AD-A251 362

(2)

CDRL No. 0002AC-3

Functional Requirements and Measure of Performance for the Manufacturing Optimization (MO) System

Linda J. Lapointe
Robert V.E. Bryant

Raytheon Company

1992

DARPA
Defense Advanced Research
Projects Agency

CDRL No. 0002AC-3

Functional Requirements and Measure of Performance for the Manufacturing Optimization (MO) System

Prepared by
Linda J. Lapointe
Robert V.E. Bryant

Raytheon Company
Missile Systems Laboratories
Tewksbury, MA 01876

May 1992

ARPA Order No. 8363/02
Contract MDA972-92-C-0020

Prepared for

DARPA
Defense Advanced Research Projects Agency

Contracts Management Office
Arlington, VA 22203-1714

Contents

Figures

1. Scope

1.1 Identification

This is the Functional Requirements and Measure of Performance Report for the Manufacturing Optimization (MO) System. The development activities are being performed under Defense Advanced Research Projects Agency (DARPA) funding, contract number MDA972-92-C-0020, by the MO Development Team which is comprised of personnel from Computer Aided Engineering Operations (CAEO) of the Raytheon Missile Systems Laboratories (MSL) with participation from the Mechanical Engineering Laboratory (MEL) and the Missile Systems Division (MSD) West Andover Manufacturing facility.

1.2 System Overview

DICE has developed a concurrent engineering model that replicates the human tiger team concept. The basic tenet of the human tiger team is to have the various specialists contributing to the project co-located. In today's environment of complex product designs and geographically dispersed specialists, DICE envisioned a "virtual tiger team" working on a "unified product model" accessible by computer networks. Such an environment must enable specialists from each functional area to work on the design concurrently and share development ideas.

Raytheon has proposed a conceptual refinement to the original DICE virtual tiger team. This refinement is a two level approach with a product virtual team having a global view supported by information supplied by lower level "specialized" process virtual teams. See Figure 1-1. This refinement is needed because of the growing complexity of our products, and supporting development processes which make it difficult for one individual to adequately support a complex manufacturing position. The "virtual process team" concept would allow comprehensive representation from each specialized process area.

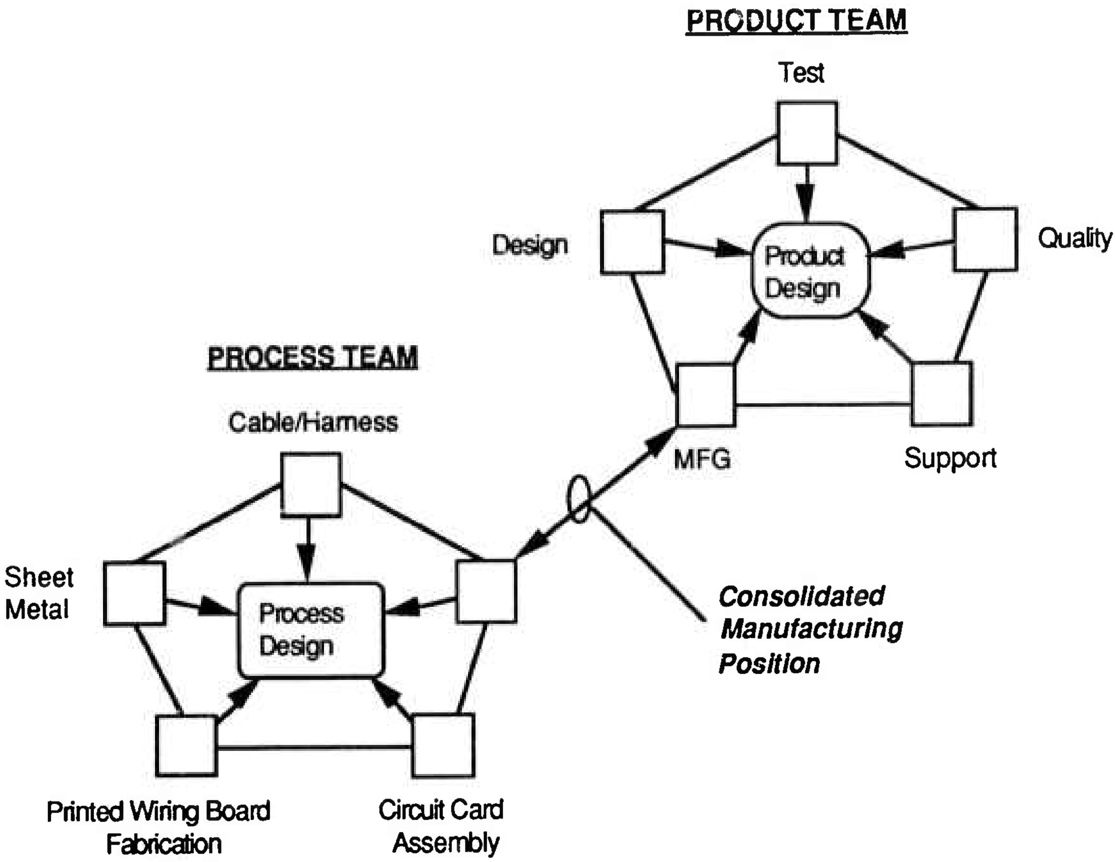

Figure 1-1. Two Level Team Concept

The purpose of the Manufacturing Optimization (MO) system is to enable all manufacturing specialists to participate in the product/process development activity concurrently. The system consists of a set of tools to model the manufacturing processes and centralize the various process tradeoffs. Recommendations can be compared and negotiated among the individual manufacturing participants. After the manufacturing team has reached a consolidated position, the results are passed back to the cross functional (top level) team for their negotiation.

1.3 Document Overview

The purpose of this report is to establish the software requirements for the Manufacturing Optimization (MO) System. It contains the functional requirements and measure of performance for MO.

2. Applicable Documents

1. BR-20558-1, 14 June 1991, DARPA Initiative In Concurrent Engineering (DICE) Manufacturing Optimization - Volume I - Technical.

2. CDRL No. 0002AC-1, 19 March 1992, Operational Concept Document For The Manufacturing Optimization (MO) System, Contract No. MDA972-92-C-0020.

3. CDRL No. 0002-AC-2, 19 March 1992, Description of CE Technology For The Manufacturing Optimization (MO) System, Contract No. MDA972-92-C-0020.

4. Object-Oriented Analysis, Second Edition by Peter Coad/Edward Yourdon, Yourdon Press Computing Series, 1991.

5. Reference Manual for the ROSE++ Data Manager, STEP Tools Inc.

6. Tutorial Manual for the ROSE++ Data Manager, STEP Tools Inc.

3. Functional Requirements

3.1 External Interface Requirements

3.1.1 Product - Process Team Communication

MO introduces the concept of a two tiered virtual tiger team. The two tiered approach consists of a cross functional product team linked to teams within each of the functions, in this case a manufacturing process team. To implement this approach there must be communication among the members of each team, and between the product and process team. The following capabilities are required to support this communication:

- Product - to - Process Team Communication
  - Notification of design task completed or other pertinent status information.
  - Notification and issuance of database available for analysis.
  - Notification of alternative designs or trade-off decisions under consideration.

- Process - to - Product Team Communication
  - Notification and issuance of analysis results.
  - Notification and issuance of modified database with recommended changes.
  - Notification of changes to the process, guidelines, cost or yield models.

