comms path C12 – provide alert data to analysts – 8 hours (note that such a delay could be attributable either to operational circumstances or to data transfer practicalities).

*Sub Total* 8 hours 0.1 seconds.

*From Soft System boundary function ‘Service provider data centre’* (note that the operator data centre is noted to have the alert data but plays no further part in the direction of the process and so is ignored).

Task T14 – process data check for alerts – 2 hours.

Comms path C13 – provide alert information to service provider operations management – 1 minute (an automatic link is assumed, but it may take this time to register with the operations centre).

*Sub Total* 2 hours 1 minute.

*Data acquisition process Sub Total time* = 10 hours 1 minute 5.12 seconds.
7.8.2 Known anomaly diagnostic process shown in Figure 7.8.2

This process is defined by Figure 7.8.2 and further illustrated by the block diagram shown in Figure 7.8.3, again specific timings relating to each task are shown only in Figure 7.8.3.
Figure 7.8.2 Soft System Reliability Risk View of the known anomaly Diagnostic Process
Figure 7.8.3 Block diagram of the Known anomaly Diagnostic Process defined by Figure 7.8.2
In this process there are the following timings:

*From Soft System boundary function ‘service provider operations management – known failure analysis’*

Comms path C21 – provide known anomaly data – 1 minute.

*Sub Total* 1 minute.

*From Soft System boundary function ‘Health Monitoring Function – Diagnostic analysis confirmation’*

Task T21 – review data to confirm anomaly type – 1 hour.

Comms path C22 – inform service provider of decision – 1 minute.

*Sub Total* 1 hour 1 minute.

*From Soft System boundary function ‘service provider operations management – known failure analysis’*

Task T22 – determine business effect of analysis – 2 hours.

Task T23 - determine business response – 30 minutes.

*Sub Total* 2 hours 30 minutes.

**Known anomaly diagnostic process Sub Total time** = 3 hours 32 minutes.

**7.8.3 Known event response process shown in Figure 7.8.4**

This process is defined by Figure 7.8.4 and further illustrated by the block diagram shown in Figure 7.8.5, again specific timings relating to each task are shown only in Figure 7.8.5.
Figure 7.8.4 Soft System Reliability Risk View of the known event response process
Figure 7.8.5 Block diagram of the known event response process defined by Figure 7.8.4
In this process there are the following timings:

*From Soft System boundary function ‘service provider operations management – maintenance notice’*

Comms path C31 – response to customer – 1 minute.

*Sub Total* 1 minute.

*From Soft System boundary function ‘operator data centre – operations’*

Task T31 – review service provider response – 2 hours.

Task T32 – review equipment business commitment – 1 hour.

Comms path C32 – alert maintenance function – 30 minutes.

*Sub Total* 3 hours 30 minutes.

*From Soft System boundary function ‘operator data centre – maintenance function’*

Task T33 – review resource and determine repair schedule – 30 minutes.

Comms path C33 – agree repair schedule – 30 minutes.

*Sub Total* 1 hour.

*From Soft System boundary function ‘operator data centre – operations’*

Comms path C34 – inform service provider of decision – 1 minute.

*Sub Total* 1 minute.

*From Soft System boundary function ‘service provider operations management – maintenance notice’*

Task T34 determine business effect of analysis/ operator response and operator response timing – 30 minutes.

*Sub Total* 30 minutes.

**Known event response process Sub Total time** = 5 hours 2 minutes.

### 7.8.4 Complete Soft System model response time

Adding each of the sub totals together gives a ‘best case’ service delivery time for closing out a known diagnostic anomaly at:
10 hours 1 minute 5.12 seconds + 3 hours 32 minutes + 5 hours 2 minutes =
18 hours 35 minutes 5.12 seconds.

This time should only be used as guidance as it by necessity shows that all
Soft System boundary functions are in place and are ready to accept their
task on demand.

