CHAPTER 1

System-based concurrent methodology for Discrete Part Manufacturing Engineering

Muriel Lombard-Gregory, Frédérique Mayer
CRAN-EACN, Université de Nancy I
54506 Vandoeuvre-lès-Nancy Cedex - France

Design Management and Manufacturing System concept, reduced to Corporate Information Management, put in place a **concurrent methodology** which couples mechanisation and automation synergies for discrete part manufacturing systems specifications, in a manner which allows a cooperation and a quasi-reactivity of the project's actors. Precepts extended to **General System theory** allow to specialize a **systemic model** for discrete part manufacturing. Particularly, we will demonstrate how different skills enlarge by their own expert valuation of the system's global conception by detailing nerve-centres of the concurrence.

1.1. INTRODUCTION

The set up of a Concurrent Engineering reduces ***Time to Market*** and improves both design and product quality in contrast to the classical linear approach [TIC 91].

This new method proposes an evolution of the **C.I.M.** (*Computer Integrated Manufacturing*) concept aiming to consider integration of all enterprise activities towards a *Computer Information Management* promoting information management rather than data treatments, then towards a *Corporate Information Management* promoting knowledge communication rather than syntactical data communication.

According to this approach, Concurrent Engineering may be an answer to manage enterprise corporations. Yet this matter is full of complexities. It must be emphasized that this approach puts in the same time and space all actors to achieve the product's design, manufacturing and management during its entire life cycle [PRA 93].

Consequently, our *Design Management and Manufacturing System* concept (D.M.M.S.), reduced by *Corporate Information Management* in a Concurrent Engineering context, consider integration and communication of framework knowledge.

R. Soenen et al. (eds.), *Advanced CAD/CAM Systems*

1.2. D.M.M.S. REFERENCE ARCHITECTURE

Historically, the D.M.M.S. Concurrent Engineering objective [MOR 92] is an extension of the CIMExpert training package idea [GER 92], reduced to design and simulate the process planning for outer profiles. Given that these functions are in sequential order, the first prototype was a partial validation, although all the software components were integrated.

Thus, our D.M.M.S. architecture (Figure 1.) proposes a broader concept :

1 coupling both mechanical and automation skills with design, manufacturing and management points of view to manufacture a discrete part,

2 semantically integrating, on top of a common technical Management D.M.M.S. Station, a Mechanical and an Automation Working Station for the Design function with a Maintenance Working Station and an Open Manufacturing Cell for the Operating function, each of these working stations being composed of a set of C.A.X. (*Computer Aided X.*) tools.

In this way, two kinds of exchanges can be outlined:

1 ***intra-corporation*** which corresponds to skill decision making,

3 ***inter-corporation***, exchanges between different corporations which correspond to system decision making and involves that information potentially available to all.

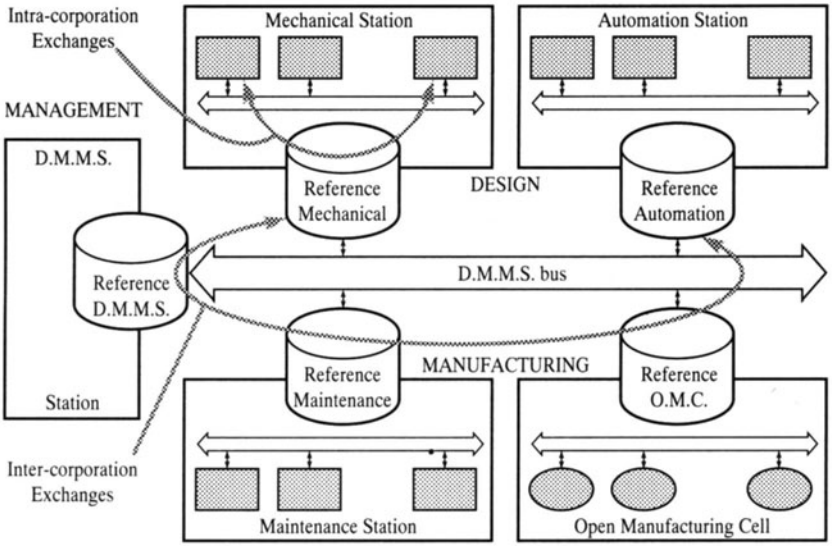

Figure 1. D.M.M.S. Referential Architecture [LOM 93]

Our study is focused toward building the modelisation of this system and to

demonstrating how and from where these inter-corporation exchanges come.
This extended and versatile Information Processing System requires :

1 for object and/or data sharing, a common repository between the heterogeneous C.A.X. tools,

2 for object and/or data processing, a coordinated cooperative methodology between the heterogeneous skills, which bring a standardized "*micro-integration*" within each working station as well as a "*macro-integration*" within the technical management station.