3.1.2 CAD Database Interface

MO will use PWB design data, stored in a ROSE database, as input. MO will support an interface between the Raytheon Automated Placement and Interconnect Design System (RAPIDS) and the ROSE database. RAPIDS will provide a graphical CAD environment for displaying and manipulating the PWB product design. The interface will be bi-directional to support manipulation of the data within ROSE and subsequent re-use by RAPIDS. Since ROSE is a neutral database that uses the PDES/STEP standards, interfacing MO to other commercial PWB CAD systems is possible.

3.2 Capability Requirements

The requirements have been modeled using Object-Oriented Analysis (OOA) techniques where the required MO capabilities were organized into five major subject areas: process analyzer, guidelines analyzer, yield & rework analyzer, cost estimator, and manufacturing advisor. Each of these areas will be addressed in the following subsections where each part of the OOA model will be described in terms of its identifiable class-&-object(s) and related services (capabilities). Appendix I contains an explanation of the OOA notations used throughout the report. Appendix II contains the complete OOA diagram for the MO system.

3.2.1 Process Analyzer

The Process Analyzer provides the capability to select or determine the process sequence required to manufacture the product design. The manufacturing process will be represented by three levels of abstraction: the process, operation, and operational step. The process is an organized group of manufacturing operations, the operation is a common unit of work that is performed on the part, and the operational step is an elemental unit of work within an operation. The process operation sequence for a given product design will be selected from a list of all available operations within the process. The operation steps sequence within an operation will be selected from a list of all available steps within an operation.

The Analyzer will have the capability to select the process sequence based on the evaluation of product design parameters or process parameters. The evaluation function will use an "if-then" structure. An "if" statement that evaluates to true will result in the execution of the "then" statement. The "then" statement will add an operation or step to the process sequence list. Below is an example of a process operation sequence list with the appropriate selection rule.

<u>**Sample Fabrication Process Data:**</u>

Selection Rule: IF number of layers > 2 THEN

Operation No.	Operation Description	LGrade	Setup	Run	Efficiency
10	"mark part no. lot no."	10	0.00000	0.12345	3.12345
20	"pierce tooling holes"	7	0.00000	0.12345	2.12345
30	"oxide treatment"	7	0.12345	0.12345	1.12345
40	"bake panels"	10	0.12345	0.12345	2.12345
50	"lay up for lamination"	10	0.12345	0.23456	2.12345
60	"laminate"	10	0.00000	0.23456	2.12345
70	"lamination teardown"	10	0.00000	0.12345	2.12345
80	"route excess epoxy"	7	0.00000	0.01234	1.12345
90	"stress relieve"	10	0.00000	0.12345	2.12345
100	"oxide strip"	7	0.00000	0.12345	1.12345
110	"drill tooling holes"	7	0.00000	0.23456	1.12345
120	"pin panels"	7	0.00000	0.12345	1.12345

The process analyzer is represented in an OOA model pictured in Figure 3-1 which is comprised of five class-&-objects. Provided below are the services and corresponding descriptions which make up these class-&-objects.

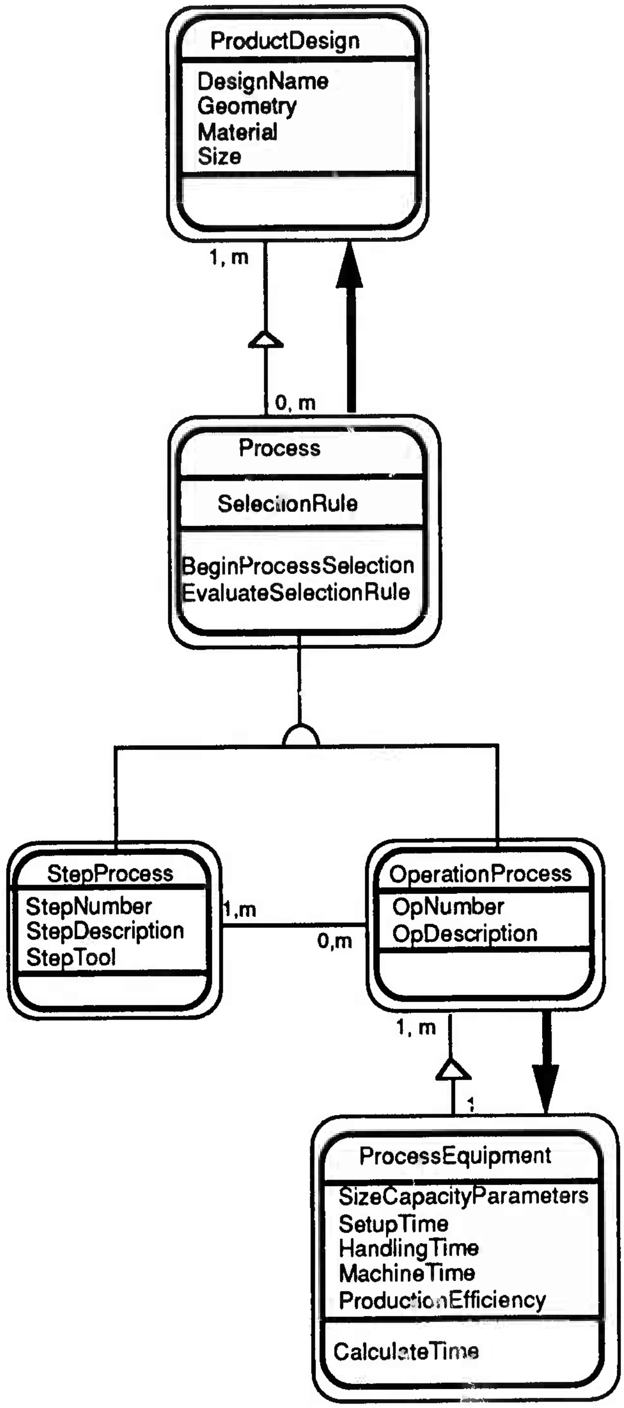

Figure 3-1. Process Analyzer OOA Model

specification **ProductDesign**
externalInput **CadDataExtraction**: Input from CAD Product Data Neutral Area
externalOutput **CadDataRequest**: Request for Data Extraction from Neutral Area

service **CreateProductDesign**
• creates and initializes a new ProductDesign object

service **GetProductDesignAttributes**
• retrieves values of the ProductDesign object attributes

service **SetProductDesignAttributes**
• set the attribute values of the ProductDesign object

service **DeleteProductDesign**
• disconnects and deletes the ProductDesign object

specification **Process**
service **CreateProcess**
• creates and initializes a new Process object

service **GetSelectionRule**
• retrieves values of the Process object attribute(s)

service **SetSelectionRule**
• set the attribute value(s) of the Process object

service **DeleteProcess**
• disconnects and deletes the Process object

service **BeginProcessSelection**
```
• repeat until all Process objects evaluated
    - evaluate process selection rules (via EvaluateSelectionRule)
    - if selection rule evaluates true then
            if Process object then
                    - insert Process object (via CreateProcess)
            else if OperationProcess then
                    - insert OperationProcess object (via CreateOperationProcess)
            else if StepProcess then
                    - insert StepProcess object (via CreateStepProcess)
            endif
    - endif
• end repeat
```