That said, if the time developed here as guidance exceeds the time in which
the diagnosed failure will progress from the time to diagnose the alert to the
potential failure point (in other words the reliability event cannot be avoided
due to the length of time it takes to schedule a maintenance action), then this
is an indication that either the detection capability has to be made
autonomous and detection will trigger an equipment stoppage, or the Soft
System maintenance response mechanism has to be modified or made more
efficient, or the Hard System design has to be modified, perhaps including
some redundancy for this particular failure mode.

In the same way that these Soft System models have been developed and
analysed, other models such as those shown in Section 7.7 can also be
analysed.

Temporal partitioning allows for interchange of process routes and process
scenarios. The data capture phase is shared by all following phases, the Soft
System model following data capture will depend upon the type of data alert
provided by the data capture phase and its completion time will vary
according to the amount of further monitoring or processing that is required.

The event response phases depend upon the attitude and contractual
agreements reached with the operator, for example in the analysis given
above the operator may not provide detailed Soft System models of their own
procedures as shown in the known event response diagram Figure 7.8.4
(operator data centre roles). This may not be a problem if it is not considered
part of the reaction time to the event, but if it is and forms part of the
contractual agreement some form of repeatability measurement of the
response time will be required to ensure that untimely communication
between the operator and the service provider does not risk either the
equipment or the profit margin of either business.
7.9 Soft System execution time reliability analysis.

The measurement of the ‘best case’ service delivery time is useful only as an initial guideline. A view that will confirm the reliability of timings in the process is more useful for testing the delivery of the service and also as a baseline on which process improvements can be made.

The ‘reliability’ of Soft System model execution time can be determined in a number of ways, the preferred method would be to determine a statistical boundary about process execution times based on a number of timings of that process being executed and from there develop a best case and worst case timing to be fed into the business case and system design iteration.

If such information is not available or the process is not fully developed or needs integration then process plans from Gantt charts can be used not only to design the process but also to give an idea of process timings for tasks.

The process representation can be organised to suit the information available in terms of the level of detail needed to model the process and get meaningful timing information. In the final instance it is more appropriate to use processes that are already in place and try to find some method of integration of these processes than to design the system from ‘top down’ at least in the first instance.

Timing analysis of the process developed from ‘bottom up’ not only builds on a solid practical base, but also will guide development, improvement and further integration of that process.

7.10 Soft System reliability assessment summary

The aim is to analyse the Soft System model in order to determine the ‘fitness for purpose’ of the maintenance response mechanism that is put in place for the service provider to respond to a data alert from the Hard System.

Without an understanding of the mitigating capability of the Soft System, it is not possible to understand whether or not the Hard System components which define the reliability monitoring system are either appropriate or fit for purpose. Without a full understanding and consequent integration and
optimisation of both systems, it is unlikely that the business case requirements will be fulfilled.

The temporal nature of the Soft System is also an issue and constructing a Soft System which is partitioned at appropriate points in time helps not only to simplify the analysis but also helps in modularising the process such that different paths in the process can be defined to suit specific alert and response conditions. Once an integrated view of the process has been developed the analysis can be started with an assessment of the reliability risk of the communication links. Such vulnerability to reliability in this sense is not related to the loss of the links themselves (as such an assessment is essentially a Hard System assessment) but to vulnerability in being able to deliver the service. In this sense, ‘manual’ links offer more risk than ‘automatic’ links and this differentiation is shown on the Soft System model.

Although Hard Systems such as computer systems and communication links play their part in delivery of the Soft System it is felt inappropriate to include Hard System reliability factors in the Soft System assessment as the effect of their failure will be captured in the assessment of the Soft System. That is if an automatic communications link fails regularly, the Soft System that relies on that link will consequently take a longer time than it should do and is therefore accounted for in that sense. Such an approach has to be held due to the nature of Soft Systems as human centred communication linkages form just as much an interface of Soft Systems as do automatic communication links and have to be treated in the same way. Other Hard System attributes such as data availability and data quality are considered in the same way, the effect of the attributes is ultimately captured using this technique.