So, our D.M.M.S. Concurrent Engineering architecture is a first *repository* for exchanges between different skills, and it is necessary to employ methodology to have a semantic and systemic guide for specification and implementation of discrete part manufacturing systems.

1.3. REFERENTIAL MODEL FOR DISCRETE PART MANUFACTURING SYSTEMS

Many modelling methods, characterised by analytical methods, perceive complex systems as being complicated, namely reducible to models, themselves being complicated yet capable of simplification and potentially furnishing a basis for their automatisation [MOI 90]. However, the inadequacy of those systems' models can be established at the time of application for the complex phenomenons' representation because it only describes their internal structure and not their finality. Thus, in the context of Discrete Part Manufacturing Systems, there is only one essential question : "*Where does the product (finality of a manufacturing system) stand ?*".

Presently, our solution to this problem is to apply another ***complementary but not contrary*** approach [ROS 75] to the analytical one, the systemic approach, considering phenomenons as complex and propounding to model them in a way to ***build their intelligibility*** and to obtain a system automatisation of Quality.

1.3.1. Systemic precepts

The Systemic theory gives the only foundation for concept formalization of systems. Indeed, to model complex systems and, contrary to the analytical modelling, the systemic approach aims to modelize the actions' system and not more objects. A process, which can be easily linked to the notion of action, is defined when, during the ***time***, there is a modification of attitude, in a referential *<Space-Shape>*, of a lot of products identifiable by their ***Shape***. In this way [MOI 90] proposes to identify process in a referential ***<Time-Space-Shape>*** allowing to define process' canonical model.

In our point of view, usage of the term operator with the sense of holding of the different operations existing in this referential. In the case of works, [GAL 93] introduces a fourth operator named "Nature" and an associated Function, "***Control***" in a way which represents controlled action. The role attributed to this operator is to direct a special transformation, a transmutation, namely a change of the nature of something. In fact, we think that each **Nature** relationship between a given **Manufacturing Function** (Mechanical point of view) and **Control Function**

(Automation point of view) may be considered as a base for a concurrent engineering gateway definition (Figure 3.).