service **EvaluateSelectionRule**
```
• parse selection rule into postfix (reverse polish notation) token list
• repeat until end of token list
    - if operand then
            get token value and push value onto stack
    - if number or string value then
            push value onto stack
    - if unary operator then
```

pop off last item in stack
evaluate unary operation
push result onto stack
- if binary operator then
pop last 2 items of stack
evaluate binary operation
push result onto stack
• end repeat
• pop evaluation result and return value

specification **StepProcess**
service **CreateStepProcess**
• creates and initializes a new StepProcess object

service **GetStepAttributes**
• retrieves values of the StepProcess object attributes

service **SetStepAttributes**
• set values of the StepProcess object attributes

service **DeleteStepProcess**
• disconnects and deletes the StepProcess object

specification **OperationProcess**
service **CreateOperationProcess**
• creates and initializes a new OperationProcess object

service **GetOperationAttributes**
• retrieve values of the OperationProcess object attributes

service **SetOperationAttributes**
• set values of the OperationProcess object attributes

service **DeleteOperationProcess**
• disconnects and deletes the OperationProcess object

specification **ProcessEquipment**
service **CreateProcessEquipment**
• creates and initializes a new ProcessEquipment object

service **GetProcessEquipmentAttributes**
• retrieve values of the ProcessEquipment object attributes

service **SetProcessEquipmentAttributes**
• set values of the ProcessEquipment object attributes

service **DeleteProcessEquipment**
• disconnects and deletes the Process object

service **CalculateTime**
• repeat until end of selected process operation list
actual time = setup + (run time * feature quantity) * Efficiency
• end repeat

3.2.2 Guideline Analyzer

The guideline analyzer will provide the capability to evaluate a design against a set of design for manufacturing guidelines. Manufacturing guidelines may delineate quantitative and/or qualitative manufacturability issues. The guidelines will be structured with an "if" statement that defines the parameters the guideline evaluates, and a "then" statement which will be the recommended action or caution related to the evaluated guideline. A sample guideline for printed wiring board design is as follows:

Guideline: IF power/ground layers are not symmetrically positioned in layer stackup THEN

Recommendation: In order to meet the bow and twist specification of less than 0.015 in/in, it is important to have a balanced construction. This means a board stackup should have nearly symmetrical positioning of power and ground planes and interconnect layers with respect to the center-line of the board cross section.

Following are the services and corresponding descriptions which make up the MfgGuideline class-&-object.

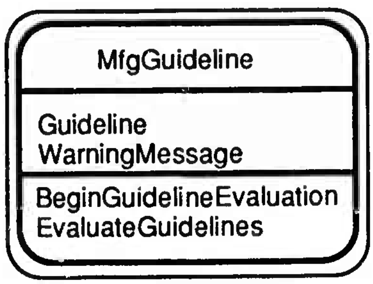

Figure 3-2. Guideline Analyzer OOA Model

specification **MfgGuideline**

service **CreateMfgGuideline**
• creates and initializes a new MfgGuideline object

service **GetGuideline**
• retrieves values of the MfgGuideline object attributes

service **SetGuideline**
• set values of the MfgGuideline object attributes

service **GetWarningMessage**
• retrieves values of the MfgGuideline object attributes

service **SetWarningMessage**
• set values of the MfgGuideline object attributes

service **DeleteMfgGuideline**
• disconnects and deletes the MfgGuideline object

service **BeginGuidelineEvaluation**
• repeat until all MfgGuideline objects evaluated
  - evaluate guideline rule (via EvaluateGuideline)
  - if guideline rule evaluates true then
    store warning message
  - endif
• end repeat

service **EvaluateGuideline**
• parse guideline rule into postfix token list
• repeat until end of token list
  - if operand then
    get token value
    push value onto stack
  - if number or string value then
    push value onto stack
  - if unary operator then
    pop off last item in stack
    evaluate unary operation
    push result onto stack
  - if binary operator then
    pop last 2 items of stack
    evaluate binary operation
    push result onto stack
• end repeat
• pop evaluation result and return value

3.2.3 Yield & Rework Analyzer

The yield and rework analyzer will provide the capability to calculate yield and rework rates for a selected process sequence associated with a product design. This capability must provide for calculation of the yield or rework rate on an operation level within the process sequence. The rate will be calculated based on the design features' influence on the operation. The yield or rework rate for each design feature associated with an operation will be calculated using either of the following techniques:

- A look-up table that will select the rate based on the value of a design feature. The table will be structured to include the operation number, the design feature, the feature value, and the scrap rate. The scrap rate is equal to (1- yield). An example yield look up table is as follows:

Yield Data:

Operation No.	Design Features	Value	Scrap Rate
180	"aspect ratio"	5.0	0.05000
180	"aspect ratio"	4.0	0.02000

- Through the evaluation of an equation to calculate the rate where the equation may include product design parameters. The slope of yield curve could be expressed as an equation for example.

The total yield or rework rate for an operation will be calculated by treating the contributing scrap or rework rates as non-mutually exclusive independent events.

Following are the services and corresponding descriptions which make up the YieldReworkAnalyzer class-&-object.

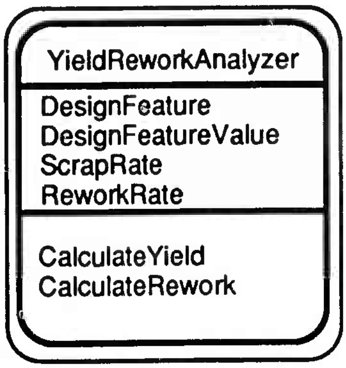

Figure 3-3. Yield & Rework Analyzer OOA Model

specification **YieldReworkAnalyzer**

service **CreateYieldReworkAnalyzer**
• creates and initializes a new YieldReworkAnalyzer object

service **GetYieldReworkAttributes**
• retrieves values of the YieldReworkAnalyzer object attributes

service **SetYieldReworkAttributes**
• set values of the YieldReworkAnalyzer object attributes

service **DeleteYieldReworkAnalyzer**
• disconnects and deletes the YieldReworkAnalyzer object

service **CalculateYield**
• repeat until end of operation list
  - if scrap rate for current operation then
    calculate sum of rates
    calculate product of rates
    calculate probability of scrap rates
    i.e. yield = 1 - (sum of scrap rates - product of rates)
  - endif
• end repeat

service **CalculateRework**

- repeat until end of operation list
    - if rework rate for current operation then
        rework = sum up rates
    - endif
- end repeat

3.2.4 Cost Estimator

The cost estimator provides the capability to calculate the recurring manufacturing cost for each operation of the process sequence. The following calculations will be performed:

- Labor standards for each operation will be calculated for setup and run time categories. The value for each of these categories will be calculated through the evaluation of an equation. The equation may include design parameters. Each category will have an associated labor grade or bid code for each operation.

- Estimated ideal cost for each operation will be calculated from labor standard values multiplied by the wage rate of the labor category performing the operation, and the production efficiency value for that operation.

- Rework operations will be calculated based on the rework rate determined by the yield and rework analyzer multiplied by labor standards for the rework condition. The labor grade wage rates and production efficiencies would then be applied.

- For each operation, the estimated actual cost will be calculated by multiplying the estimated ideal cost by the number of units processed, including both good and scrapped units. The number of units processed by each operation will be calculated from the value of the required good units at the subsequent operation divided by the yield at the operation under evaluation.

