

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Application et Critique d'une démarche de Conception d'Interface Homme-Machine

Melebeck, Valérie

Award date:
1991

Awarding institution:
Université de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

FACULTES UNIVERSITAIRES NOTRE - DAME DE LA PAIX

NAMUR

INSTITUT D'INFORMATIQUE.

Application et Critique d'une Démarche
de Conception d'Interface Homme-Machine.

par Valérie MELEBECK

Promoteur : Mr François BODART

Mémoire réalisé en vue de
l'obtention du grade de licenciée
et maître en informatique.

Année académique 1990 - 1991.

Remerciements.

Je remercie tout d'abord Monsieur François Bodart pour m'avoir guidée et conseillée tout au long de cette année.

Je tiens également à remercier particulièrement Monsieur Benoît Sacré qui m'a suivie durant les derniers temps de mon mémoire. Il me fut d'un grand soutien.

Je remercierai enfin Monsieur Christian Beugnier qui m'a aimablement accueillie au sein de son équipe pendant quatre mois. Ce stage restera toujours un bon souvenir et fut une expérience très riche en découvertes.

Table des matières.

Chapitre 1. Introduction.	p. 1
Section 1. Exposé du sujet : but du mémoire, raisonnement suivi.	p. 1
Section 2. Exposé du problème d'application.	p. 2
Section 3. Exposé de la solution retenue.	p. 5
Chapitre 2. Proposition d'interface - Critique.	p. 6
Section 1. Proposition initiale d'interface - Critique.	p. 6
Sous-section 1.1. Exposé de l'interface proposée par la C.G.E.R.	p. 6
Sous-section 1.2. Critique selon les règles ergonomiques et selon le modèle de la tâche.	p. 11
Sous-section 1.3. Critique d'après les défauts de l'application.	p. 15
Sous-section 1.4. Synthèse de la critique sous l'angle des règles de Scapin et Schneiderman.	p. 20
Sous-section 1.5. Enseignements.	p. 23
Section 2. Définition des contraintes.	p. 24
Sous-section 2.1. Contraintes contextuelles.	p. 24
Sous-section 2.2. Contraintes techniques.	p. 24
Sous-section 2.3. Contraintes dûes à la standardisation de la conception des interfaces des agences.	p. 25
Chapitre 3. Application de la démarche de conception d'interface Homme-Machine proposée par B. Sacré et I. Provot.	p. 26
Section 1. Exposé de la démarche et du formalisme employé.	p. 26
Section 2. Conception de l'interface réelle.	p. 34
Sous-section 2.1. Etapes préalables à la conception de l'interface homme-machine.	p. 34
A. La décomposition de la tâche.	p. 34
B. La structuration des traitements.	p. 40
C. La statique des traitements.	p. 48
D. Graphes d'enchaînement des fonctions atomiques composant les fonctions interactives.	p. 61
E. Résumons le formalisme retenu.	p. 81
Sous-section 2.2. Conception de l'interface proprement dite.	p. 84
A. Choix au niveau de la présentation.	p. 84
B. Choix au niveau de la macro-conversation.	p. 84
C. Choix au niveau de la conversation et conception de la maquette.	p. 87
Section 3. Conception d'une interface moderne.	p. 98
Sous-section 3.1. Comparaison des graphes d'enchaînement des fonctions atomiques.	p. 98
Sous-section 3.2. Choix au niveau de la présentation.	p. 109
Sous-section 3.3. Choix au niveau de la macro-conversation.	p. 111
Sous-section 3.4. Choix au niveau de la conversation - Usage de l'arbre de sélection.	p. 114
Section 4. Comparaison des deux interfaces.	p. 132

Chapitre 4. Critique de la démarche.	p. 137
Section 1. Problèmes rencontrés - Causes.	p. 137
Sous-section 1.1. Etapes préalables à la conception de l'interface.	p. 137
Sous-section 1.2. Conception de l'interface proprement dite.	p. 137
Section 2. Avantages.	p. 138
Chapitre 5. Conclusion.	p. 140

Chapitre 1. Introduction

Section 1. Exposé du sujet

Le but de ce travail est l'application et la critique d'une démarche générale de conception d'interface homme-machine conçue par Benoît Sacré et Isabelle Provot.

Cette démarche a été appliquée au problème d'automatisation d'une opération de guichet de la Caisse Générale d'Epargne et de Retraite.

Raisonnement suivi.

Nous sommes partis de l'interface initialement envisagée par la société pour la critiquer à l'aide des règles ergonomiques de Schneiderman (SHN,87) et des étapes du modèle de la tâche mis au point par D. A. Norman et S. W. Draper (NORMAN).

Cette critique aboutit à la nécessité de prendre en compte l'utilisateur et son environnement de travail.

Nous appliquons alors la démarche de conception d'interface homme-machine jusqu'à l'obtention d'une interface appropriée aux contraintes définies par la C.G.E.R., et par l'environnement de l'utilisateur.