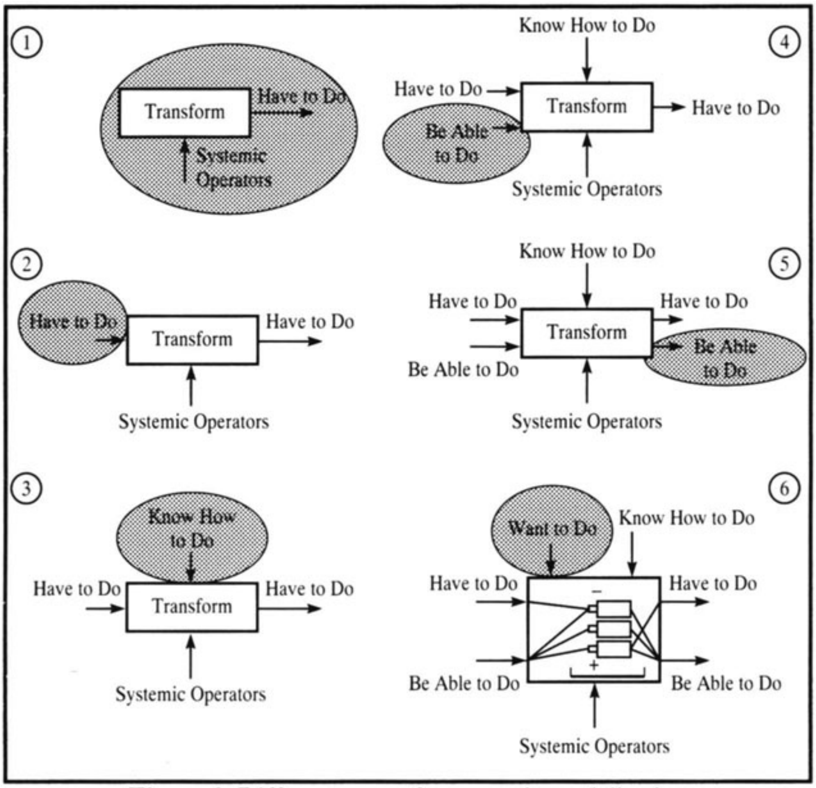

Figure 2. Different stages for systemic modelisation

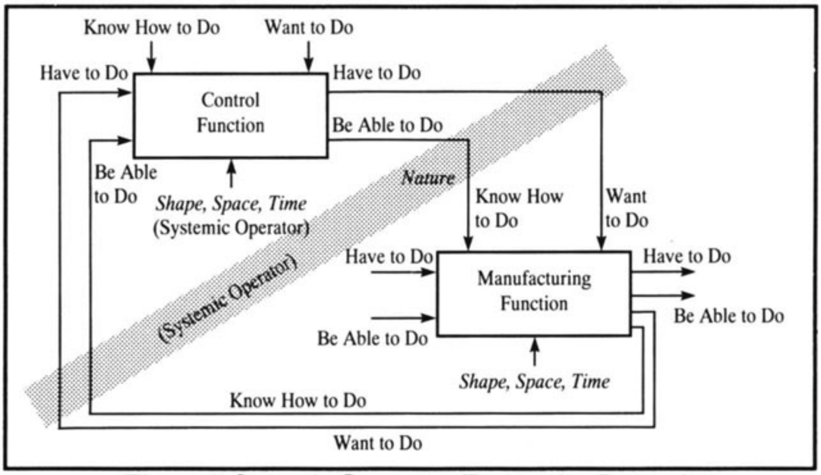

Figure 3. Systemic Concurrent Engineering Processing

1.3.2. Referential model

The various stages in the development of a scenario, according to the systemic methodology, show how different skill stations work in concurrence. To do this, each operator is associated to a **Manufacturing Function**, namely for the component Shape, "***Transform***" (millling, turning, ...), for the component Space, "***Transport***" (moving, convoying, ...) and for the component Time, "***Stock***" or to a **Control Function**, respectively named "***Process***", "***Communicate***", "***Store***". So, those verbs have been choose in relation to terms usually used in our application's fields of Discrete Part Manufacturing Systems.

So, in practice, the organisation of activities must follow a syntagmatic scheme, according to [COQ 89], that is to say along the axis of a sequence of words which correspond to activities considered. In fact, according to this organisation, it is not possible to have two successive activities of the same nature. In this case, it is probable that one activity has been forgotten or one is more complex and requires a decomposition.

From a pratical point of view, our systemic concurrent engineering methodology, dedicated to discrete part manufacturing modelling, starts with the basic Transform Activity (Figure 3.). To describe interactions between these functions, [COQ 89] proposes a modal typology which can be applied to our model with four flows inducing partial behaviors (part behavior, tool behavior, ...) (Figure 3.6) to come closest to explain "*what it has to do*", "*what it knows how to do*", "*what it is able to do*", and "*what it want to do*".

According to [GAL 93], it is fitting to take into account the ***Nature operator*** with which the Control systemic function is associated [MOR 94], to link any flows and finally to obtain a complete system (Figure 3.).

For example, Systemic modelisation of the ***Transform function*** corresponds, in Discrete Part Manufacturing Engineering, to a mechanical point of view to build system architecture that its finality is to transform. Some other system architectures can be described with other points of view (robotic, ...) or by considering *Know How to Do* flow production (Management function, ...).

1.4. CONCURRENT ENGINEERING METHODOLOGY FOR DISCRETE MANUFACTURING SYSTEMS [LOM 94]

Our objective is to propose a logical cell architecture on which manufacturing process planning is executed.

In order to achieve our objective successfully, Concurrent methodology is applied to a realistic example.

1.4.1. Systemic Methodology

1.4.1.1. Example

To illustrate, we apply a simplified example (Figure 4.) : We want to manufacture A1, A2, C, F1. F2 is already manufactured. B1, B2, B3 and B4 are rough features. We have some machines and tools which compose our workshop.

Expert rules define these sequences :

feature (1) :	**Rough(1)**	**Half Rough (1)**	**Finition (1)**	**A1**
feature (2) :			**Finition (2)**	**C**
feature(3) :			**Finition (3)**	**F1**
feature(4) :	**Rough (4)**		**Finition (4)**	**A2**