- The total estimated ideal cost and total estimated actual cost for each process sequence will be calculated by summing the individual operation cost of each. The estimated actual cost for a good unit will be calculated by dividing the total estimated actual cost for the process by the number of good units produced.

Following are the services and corresponding descriptions included in the CostEstimator class-&-object.

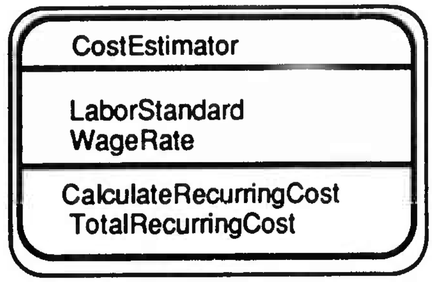

Figure 3-4. Cost Estimator OOA Model

specification **CostEstimator**

service **CreateCostEstimator**
• creates and initializes a new CostEstimator object

service **GetCostEstimatorAttributes**
• retrieves values of the CostEstimator object attributes

service **SetCostEstimatorAttributes**
• set values of the CostEstimator object attributes

service **DeleteCostEstimator**
• disconnects and deletes the CostEstimator object

service **CalculateRecurringCost**
• repeat until end of operation list
  - calculate time (via ProcessEquipment.CalculateTime)
  - calculate yield (via YieldReworkModeler.CalculateYield)
  - calculate rework (via YieldReworkModeler.CalculateRework)
  - calculate recurring cost (via CostEstimator.CalculateRecurringCost)
• end repeat

• repeat until end of operation list
  actual quantity = ideal quantity/yield
• end repeat

• repeat until end of operation list
  recurring cost = actual quantity * actual time
• end repeat

service **TotalRecurringCost**
• repeat until end of operation list
  total recurring cost = current sum of cost + current operation cost
• end repeat

3.2.5 Manufacturing Advisor

The manufacturing advisor provides the capability to view the results produced by each process participating in an analysis. The advisor will include the following capabilities:

- Provide viewing capabilities for single process analysis results including process sequence, yield and rework, cost, and guidelines.
- Provide a mechanism for comparing and displaying the results from two runs of an analysis on a single process sequence.
- Provide the capability to summarize design features causing manufacturing guideline violations across multiple processes. Report recommendations on these guideline violations.
- Provide a summary report, identifying cost drivers, for each process contributing to a multi-process analysis for a given design database.

Provided below is a sample report generated from the Manufacturing Advisor based on the analysis results of the Process Analyzer for a PWB Fabrication process and the corresponding Yield and Rework Analyzer report.

Fabrication Process Selection/Cost Estimation Report

MLB - layers 1, 14 OVERALL YIELD is 94 percent

Opno	*Description*	*Ideal($)*	*Actual($)*	*Rework($)*	*Yield*	*Rework*	*# Units*
10	mark part no	0.123	0.12	0.00	100	0.000	137
20	pierce tooling	0.123	0.12	0.00	100	0.000	137
30	oxide treat	1.111	1.11	0.00	100	0.000	137
40	bake panels	C.444	0.44	0.00	100	0.000	137
50	lay up	3.123	3.12	0.00	100	0.000	137
60	laminate	0.600	0.80	0.00	94	0.000	137
70	lamination	3.123	3.12	0.00	100	0.000	128
80	route excess	0.715	0.92	0.00	100	0.000	128
90	oxide strip	0.250	0.32	0.00	100	0.000	128
110	drill tooling	0.220	0.28	0.00	100	0.000	128
120	pin panels	0.184	0.24	0.00	100	0.000	128
130	drill	12.123	15.12	1.23	92	0.005	128
140	smear remove	3.123	3.12	0.00	100	0.000	117
150	pressure	0.888	0.89	0.00	100	0.000	117
160	electroless	0.661	0.66	0.00	100	0.000	117
170	copper panel	0.555	0.55	0.00	100	0.000	117
180	electrostrike	0.512	0.70	0.00	98	0.000	117

Fabrication Yield Analysis Report

MLB - layers 1, 14 OVERALL YIELD IS 94 percent

Opno	*Design Feature Description*	*Value*	*Scrap Per Feature*	*Opno Yield*
60	14 layers and 8 substrates	N/A	6.000	94
130	annular ring	8.00	8.000	92
180	aspect ratio	4.00	2.000	98

Following are the services and corresponding descriptions included in the MfgAdvisor class-&-object.

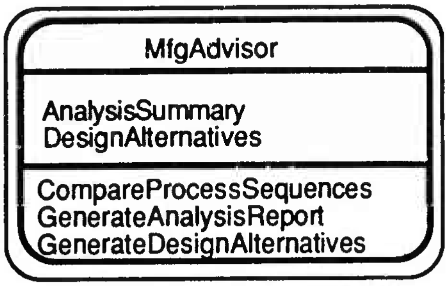

Figure 3-5. Manufacturing Advisor OOA Model

specification **MfgAdvisor**
objectState AnalysisSummary = Individual or Team

service **CreateMfgAdvisor**
• creates and initializes a new MfgAdvisor object

service **DeleteMfgAdvisor**
• disconnects and deletes the MfgAdvisor object

service **CompareProcessSequences**
• compare two process analysis runs
• generate a summary of the differences between the two

service **GenerateAnalysisReport**
• retrieve and print out design features and values
• retrieve and print out process operation sequences
• retrieve and print out the yield and rework summary
• retrieve and print out recurring cost summary

service **GenerateDesignAlternatives**
• retrieve and print out a consolidated set of design alternatives

3.3 User Interaction Requirements

The MO system will allow the manufacturing specialists to capture and maintain multiple copies of process data models through a set of utilities, which will provide direct access to the product design features, process logic flows, process selection rules, manufacturing guidelines, yield/rework, and labor standards. Through the use of these utilities, the process team will have the ability to modify the process model data to explore alternative process approaches and plan process improvements, and then analyze the effects of these changes on the product design cost.

The users of the MO system will be able to initiate the following analysis capabilities: selection of manufacturing process model for analysis to be based on, perform process selection(s), perform guideline evaluations, generate analysis reports, compare process analysis results, and generate consolidated design alternatives report.

3.4 Internal Interface Requirements

During process selection, the process analyzer capability utilizes the internal product design data structures while evaluating the process selection rules. The data structures created in the process analyzer that contains the selected process operations will be used when executing the yield and rework analyzer and cost estimator capabilities.

The guideline analyzer will utilize the internal product design data structures during the evaluation of the guideline rules.

The output data structures from the process analyzer, guidelines analyzer, yield and rework analyzer, and cost estimator will be used as input to the manufacturing advisor capability.