Further analysis is carried out on the timings of the process for each individual process case and a ‘best case’ service delivery time is developed for each Soft System model case. This is especially important when monitoring mechanical components failure progression as these will be unique to the component, the failure mode it experiences and its circumstances of operation. If these cases have timings that are not able to support failure detection timings then the Hard System and Soft System designs may need to be revised.
Determining the accuracy of these timings is also important and developing a statistical boundary about these timings not only gives a more accurate view of the overall timing for each process case but also forms the basis for future process improvement.
8.1 Summary of work

The aim of this thesis is to develop a methodology which can be used to provide an assessment of the Hard and Soft elements of the reliability monitoring systems used to underpin the business case of Servitised products.

This is because any assessment methodology for reliability monitoring systems will necessarily have to address system components that relate not only to Hard System product faults but also to Soft System failures in terms of delivering a timely maintenance response through the maintenance support organisation which responds to detected faults in order to maximise product availability.

A number of maintenance methodologies were reviewed. There were no current techniques found in the literature able to address both of these aspects. A brief summary of monitoring techniques was also researched as this provides guidance on the primary technologies used to support the condition based monitoring methodology and provide reliability monitoring systems currently. The monitoring technologies used are employed to detect specific component failures, they are considered largely to be ‘stand alone’ rather than integrated systems as proposed here.

In developing a methodology for this research, a comparison was made with another activity that involves both Hard and Soft System components and that is project management. Here risk assessment techniques are well developed and provide outline guidance to develop key requirements for the reliability monitoring system assessment methodology.

A brief review was also made of the techniques that relate specifically to Hard and Soft System modelling and assessment.

The key requirements believed necessary to enable development of an assessment methodology were defined and a modelling method was developed from them. The modelling method proposed is not a strict formal representation of either the Hard or the Soft system but rather a view of the
core and common aspects of each system type in order to provide a condensed but pertinent representation of each system. This is considered to be especially important when considering large and complex systems.

The common aspects of each type of system relate to the system building blocks which characterise the system, the ownership of those building blocks and the connections between the building blocks. All of these aspects can contribute to a failure to realise hoped for business benefits either singly or in combination. Each can also contribute in part to a less effective operational performance. These are important issues that do not solely relate to the engineering domain but can also have an impact on company organisation and commercial and legal agreements between the service provider and its partners and the customer.

An assessment technique that not only addresses product failure in engineering terms, but also allows organisational, commercial and legal aspects to be examined is crucial when dealing with Servitised products. Failure of the company infrastructure or organisational response hazards the business case for Servitisation in equal or greater measure to product failure. Definition and apportionment of contractual responsibility for such a failure is also crucial, especially if the organisational response crosses company boundaries as a shared responsibility.

This methodology was then applied to a simple case as an example. The modelling approach shows all design components of the Hard System but also differentiates those components that will perform the monitoring function. This may be obvious by inspection from the design but when the monitoring components are separated out in this way it provides the analyst the opportunity to distinguish whether those components could be relied on to perform some other diagnostic or prognostic capability for another part of the system, equally it will show those components that are relied upon to perform more than one system function, for example a monitoring and a control function.
In terms of Hard System assessment, it also allows discrimination of those components that are not monitored in order that an appropriate maintenance strategy can be determined for them.

In terms of Soft System assessment the linkages between different parts of the maintenance response are identified and highlighted if they are manual interfaces which are more open to interpretation, and are more liable to delay, corruption and disruption than those interfaces which are automatically made (as the interface format is defined and repeatable). Giving credit for automatic interfaces in this way highlights the need for well designed and tested automatic interfaces to be used wherever possible to increase the speed of response. This approach provides the means to perform such an assessment.