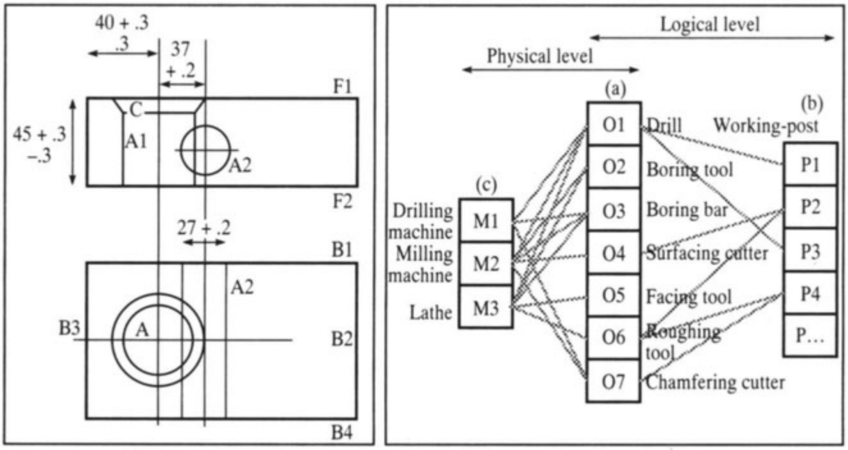

Figure 4. Definition drawing of the part and workshop

The set of elements, above mentioned, allow to start the modelisation of a Manufacturing System, by considering it as a *black engine* fulfilling a mission (Transform, for example) to produce a **finality** (a manufacturing part, for example) and where its **environment** and ***Concurrent Engineering Methodology*** applied determines the four types of flows which will be input from and output to the system interface.

Note that the use of systemic rules leads to achieve modelisation with basic Transport and Stock activities and, consequently, to introduce concurrent engineering potentialities between the automation fields and each of these respective skill fields. This way is recursive for any level and any skill processing.

1.4.1.2. Mathematical formalism

To minimize this problem, it is agreed upon to say that the equation to be resolved, proposed by [FUS 83] and adapted to our problem by [PTA 87] of process planning:

$$\text{Operator} \quad \text{Operation} \supset \text{Process}$$

translates the passage from conditions of contract to finality of process, namely in terms of manufacturing the part, to logical architecture as to part-station by finding systemic operators <Shape-Space-Time/Nature> which caracterize the process.

To do this, we use mathematical formalism with temporal logic : (E) is the set of manufacturing features which is defined as a geometric shape and a set of specifications for which a machining process is known, for which [GAR 92] defines two **algebraic structures**. The first is **(P)** algebra and permits the use on (E) of logical operations **V** and (or, and). The second **(O)** algebra (Figure 6.) is an extension of (P), having in our case a **temporal connector M** with M(A) signifing that A will be true the next time. We shall note **W** as the exclusive or. It is the notation for : A W B W (A B) = A V B.

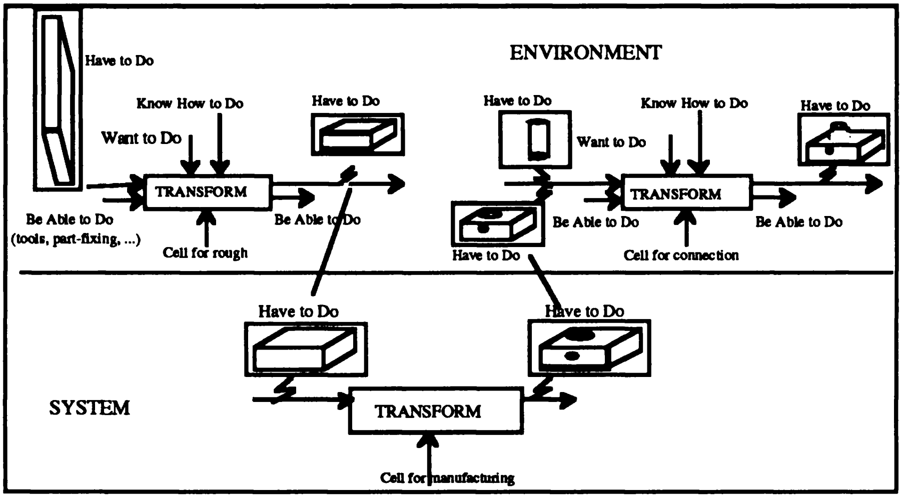

Figure 5. Transform system in relation with its environment

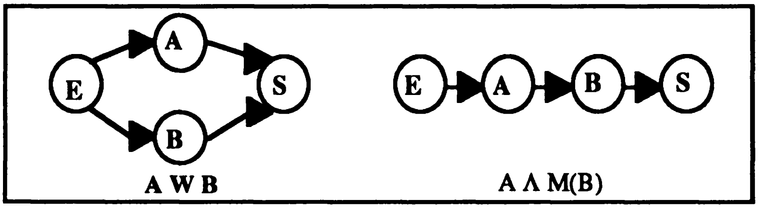

Figure 6. Graphical interpretation of (O) atom

A part i to be manufactured can be described as a set of features which composes an under set of (E) and translates a decomposition of the *Shape* operator.