3.5 Data Element Requirements

Listed below are the MO Class-&-Objects and their corresponding attributes.

specification **ProductDesign**
  attribute **DesignName**: Product Design Name
  attribute **Geometry**: part features, attributes, and corresponding values
  attribute **Materials**: list of part material
  attribute **Size**: part dimensions
  externalInput **CadDataExtraction**: Input from CAD Product Data Neutral Area
  externalOutput **CadDataRequest**: Request for Data Extraction from Neutral Area

specification **Process**
  attribute **SelectionRule**: reason for selecting the process

specification **StepProcess**
  attribute **StepNumber**: Step Sequence Number
  attribute **StepDescription**: Step Text Description
  attribute **StepTool**: Tool used at the Step

specification **OperationProcess**
  attribute **OpNumber**: Operation Number
  attribute **OpDescription**: Operation Text Description

specification **ProcessEquipment**
attribute **SizeCapacityParameters**: maximum machine size capacity
attribute **SetupTime**: Machine Setup Time
attribute **HandlingTime**: Handling Time of Part
attribute **MachineTime**: Run/Machine Time to Perform Operation
attribute **ProductionEfficiency**: Mfg K_factor or efficiency rate

specification **YieldReworkAnalyzer**
attribute **DesignFeature**: associated design feature for scrap/rework rate
attribute **DesignFeatureValue**: corresponding design feature value
attribute **ScrapRate**: scrap/yield percentage
attribute **ReworkRate**: rework percentage

specification **MfgGuideline**
attribute **Guideline**: manufacturing guideline rule
attribute **WarningMessage**: corresponding guideline warning message

specification **MfgAdvisor**
attribute **AnalysisSummary**: analysis report details (individual or team)
attribute **DesignAlternatives**: process team design recommendations

specification **CostEstimator**
attribute **LaborStandard**: labor grade or bid code
attribute **WageRate**: dollars per hour of the laborStandard

4. Measure of Performance

4.1 Introduction

It is the goal of this program to develop a system that performs Design For Manufacturability and Assembly in a concurrent engineering environment. This implies that multiple manufacturing engineers will be able to simultaneously analyze a design, compare or merge results, and conduct trade-off studies that will result in a consolidated and comprehensive evaluation of the manufacturability of the product.

Raytheon will capture metrics against the demonstration vehicle, an original R&D design modified for manufacturability, as described in section 4.4. For the original design, we have data on the initial design parameters, the manufacture of the validation build, and review. The original design was altered based on the producibility review and is believed to have enhanced producibility. The modified design parameters are known and manufacturing data is being collected during Proof-of-Manufacturing build. To demonstrate the performance and benefits of the proposed system we will compare the design features, actual manufacturing costs, and producibility review recommendations of the original design against an analysis of that design using MO. We will then compare those same attributes for the new design against a MO analysis. Finally, we will compare MO guideline/recommendations on the original design to the modified design (the selected alternative). The system that facilitates this process should accurately estimate manufacturing costs and yields, and flag guideline violations.

The following sections, which are an overview of the typical design and manufacturing cycle for a standard through hole printed wiring board, highlight the inter-relationships among design and manufacturing processes and provide a baseline for examining the role of Design for Manufacturing and Assembly in a concurrent engineering environment.

4.2 PWB Design Process

A typical high level design flow for printed circuit boards, excluding revision cycles, consists of the following steps: packaging concept, design capture, design analysis/verification, component placement, interconnect routing, and documentation/transition to manufacturing. Figure 4-2 depicts the high level printed circuit board design flow. Each step will be described in terms of the functions performed and the design features, or attributes, determined.

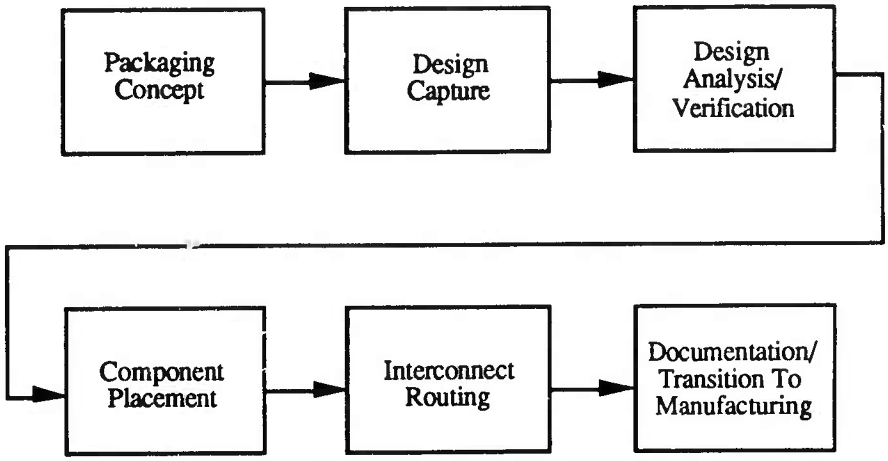

Figure 4-1. Printed Wiring Board Design Flow

During the concept phase, the product team establishes a packaging concept based on the system requirements specification. The concept is reviewed for functionality, performance, fit, power and thermal, cost and schedule, reliability, manufacturability, and test. Among the attributes determined at this stage are: number of required board types, board geometries (area, length, width, aspect ratio), number of interconnect and power/ground layers, layer stackup, layer-to-layer spacing and tolerances, PWB materials, design rules (trace widths, spacings, and via sizes), component family and attachment method, preferred parts, thermal management, test strategy, and cost, performance and reliability budgets. Thus, a packaging baseline is established, although many of these parameters may be challenged and re-evaluated during the detail design process.

At design capture the design details are defined, the electrical schematic is captured, and the parts list is specified. This process implicitly determines circuit complexity, packaging density, operating frequency, and numerous attributes associated with the selected components including auto-insertability, attachment method, mounting hardware, and thermal, static, and noise sensitivity.

Design analysis and verification commence upon completion of design (schematic) capture. These processes are inherently iterative with schematic capture, since refinements are made to the design as a direct consequence of design analysis and verification. Depending upon its characteristics, the design may be subjected to circuit analysis, digital simulation, loading, timing, and signal fidelity analyses, thermal, stress, and reliability analyses, testability analysis, and a monte carlo tolerance analysis. Critical signals and timing margins are identified.

Preliminary test requirements are defined. A design review is conducted, with representation from the design, manufacturing, test, and quality assurance communities.

Component placement is often initiated by the circuit design engineer, and later refined by the layout draftsman. Approval by the circuit designer is always required prior to initiating detailed layout. Crucial to the placement function is the need to achieve an effective balance between circuit performance and thermal behavior. Thus, critical paths must be kept short, while reactive and hot components must be kept apart. Density, timing simulation, and thermal analysis tools are frequently invoked to identify conflicts even prior to detailed interconnect routing. Manufacturability analysis tools are also employed to assess compatibility with automated component insertion equipment and to evaluate the cost/yield implications of alternative layer stackups and interconnect design rules. During this process, part substitutions may be evaluated in order to resolve packaging, reliability, and manufacturability problems. Upon conclusion of the placement process, component locations and orientation have been determined and the required parts list substitutions have been implemented.

Next, detailed interconnect routing is performed, beginning with critical paths. Routing may be automatic, manual, or a combination of both, subject to design characteristics. Subject to complexity, design rules may have to be re-evaluated, stackups altered, and components moved. Upon completion of the routing process, the final interconnect topology is determined, including trace width, feature spacing, pad/hole sizes, via sizes, conductor density and distribution, number of layers, layer stackup, and final component locations and orientation. Design verification is achieved through continuity checking against the schematic data base, layout effects analysis, and physical design rule checking.

Finally, a technical data package is generated in preparation for release to manufacturing. This consists of a manufacturing compliant design data base, documentation, and post-processed manufacturing outputs, if requested. Specific data items include schematics, parts list, master artwork, drill drawing, and n/c tapes. Detailed checking and a multi-disciplined review of the technical data package precedes a formal release to manufacturing.