A critical appraisal of the application of this methodology to the simple case was then carried out. The results suggested that the initial methodology proposal should be extended in order to deal with larger scale integrated Hard and Soft Systems. The proposed methodology extension again uses a commonality of approach when modelling larger Hard and Soft Systems. Again this extension is not a formal representation of either type of system but is a method for demonstrating how each system and its subsystems relate both internally and externally to other systems and subsystems. This allows the modeller to examine specific system and sub-system interfaces without losing context of how those systems relate in the wider world. Such a method also provides focus to subsequent assessment as specific subsystem relationships can be highlighted as ones which need to be examined in more detail. An additional benefit of this approach is that external links can be ‘recognised’ as ones that although not yet defined cross either functional or organisational boundaries which, if required, can be more fully defined at a later date.

A further extension to Hard System modelling was also proposed which allows for Hard System modelling to be performed using the FMEA as a source rather than design documentation. This modelling approach was applied to large and highly integrated Hard System example and the proposed
methodology extension was applied to a more complex Soft System example where both formed a part of a reliability monitoring system.

Such an approach in Hard System terms, unlike the design modelling approach, offers the analyst the opportunity to take advantage of existing information regarding the vulnerability of the system to enhance the System models that are then produced for the reliability monitoring system assessment. The modelling technique differs here in that design is inherent in the organisation of the FMEA document and the modelling notation takes advantage of this fact. The same basic approach applies however in that component ownership issues are maintained as part of the notation for the model. The change in emphasis in using the modelling notation in this instance is with regard to the manner in which the model is partitioned according to system interface type. Partitioning the model in this way is an aid to developing corroborative monitoring strategies.

Developing such models forms the basis of a qualitative assessment but in addition inclusion of FMEA information on component vulnerability provides a quantitative aspect to the model.

Having such information available in a graphical format provides a clear view of reliability vulnerability in what can be quite complex designs.

Using these models, monitoring strategies were defined and then summarised graphically in order to determine what system reliability risks remained following the application of monitoring system mitigation. Developing these monitoring system models also highlighted which monitoring sensors were most important to the overall system design purely by how frequently they appeared in the monitoring mitigation views.

Soft system models were also generated from example process charts and developed using the proposed methodology. These models provide a qualitative assessment of the process and at the same time provide opportunities for process design rationalisation in terms of a reduction in departmental and cross company boundary interfaces and the standardisation of those interfaces.
The common Hard and Soft Systems interface was viewed as having different attributes for each system type. The Hard System attributes were related to the physical aspects of data transfer such as failure rates for the components involved, quantified by the FMEA. The main attributes of the Soft System were defined as the time taken for the data to cross the interface and how reliable and repeatable that time is.

In terms of the assessment aspect of the methodology, the aim is to provide an understanding of the efficacy of the Hard System aspect of the reliability monitoring system in terms of level of integration and suitability of sensors and level of monitoring coverage. Producing a Hard System model using FMEA information as a source highlighted these issues quantitatively as a matter of course and also revealed those components most likely to provide a risk to system availability. Such information provides guidance as to whether or not the Hard System aspect of the reliability monitoring system meets the business case needs.

In terms of reliability assessment of the Soft System, the important factors are those of maintenance response time and its repeatability both of which have a distinct impact upon the Servitisation business case. The Soft Systems model is produced such that it highlights interfaces and their type and allows examination of a variety of response times according to defined fault detection scenarios. A simplistic time based analysis approach for response time is defined. This assessment of response provides a broad coverage of aspects of both human and machine unreliability and in doing so, identifies parts of the System which may be improved rather than defining specific causes of unreliability.

Repeatability of the timing analysis is not demonstrated here but the use of Monte Carlo methods on the models produced is suggested as a way of developing such a metric initially, with more detailed time and motion studies to confirm the model and its metrics when in operation.

### 8.2 Conclusions

A methodology has been developed which assesses both Hard and Soft System components of a reliability monitoring system used to support a
Servitised business proposition. The methodology highlights system design issues holistically such that the business proposition can be examined in light of these inherent system weaknesses. The proposition thus informed may then require system redesign, extra capital investment or renegotiation of contractual obligations to deliver the proposed business benefits.