Pi = {feature(i,1), feature(i,2), ..., feature(i,n)}

So, the finality of our system is, for our part (Figure4.), to describe manufacturing as :

Pf = feature(f,1) feature(f,2) ... feature(f,n)

For each feature identified, **boolean matrix *R*** is applied such as for **A** and **B** of (E) :

Rab = 1 if **A** must be realized before **B**, where **R** represents knowledge of designer.

A function"*f*", for regrouping, allows the transformation of a proposition of (P) in a proposition in the (O) algebraic language. The first structure (P) allows to describe all specifications.

"*f*" is a function from (P) to (O) defined by :

if **Rab** =1	f(A B) → A M(B)
if **Rba** =1	f(A B) → B M(A)
if **Rab** ≠ 1 and **Rba** ≠ 1	F(A B) → A B = (A M(B)) **W** (B M(A))

A	:	feature according to Shape point of view
f(A)	:	feature according to manufacturing point of view f(A) is the process associated to A feature

So, the "*f*" function converts specification into representation with physical structure associated to process planning.

4.1.3. Space/Time

Formalisation of our concurrent engineering problem consists to describe the relation between input and output *Have to Do* flows, which corresponds to a mechanical knowledge, which in our opinion, grows. Indeed, an output *Have to Do* flow is obtained by the combination of input flow and internal behavior that our methodology defines.

For the example under consideration, expert rules put in place are :

- *Do rough shape of bore before chamfer finition*
- *Do chamfer finition before bore finition*
- *Do face finition before bore rough state which is open onto face*
- *Do finition of little bore before rough state of big bore, because there is rule which says "do little bore before big bore when they are tangents"*

These rules permit to apply **R matrix** for making operation sequence with anteriority criterae :

[(F3 Λ M(R4) Λ M^2(F4)) **W** (R4 Λ M(F3) Λ M^2(F4)) **W** (R4 Λ M(F4) Λ M^2(F3))]
M^3 [R1 Λ M[(F2 Λ M(HR1)) **W** (HR1 Λ M(F2))] Λ M^3(F1)]

Six arrangements with six logical machines are able to manufacture our part. One system choice can be made here for the *Know How to Do* flow, so 1 on 6 (Figure 7.).

Next Knowledges give some orientation manufacturing informations, namely :

- *Bring together manufacturing around the same axe machining*
  → thus, bring in sub-phases and
  determination of number of part-station, and so on for part-fixing
- *Bring together rough and finition of one feature in the same sub-phase as possible*

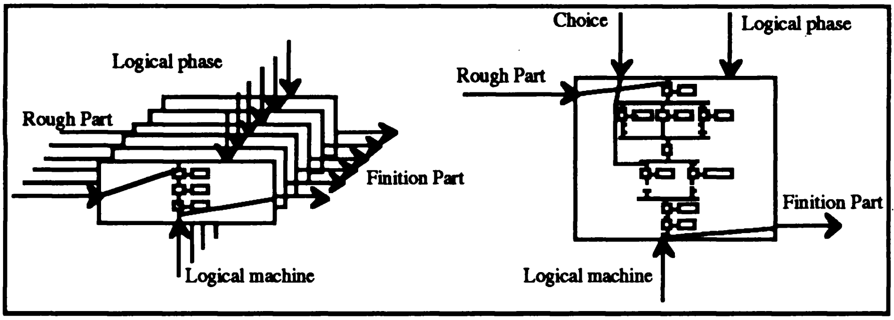

Figure 7. System choices

So, two axes machining are identified by the normal vector of each considered feature :

Axe Machining 1 : A1, C, F1	***Axe Machining 2 :*** F2

Next equation is reduced and now there is only 2 system choices (see around W):

[R4 Λ M(F4)] (1)
Λ
[M^2(F3) Λ M^3[R1 Λ M[(F2 Λ M(HR1)) W (HR1 Λ M(F2))] Λ M^3F1] (2)

(1) : Sub-phase1 with part-station1 (2) : Sub-phase2 with part-station2

Our basic ***Transform*** function is decomposed around two axes machining and installs two sub-phases with their working-post [VOG 87].

The Working-post concept [VOG 87] is composed of ***Loading***, ***Working*** and ***Unloading*** Stations (Figure 8), namely in systemic as *Time* Stations, with **Transports** (*Space* operator) between these.