4.3 PWB Manufacturing Process

Figure 4-1 depicts a typical manufacturing process flow for printed wiring boards. At each step in the flow there are design attributes that influence the manufacturing process. Four of the manufacturing processes will be described as highlighted in the figure.

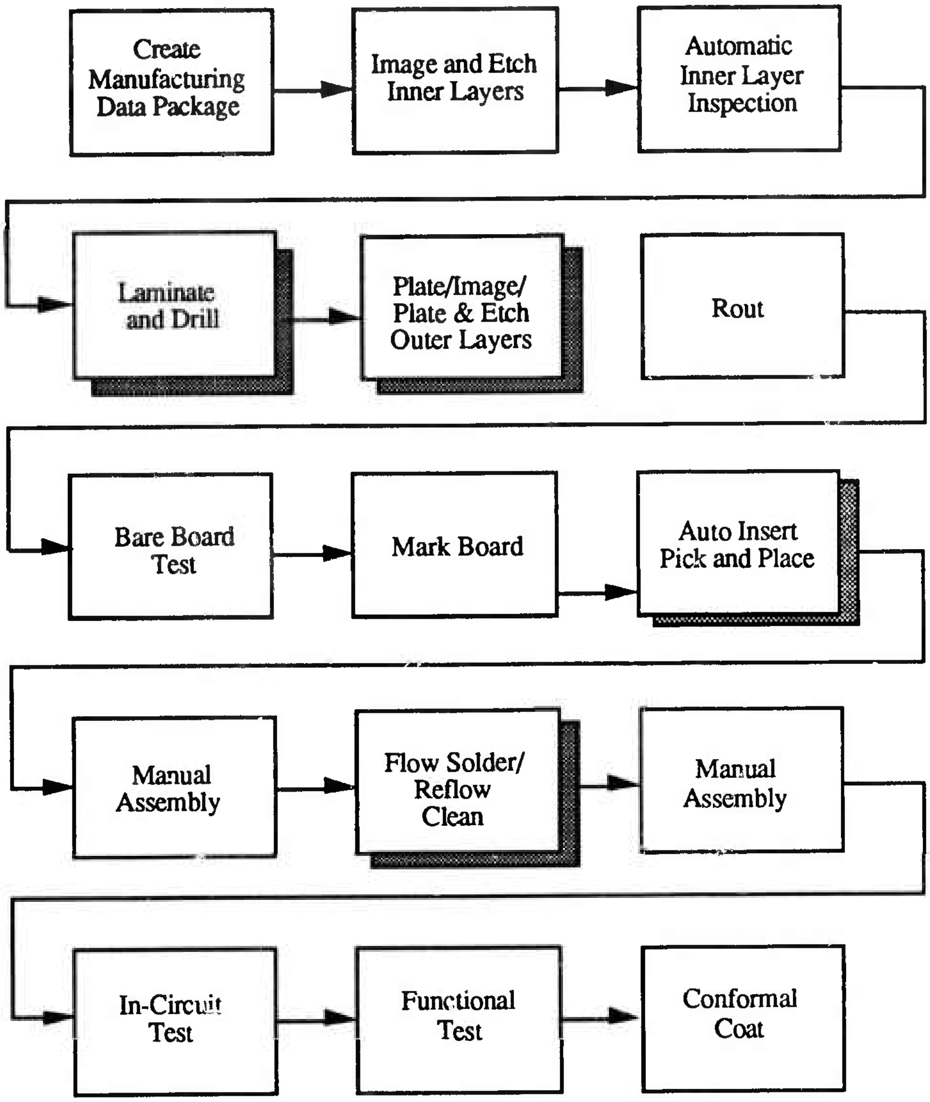

Figure 4-2. Printed Wiring Board Manufacturing Flow

Laminate and Drill are the processes where multiple circuit layers are laminated under heat and pressure, and the required holes are automatically precision drilled. The critical design attributes, which influence the lamination process include: blind/buried vias, number of layers, copper balance, layer stackup, board/laminate thickness, impedance control requirements, laminate/prepreg material, and board dimensions.

While the effect of design features upon the Laminate and Drill processes are many, one of the most profound influences upon cost and yield comes from the requirement for blind and/or buried vias. Traditional through-hole PWB construction requires a single lamination step, followed by precision drilling of the through holes. However, boards requiring blind vias (i.e., visible from one external surface only) or buried vias (i.e., completely blind from both external surfaces) will require multiple lamination, drilling, imaging, plating ... lamination, drilling ... steps. Additional costs include tooling, inspection, yield loss due to via fracturing, and multiple setups.

A more common occurrence is the adverse influence of non-uniform copper balance upon lamination, whereby warping of the laminated structure occurs, causing via and component attachment failures. Efforts to minimize warping result in added cost and are not always successful.

A third critical design attribute is that of impedance control. Stringent control of interconnect impedance is dependent upon distance between interconnect layers and their associated copper planes. This requirement translates into non-standard layer stackups and more precise control of the lamination processes.

Yet a fourth is the relationship between pad size, drill diameter, and board thickness whereby the requirement for close tolerance drilling of many "large" holes in small pads, through thick panels, results in high inspection costs and low yields.

The Image, Plate, and Etch Outer Layers processes involves photographic imaging and chemical plating/etching operations. The critical design attributes, which influence these processes include: plated through-hole (PTH) diameter, PTH aspect ratio, available registration aids, feature sizes, spacing and tolerances, material selection, layer stackup (presence of interconnect on outer layers, position of ground planes, metal balance/density, outer laminate copper thickness), and length of parallel interconnect lines.

Again, many design feature/manufacturability relationships exist. One of the more notable dependencies is the aspect ratio of plated-through-holes whereby the greater the barrel length:hole-diameter (aspect) ratio, the more difficult it becomes to deposit copper inside the barrel. Proper plating and copper adhesion are imperative if the board is to ultimately withstand temperature cycling and vibrational stress.

The Auto Insertion/Pick and Place processes involve automatic insertion of through-hole components and the attachment of surface mount devices. The critical design attributes, which influence the auto insertion process, include: insertability of each component type, number of components by insertion type, component orientation by component type, component-component spacing, component-obstruction spacing, board thickness versus component lead length, lead diameter versus hole diameter, static sensitivity, sequencer compatibility of components, and component bonding/attachment method.

Fundamental to automated insertion is the inherent insertability of specific component types. A secondary, but similarly important criteria is the number of components of a given class (e.g., dual in-line IC or surface mount packages) that ultimately determines the economic feasibility of allocating and setting up a specialized insertion machine for a limited number of components. Even were the component type and count meet the required criteria, component orientation and component spacing become important factors. The best case occurs where all components are aligned in the same direction, less attractive is where components of a given type occur at zero and ninety degree rotations, and worst case is where components are aligned off-axis (i.e.,, 45 degrees or other).

The Flow Solder process is where through-hole components are soldered onto the board using a "wave" of molten solder and surface mount components are reflowed using a vapor phase, IR, convection, or combination. The critical design attributes, which influence the flow solder process include: board thickness/lead protrusion, thermal sensitivity of components, metal balance, component orientation, board geometry, presence of interconnect or ground plane on solder side, pad geometry, plated through-hole lead diameter, and aspect ratio of through-hole.