A modelling notation has been developed to provide a basis for the assessment and is used to deliver a common model representative of both Hard and Soft Systems. The models can be developed from a variety of source data using a notation which is scalable in order to enable investigation of larger and more complex systems and can be developed without need of specialist modellers.

The chosen modelling method is graphical and flexible. The flexibility of the approach means that the resulting model is a broad brush approach which could be left open to misinterpretation as the notation rule set is inclined to leave some areas ill defined in comparison to other notations. This disadvantage however is also seen as an advantage as the idea of the model is to engage participants in the design process and to subsequently stimulate debate on contentious issues. In addition a more open notation set is likely to provide a greater opportunity for non technical participants to engage with the modelling activity.

The graphical nature of the model is seen as a great advantage as it is likely to give sight to many issues that would be left hidden if they remained within a textual system description. There are however many other modelling notations that would provide the same benefit. The advantage of this modelling notation is the manner in which it is applied to both Hard and Soft Systems. By modelling only principal aspects of each system which relate to reliability monitoring systems a clarity of purpose and responsibility for that purpose can be derived from the model.

The modelling notation is not intended to provide a highly detailed and accurate model of the reliability monitoring system as to do so would inevitably duplicate the design process. It is intended to provide a concise
view of large system issues which raise the level of risk against the design aims of the system as a whole.

Reliability assessment techniques require that each model is enhanced to highlight perceived design weaknesses. Design weaknesses have different attributes for both Hard and Soft Systems. The Hard System design weaknesses in terms of reliability monitoring are straightforward and model enhancements are determined from existing FMEA information when available. Soft System design weaknesses are difficult to precisely encapsulate but it should be remembered that the main purpose of the Soft System is to support the monitoring strategies developed for Hard Systems, so any model enhancement for assessment is based around those aims specifically.

The Hard System monitoring strategies developed in response to perceived design weaknesses may be many and varied due to component type and function and may act corroboratively with respect to certain failure mechanisms but will mainly fall into either diagnostic or prognostic approach categories. Different Soft System approaches will be required for each category and may also be required for specific failure conditions. The graphical nature of the model enables discussion, formulation and documentation of a variety of modelling strategies that could be adopted for both types of system. This again demonstrates an advantage of the flexibility of the modelling approach.

Ensuring configuration of these approaches will rely heavily on expert opinion in the design stage and may not capture all possible maintenance responses to failures, especially those that are unanticipated and occur in service.

The overall aim of the assessment is to provide a comprehensive and holistic understanding of the maintenance response required to address a failure event. This is done not only through modelling and understanding the risk to availability of those components that cannot be monitored but also through an understanding of those components that can be monitored and the specific monitoring strategies that will be applied to them. A key aspect of these
monitoring strategies is the additional definition and assessment of the Soft System model that supports the associated monitoring strategy.

Quantitative Hard System assessment relies on models developed using FMEA information (integration of the methodology with these forms of analysis is desirable as it reduces duplication of effort). These models in effect show the information contained in the FMEA graphically. It is felt this approach makes the FMEA information more accessible and also shows that information in context in a more accessible manner than if it were distributed through a large text based document. If however there are a large number of failure modes which relate to a specific component, then it becomes difficult to represent all but the most important failure modes graphically.

Quantitative assessment of Soft Systems is based on reaction times to monitored failure events. It is accepted that these reaction times will also be affected by factors such as cultural and motivational issues which are not easy to filter out. Despite this, the overall reaction time (within specified time repeatability limits) is still seen as the main arbiter that supports achievement of contracted levels of product availability. Note that component failure progression time is considered inherent in the Soft System and is not considered as a distinct component of the system as a whole. This is because failure progression time is arbitrary (due to duty cycle) and specific (due to the component failure mode). The monitoring strategy and corresponding supporting Soft System should accommodate and reflect component failure progression as a whole.

It is recognised that the approach taken by this research is a simplistic one and would at best only differentiate large anomalies in the Soft System, but is an approach worth considering in the first instance, with the prospect of more sophisticated techniques being used to confirm identified areas or highlight new areas of Soft System design weaknesses in the future. A recommendation for one such method is made in section 8.3.