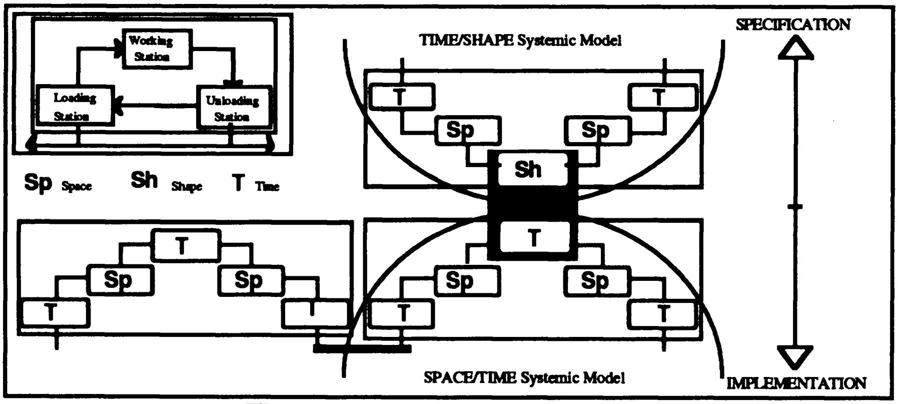

Figure 8. Working-post modelisation

This concept translates a logical station in relation to part-fixing, sub-phase and control-command functions. Our modelisation shows that the ***Shape*** operator is our preoccupation in specification while the ***Time*** operator is in implementation (Figure 8.). Our strategy consists to find the *Shape* operator, while, from the automation point of view, ***Space/Time*** is only controlled.

In this manner, our *Transform* function, supported by Working-post is composed of *Stock* places. By simplification, we only consider the *Stock/Transform* function.

The basic Systemic rule, above mentioned, follows a syntagmatic scheme of functions, namely ***<Shape, Space, Time>***. When there are two *Transform* functions, the rule applied assigns to have *Stock* and *Transport* functions between these. Given that Working-post is, by definition, a *Stock/Transform* function, our modelisation follows ***<Shape, Space, Shape>*** (Figure 9.) rather than *<Shape, Time, Space, Time, Shape>*.

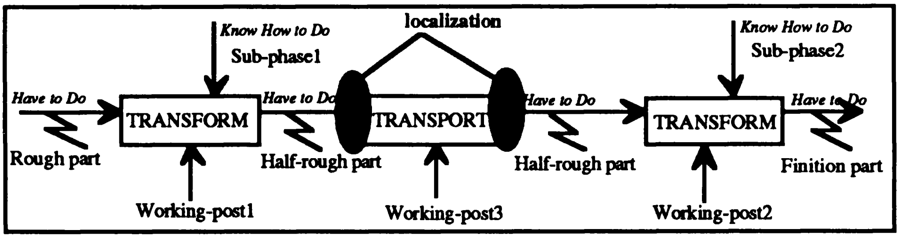

Figure 9. Transport function put forward

Accordingly, we introduce the nature of input and output flows of *Transform* function by the **part localization** concept (Figure 9.), so localization in space to move a part from one referential to another. This information is given, for example, by a robotic skill.

The same approach must be done for ressources used by Discrete Part Manufacturing Systems to give *Be Able to Do* input and output flows definition.

According to workshop definition in terms of tools and machine-tools, process planning of our part is written as a triplet ***<Feature, Part-fixing, Tool>*** on ***Machine***:

Machine I	:	[R4, (P1 W P2), O1] Λ M[F4, (P1 W P2), O3]
Λ		
Machine II	:	M^2(F3, P3, O4) Λ M^3[(R1, P3, O1) Λ M[((F2, P3, O7 Λ M(RH1, P3, O2)) W ((RH1, P3, O2) Λ M(F2, P3, O7))] Λ M^3(F1, P3, O3)

1.4.1.4. Nature - Shape/Space/Time

Since the mechanical finality is to ensure, in terms of cooperation, the system different *Transform* functions, it must be then to put in place their coordination to fullfil the system assignment. This coordination is made by automation skill taking in account different dysfunctions as management of system on going methods.

The **Control Function**, above mentioned (Figure 3.), represents this coordination. It is generating of *Action Decision* flow and receiving of *Action*

Report flow to guarantee sequence order of the different activities of our studied system (Figure 10).