Thick boards, coupled with short lead protrusion, and improperly sized/shaped pads on the solder side create lead clinching problems. Pads that do not have proper thermal relief with respect to copper power and ground planes become heat sinks and promote poor solder joints. Conductor paths on the solder side cause solder bridging. Heat sensitive components require special handling and solder-side ground planes cause a variety of problems associated with heat absorption.

4.4 Test Case Demonstration Vehicle

The demonstration vehicle for this program will be modules from the Patriot System's Expanded Weapons Control Computer (EWCC). This system is currently under development. When deployed, it will provide substantial upgrades to the performance of the Patriot System.

Four years ago a R&D program was conducted to determine the system architecture, detailed electronics design, and potential packaging approach for the EWCC. One system was fabricated to validate these concepts.

The system packaging concept stressed packaging density. The packaging technology selected for this system utilized leadless hermetic chip carriers on double sided modules. These modules consisted of two PWB's bonded back-to-back on a composite core.

Each PWB consisted of 13 layers: 2 outer pad layers, 7 interconnection wiring layers, and 4 power and ground planes. The wiring layers were configured as controlled impedance microstrip layers in order to support high speed digital signals. Very small feature sizes were utilized. For instance, the line widths were 0.005 ± 0.001 inch, the plated through hole diameter was 0.016-0.022 inch, and the inner layer land diameter was 0.030 inch. These boards had small features, small spacings, and tight tolerances throughout. They were difficult to manufacture.

In order to improve the producibility of the printed wiring boards, some fundamental changes were made to the packaging approach. First, the electronic devices were changed to leaded hermetic chip carriers. The modules were still double sided, but the utilization of leaded packages permitted simpler assembly. Second, the inner configuration of the printed wiring boards was changed from microstrip to stripline.

The boards now consisted of 2 outer pad layers, 6 interconnection wiring layers, and 4 power and ground planes. This, combined with the change to the leaded packages, permitted increasing feature sizes, spacings, and tolerances. Now the line widths were 0.006 ± 0.002 inch, the plated through hole diameter was still 0.016-0.022 inch, but the inner layer land diameter was increased to 0.052 inch. Spacings and tolerances were relaxed.

The EWCC module test case will be loaded in to MO and an analysis/optimization session will be run to demonstrate the ability of MO to advise the users, accurately estimate costs and yields, and compare two design alternatives. Each version of the design concept will be run and the analysis data compared against actual data.

5. Notes

5.1 Acronyms

CAEO	Computer Aided Engineering Operations
CDRL	Contract Data Requirements List
DARPA	Defense Advanced Research Projects Agency
DBMS	Database Management System
DFMA	Design for Manufacturing and Assembly
DICE	DARPA Initiative In Concurrent Engineering
MEL	Mechanical Engineering Laboratory
MO	Manufacturing Optimization
MSD	Missile Systems Division
MSL	Missile Systems Laboratories
PWA	Printed Wiring Assembly
PWB	Printed Wiring Board
PWF	Printed Wiring Fabrication

Appendix I — OOA Modeling Notations

10. OOA Model Representation

The Object-Oriented Analysis (OOA) approach to requirements analysis consists of five major activities: finding Class-&-Objects, identifying structures, identifying subjects, defining attributes, and defining services. The OOA model reflects the problem domain and system responsibilities.

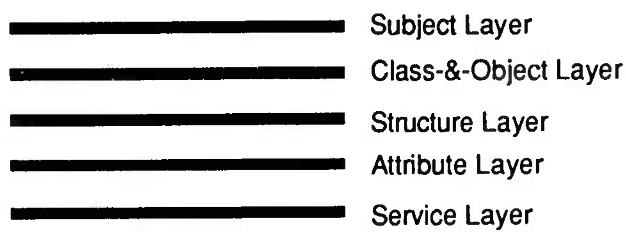

Figure I-1. Multi-Layer OOA Model

10.1 Class-&-Object Layer

The Class-&-Object Layer is used to match the technical representation of the system to the conceptual view of the real world. Class-&-Objects represent an initial expression of the problem domain. The Class symbol is used to represent a generalization of Objects that are portrayed by specializations with Class-&-Object symbols. Figure I-2 shows the Class-&-Object and Class symbols.

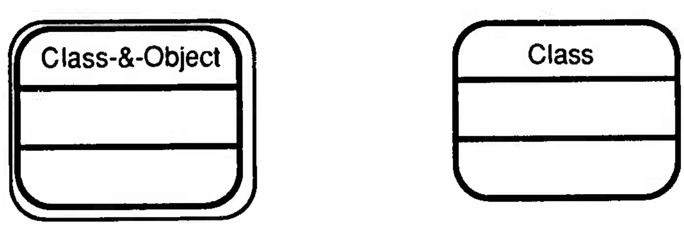

Figure I-2. OOA "Class-&-Object" and "Class" Symbols

10.2 Structure Layer

10.2.1 Generalization-Specialization Structure

The generalization-specialization (gen-spec) structure is used to express the complexity in the problem domain. The gen-spec structure can be thought of as an "is a" or "is a kind of" structure (i.e. OperationProcess is a (is a kind of) Process). The gen-spec structure is the vehicle to representing inheritance.

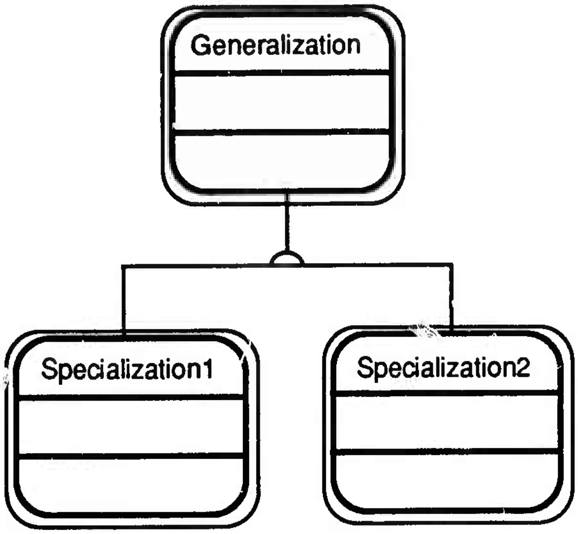

Figure I-3. Gen-Spec Structure Notation

10.2.2 Whole-Part Structure

The Whole-Part structure is used to represent a "has a" structure (i.e. a Vehicle has an Engine). Each whole-part structure line is marked with a range that indicates the number of parts that whole may have, as well as how many wholes a part belong to at any point in time. Figure I-4 depicts a Whole can have one or more Part1 and Part2, and Part1 and Part2 can belong to at most one Whole.

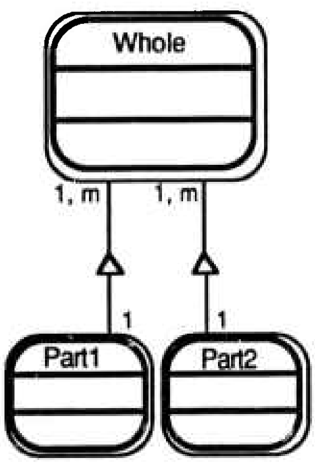

Figure I-4. Whole-Part Structure Notation

10.3 Subject Layer

The Subject Layer is used to assist a reader through a large, complex model, as well as to organize a project into work packages.