These assessments highlight potential areas of either system redesign or the need to modify how the business is intended to operate under its contractual agreement.
Providing an integrated reliability assessment across both the Hard and Soft System models is theoretically possible with this notation but has not been attempted here as the assumption is that Human centric Soft Systems are inherently (perhaps by some orders of magnitude) more unreliable than Hard Systems. Use of the common notation in modelling both Hard and Soft Systems has identified that there is a crucial linkage point at the interface between both system types.

It may be difficult to completely model this interface point using the common notation as the interface has different attributes which can be both Hard System and Soft System related. In the examples modelled in this thesis, no attempt has been made to accurately model the interface point, as the Hard System modelling concludes at the failure detection sensor point (see Figure 5.9.1 – Oil Analyser, Electro Mech. Chip Det. ) and the Soft System begins at a sensor detection of a failure point (see Figure 7.7.1 – receive sensor input T11).

The example systems used in this thesis did not extend to coverage of the interface between the sensor and the sensor data acquisition and dissemination device. There is a need to further investigate this issue. In most cases it is expected that the linkage will be delivered by a single physical entity such as a Hard System control element. In such cases, this single element is not detrimental to the methodology proposed as many of its functions, such as delivery times of data will be considered within the Hard System and may be available automatically to the Soft System with little extra processing required. In systems where this is not the case, the issues surrounding the interface point may be more involved and therefore worthy of further investigation.

This interface point should be reflected in either model as it is nominally the point at which failure data is passed between both models. The attributes of each system type at this point are shown to differ, but are seen as complementary to the system as a whole.

The proposed methodology as a whole provides a novel, simple and integrated approach to defining and ensuring that reliability monitoring
systems are able to meet their technological and commercial aims in providing enhanced product availability.

8.3 Further research

There is scope for further research to identify exact causes of unreliability in Soft Systems. As Soft Systems have many attributes such an assessment is likely to be performed over a long period of time and is therefore not ideally suited to concept stage assessments. It is suggested however that the approach proposed by this research is compatible with and supports the method of assessment proposed by Murdoch et al [53] for long term Soft System assessment.

The other topics for further research will need to be based on actual case study material as they will need particular rather than general information to inform a way forward.

For Hard Systems in particular, there is no defined method for indicating or recording how many failure modes can be detected overall for each component. This is a refinement on showing only that the component is monitored.

Further work also needs to take place regarding the organisational response time. In particular, at the Hard and Soft Systems interface. The capacity and reliability of this data transfer point is seen now to be a key feature of the assessment. In the examples given in this research, the Hard System is assumed ‘perfect’ at this point and in the Soft System only cursory modelling is performed. In some cases the data transfer time may potentially be greater than the failure detection time or when performing predictive monitoring that a ‘trigger’ event will need to be monitored for a potentially long time in the order of days or even weeks before an organisational response is required. Such issues should be highlighted in some way if they occur as part of the assessment as they are not at present. This is particularly important when trying to define and verify ‘physics of failure’ times in order to ensure that they are consistent and the monitoring system as a whole responds as and when it should do.
References


8. [http://www.gastops/MetalSCAN.html?s=13](http://www.gastops/MetalSCAN.html?s=13)


<table>
<thead>
<tr>
<th>No.</th>
<th>Author(s)</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>29</td>
<td>R Proctor, T Van Zandt</td>
<td><em>Human Factors in simple and complex Systems</em>, Allyn &amp; Bacon, Boston, 1994, page 51</td>
</tr>
<tr>
<td>38</td>
<td>SAE ARP 5580</td>
<td><em>(for aerospace applications)</em></td>
</tr>
</tbody>
</table>


54  K Knight, Matrix management, A cross functional approach to organisation, Gower Press, Farnborough 1977, pages 142-152.


60  http://www.rolls-royce.com/civil/services/totalcare/