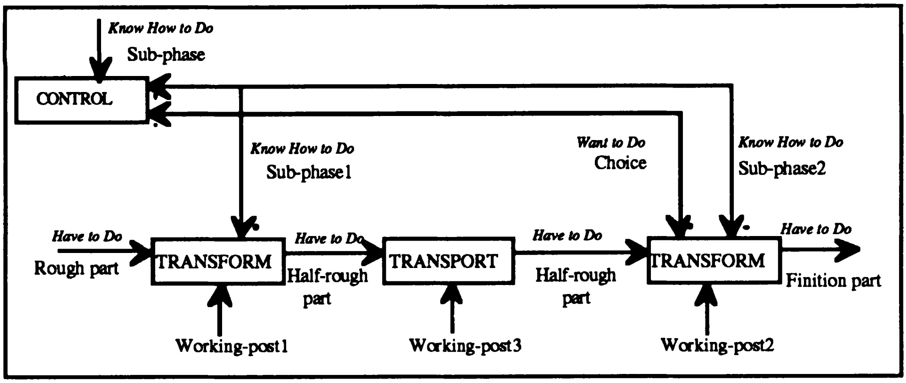

Figure 10. Transform/Nature System choice

1.4.2. Systemic Concurrent Engineering Processing

1.4.2.1. Processing in Referential Model

Usually, in Discrete Part Manufacturing Systems, all things are modelized in this system activity (Figure 11.).In fact, this activity must be processed by different skills, for example on the basic C.A.D./C.A.M. (Computer Aided Design / Computer Aided Manufacturing) ***External-Internal*** axis : Draft definition of the part, Process Planning, Numerical Code for machine-tools, ...

Yet, this processing comes to ***Physical*** structures from ***Functionnal*** definitions.

Furthermore, according to expert skills criterae, the system modelisation goes through a ***Global*** view to a more ***Local*** point of view. Consequently, our system modelisation is decomposed in sub-systems.

Nevertheless, in point of systemic view, all functions and flows are not modelized at this top level can not modelized in any down level. So, each system activity is most important.

For example, to put in place different skills, it is necessary :

1 to process *Have to Do* flow by mechanical corporation,
2 to process *Be Able to Do* flow by mechanical and automation corporations,
3 to couple these different behaviors (Tools, Machine-tools, Part, ...) by automation corporation for defining *Know How to Do* flow,
4 to define events to release the activity considered.

1.4.2.2. Multi-representation across skills to an unique representation for management exchanges

In our D.M.M.S. context and for a skill cooperation, it is necessary to have a common representation for all actors, kind of Esperanto permitting semantic explanations. Usage of Natural Language allows to modelize any scheme of mental pictures around action. [VOG 93] proposes a **semantic actinomy representation** built with an antinomic criterion by alternative combinaison of *Transform*, *Transport* and *Stock* objectivies processed in a synergetic aim (Figure 12.).

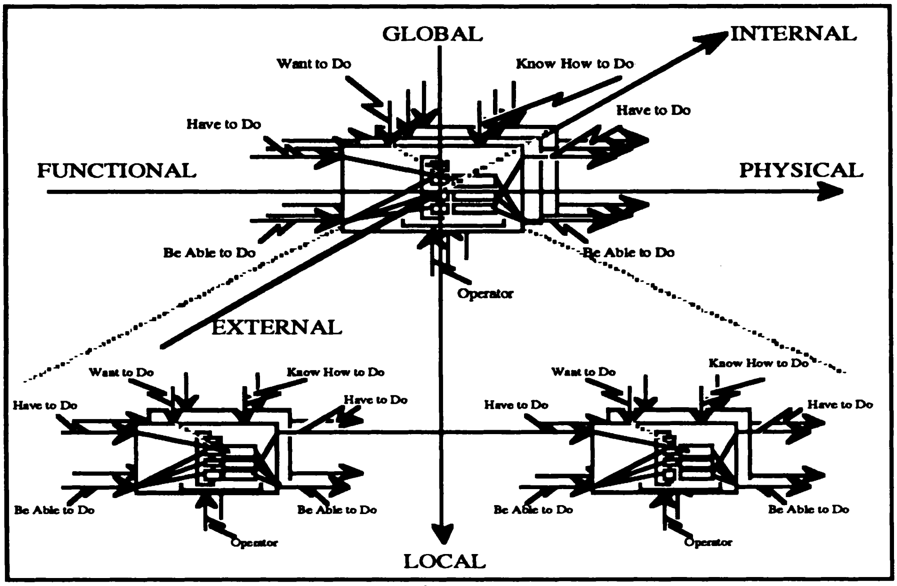

Figure 11. Skill processings

For example, actinomy of a traditional NC-part program is in fact the fusion of individual actems corresponding to each action within a specific manufacturing process. Unfortunately, actinomy of whole manufacturing process is not the sum of these specific sequences. For example, hidden actions from a mechanically perfect behavior corresponds to fault actions from a control-command point of view.