Figure I-5. Subject Notation, Collapsed

10.4 Attribute Layer

10.4.1 Attribute Notation

Attributes are used to add detail to the "Class-&-Object" and "Structure" abstractions. They describe values (states) kept within an Object which are exclusively manipulated by the services.

Figure I-6. Attribute Notation

10.4.2 Instance Connection

Instance Connections add to the state of Objects by adding the required problem domain mapping that one Object needs with other Objects to fulfill its responsibilities. An Instance Connection is shown with a line drawn between Objects. The ends of each Instance Connection are marked with an amount (m) or range (m,n) that reflect the constraints on the objects.

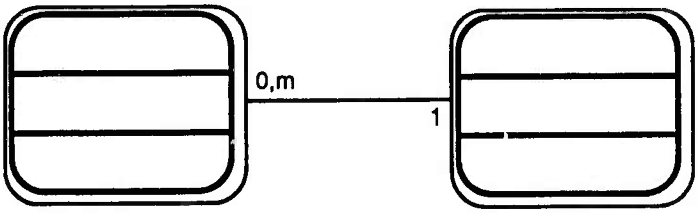

Figure I-7. Instance Connection Notation

10.5 Service Layer

10.5.1 Service Notation

The Service Layer adds the system's processing and sequencing requirements. Services further detail the Class-&-Object abstraction of the problem domain being modeled by indicating what behavior will be provided by an Object within the Class.

Figure I-8. Service Notation

10.5.2 Message Connection

Message Connections depict the mapping of one Object to another where the sender "sends" a message, the receiver "receives" the message, and the receiver takes some action and returns the results to the sender. These message connections exist solely for the benefit of the services.

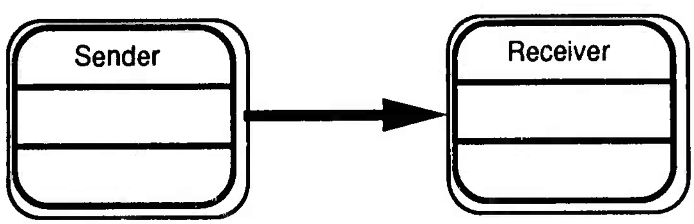

Figure I-9. Message Connection Notation

20. Class-&-Object Specification

The Class-&-Object specification is the method used in OOA to fully specify each Class-&-Object. The specification contains attributes, external interfaces, object state diagrams, any additional constraints, notes, and applicable service descriptions, in bullet list formats, for each class-&-object service.

specification
  attribute
  attribute
  attribute
  externalInput
  externalOutput
  objectStateDiagram
  additionalConstraints
  notes
  service <name & Service Bullets>
  service <name & Service Bullets>
  service <name & Service Bullets>
  and, as needed,
    traceabilityCodes
    applicableStateCodes
    timeRequirements
    memoryRequirements

Figure I-10. Class-&-Object Template

Appendix II— MO OOA Model

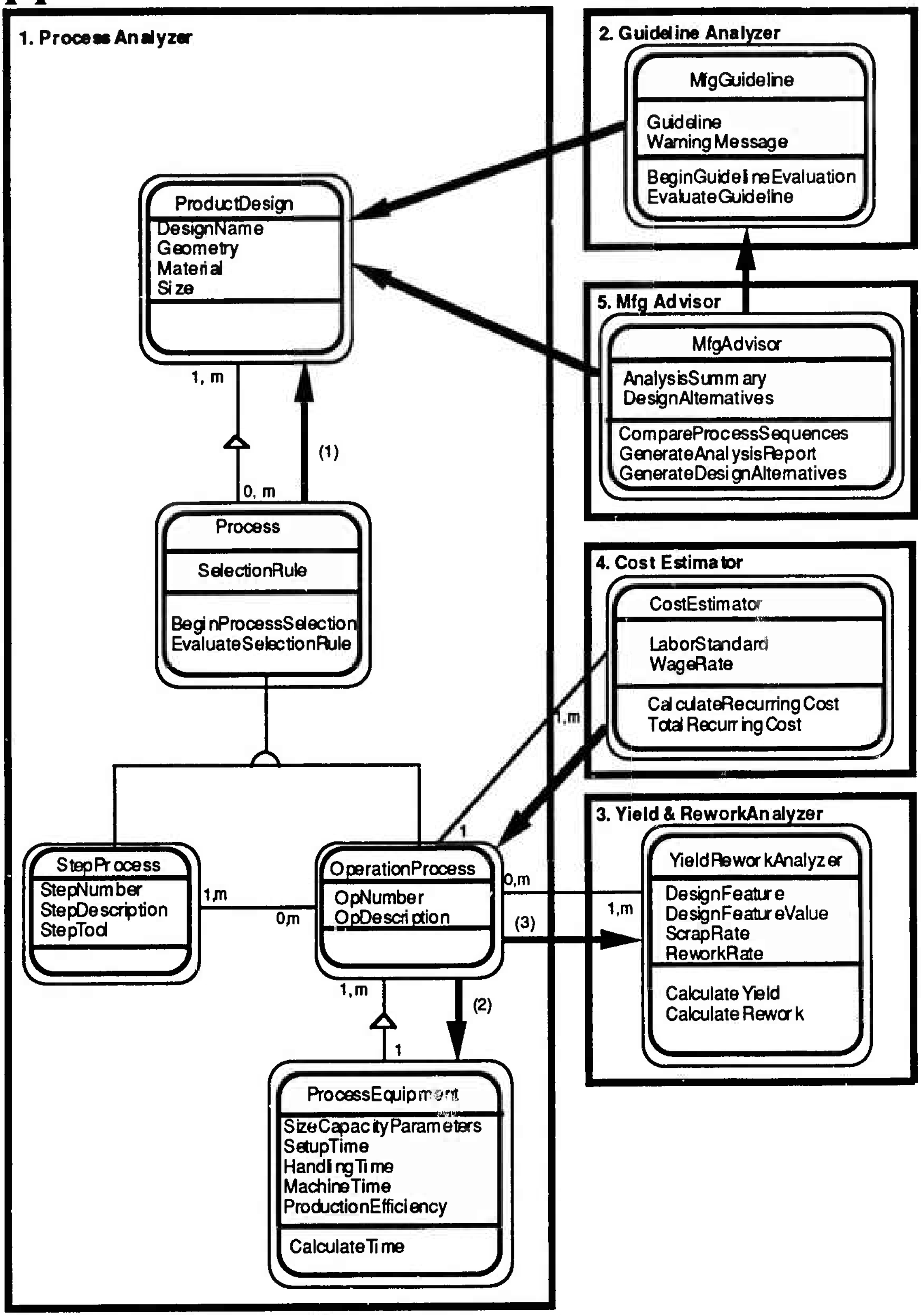

Figure II-1. MO OOA Diagram

Distribution List

DPRO-Raytheon
C/O Raytheon Company
Spencer Lab., Wayside Ave.
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Defense Sciences Office; Dr. H. Lee Buchanan
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Electronic Systems Technology Office; Capt. Nicholas J. Naclerio, USAF
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Advanced Research Projects Agency
ATTN: Contracts Management Office; Mr. Donald C. Sharkus
Virginia Square Plaza
3701 N. Fairfax Drive
Arlington, VA. 22203-1714
(one copy of each report)

Defense Technical Information Center
Building 5, Cameron Station
ATTN: Selections
Alexandria, VA 22304
(two copies of each report)