- take sensor in shooting's position	
- switch on light	*- put on tool*
- take picture	*- displace palet to manufacturing's reference*
- switch off light	*- approach tool*
- analyse picture	***- rough HOLE2 out***
- take sensor in fold's position	*- take out tool to release's reference*
- displace palet to change's reference tool	
- put down tool	
- take sensor in fold's position	
- put on tool	*- switch on light*
- displace palet to manufacturing's reference	*- take picture*
- approach tool	*- switch off light*
- execute manufacturing	*- analyse picture*
- take out tool to release's reference	*- take sensor in fold's position*
- displace palet to change's reference tool	
- put down tool	

Figure 12. Example of actinomy fusions : vision control and manufacturing

1.5. CONCLUSION

Our D.M.M.S. concept proposes an architecture for knowledge distribution and exchanges between each skills.

Problems caused by the concurrent management are put in evidence by our systemic approach and are solved in a semantic way.

Actinomy representation for data exchanges is not sufficient. Namely, to capitalize a more structured knowledge on the D.M.M.S. Station (Management exchanges), it is attractive to have a classification with **taxinomy criterea**. In this way, systems can be specialized along *Global-Local* axis with this scheme with "is-type-of" criterion and can be specified with "is-composed-by" criterion from *Functional* to *Physical* point of views.

This extend will permit to have a knowledge systemic library for generic system, and its instanciation will allow to modelize a specific studied system.

1.6. BIBLIOGRAPHY

[COQ 89] COQUET J. CL., Dissertation and its subject. Essay of modal grammar, *Paris Méridien Klincksieck*, 1989 (in French).

[FUS 83] FUSAOKA A., SEKI H., TAKAHASHI K., A description and reasoning of plant controller in temporal logic, Central research laboratory MITSUBISHI (IJCAI), 1983.

[GAL 93] LE GALLOU F., Graphical representation and global modelisation, Proceedings of the 2nd *European Systemic Congress*, Prague (Czecho-Slovak), 4-8/10/93 (in French).

[GAR 92] GARRO O., MARTIN P., MARTI H., Algebraic description for machine tool design - International Journal of Production Research - Vol.30, N° 11, p. 2603-2614, 1992.

[GER 92] GERVAL J.P., LOMBARD M., MOREL G., CIMExpert : more than a training package, Proceedings of *CARS&FOF'92*, p. 341-354 - Metz (France), 17-19/08/1992.

[LOM 93] LOMBARD M., MOREL G., GARRO O., LHOSTE P., Concurrent Design Management and Manufacturing Architecture, *Proceedings of IFA'93*, Vol. IX, p. 317-320 - Sydney (Australia), 18-23 July 1993.

[LOM 94] LOMBARD-GREGORI M., Contribution to discrete part manufacturing engineering : prototyping a concurrent engineering architecture for manufacturing integrated systems, Thesis of the University of Nancy I, 16 February 1994, (in French).

[MOI 90] LE MOIGNE J.L., Modelisation of Complex Systems, AFCET Systèmes, *Dunod*, Paris (France), 1990, (in French).

[MOR 92] MOREL G., Contribution to Computer Integrated Manufacturing and Engineering. Capacitation to Research Training - Univ. of Nancy (France) - (in French)

[MOR 94] MOREL G., LHOSTE P., Prototyping a Concurrent Engineering Architecture, *Proceedings of fifth international Symposium on Robotics and Manufacturing - ISRAM'94* - Hawaï (USA) - 15-17 August 1994.

[PRA 93] PRASAD B., MOREN R.S., RANGAN R., Information Management for Concurrent Engineering : Research Issues. Concurrent Engineering : Research and Applications Vol.1, p. 3-20 - INSB 1063 293X - March 1993.

[PTA 87] RNUR, MICHELIN, ADEPA, PSA, SGN, ISMCM, LACN, LAG, LAMM, Automation Working Station, Journée de Synthèse MRES - Paris (France) - 1987 (in French)

[ROS 75] DE ROSNAY J., The Macroscope, *Collections Points* - Le Seuil, 1975 (in French)

[TIC 91] TICHKIEWITCH S., PRIMECA, a French project for the creation of engineer's formations in mechanical integrated computer design, *Proceedings of the 4th International IFIP TC5 Conference on Computer Applications in Production and*

Engineering : Integration Aspects (10,12/09/1991)- p. 259-266 - Bordeaux (France).

[VOG 93] VOGEL C., The management of large scientific repositories ; Case-study of the knowledge repository on P/L design, ESA. Integrated Logistics & Concurrent Engineering. *ILCE 93* - p. 233-248 - Montpellier (France) - 22-26/03/1993.

[VOG 87] VOGRIG R., BARACOS P., LHOSTE P., MOREL G., SALZEMANN B., Flexible Manufacturing Shop Operation, *Manufacturing Systems* - Vol. 16, p. 43-45, 1987.