Nous concevons ensuite une seconde interface ignorant les contraintes techniques dictées par la C.G.E.R., supposant l'usage des objets interactifs existants et de la souris ainsi que l'existence d'une base de données locale. Cela nous permet de tester plus en avant la démarche proposée.

Nous terminons par une comparaison des deux interfaces. L'idéal aurait été de soumettre une maquette de celles-ci aux utilisateurs. Cette étape n'ayant pu être effectuée, nous nous sommes contentés de les comparer en fonction de critères ergonomiques tels que le retour d'informations, l'adaptation à l'expérience de l'utilisateur, le guidage, ...

Nous concluons par une critique de la démarche.

Section 2. Exposé du problème d'application

Le problème consiste en l'automatisation du remboursement de titres et du paiement de coupons au guichet des agences de la C.G.E.R..

Lorsque le client vient déposer à l'agence une ou plusieurs obligations arrivées à leur dernière échéance et en demande la contrevaletur, il s'agit de l'opération de remboursement de titres. Nous avons un paiement de coupons quand le client dépose des coupons d'actions ou d'obligations arrivés à leur échéance. Un coupon d'obligation, s'il est arrivé à échéance, est normalement payable. Un coupon d'action n'est payé que si un bénéfice est distribué pour l'année en cours par l'émetteur de cette action.

Précisions générales relatives aux titres.

Tout titre renferme un certain nombre d'informations. Toute obligation et tout coupon d'obligation ont une date d'échéance et une date d'émission. Une obligation a un taux de remboursement. A la dernière échéance, ce taux est appliqué à la valeur de l'obligation pour donner la valeur de remboursement de cette obligation. Cette valeur n'est pas soumise à l'impôt.

Une obligation peut aussi bénéficier de primes. Il peut s'agir d'une prime d'émission ou d'une prime à l'échéance. Une prime d'émission signifie une remise faite sur la valeur de l'obligation à l'achat de cette obligation. La prime à l'échéance est un montant alloué au client à l'échéance finale du titre en supplément de la valeur de remboursement. Ces deux primes peuvent être exprimées sous forme de taux ou en un montant. Elles sont soumises à l'imposition.

Les coupons d'action et d'obligation sont également soumis à l'impôt. Le taux d'imposition appliqué aux primes et aux coupons est appelé taux de précompte. Le taux appliqué dépend des dates d'émission et d'échéance du titre concerné.

Le coupon d'une action n'a pas de date d'échéance mais un numéro d'ordre de détachement. Si, par exemple, il y a 12 coupons à l'action, le premier à retirer a le numéro un, le deuxième a le numéro deux, etc

Toute famille de titres, c'est-à-dire un ensemble de titres émis à une date donnée par un certain émetteur et en différentes coupures a le même numéro l'identifiant. Dans notre document, nous appelons ce numéro le Numéro-Valeur. Ce numéro peut être présenté sous différentes codifications. La codification la plus usuelle en Belgique est celle du Secrétariat des Valeurs Mobilières. Cet organisme attribue un Numéro-Valeur à tout titre émis en Belgique (sauf les bons de caisse) et à certains titres étrangers. D'autres organismes étrangers codifient les titres (Cédel, Euroclear, ...). Sur le titre, ce numéro se trouve habituellement apposé tout en haut à droite.

Une dernière information importante est le numéro propre au titre. Ce numéro permet d'identifier tout titre au sein d'une famille de titres ayant le même Numéro-Valeur.

Tout titre peut être soumis à l'opposition ou au tirage.

L'opposition posée sur un titre en interdit le paiement et oblige la saisie de ce titre. Elle est demandée par le propriétaire du titre qui se l'est fait dérobé. L'opposition porte sur les numéros propres à chaque titre au sein d'une famille de titres. Ainsi, il est possible d'avoir deux titres de même Numéro-Valeur avec l'un soumis à l'opposition et l'autre non.

L'émetteur de titres peut procéder à un tirage. Il s'agit du tirage "au sort" de numéros propres à chaque titre. Les titres ayant les numéros tirés sont remboursés anticipativement et à un taux plus intéressant que le taux de remboursement figurant sur le titre. Le tirage porte uniquement sur les obligations.

Précisions relatives au traitement des titres lors d'un remboursement.

1. En agence.

Actuellement, à la réception des valeurs, le guichetier établit trois bordereaux: le premier est destiné à accompagner au siège central les valeurs déposées, le deuxième reste à l'agence et le troisième est remis au client déposant. Si la valeur est belge, le client connu, le calcul possible et le montant à payer pas trop élevé ($\pm$ 50 000 francs belges) alors le client peut éventuellement être payé directement en liquide. S'il y a opposition et que le guichetier n'a pas vérifié, il est le seul responsable de cette erreur. Pour une somme plus élevée, le client doit prévenir l'agence afin qu'elle puisse se munir des fonds. Toute autre valeur sera payée sur compte (compte universel ou compte en devise).

Tout paiement occasionne l'enregistrement via le terminal PHILIPS 6X00 du montant net correspondant dans la comptabilité de l'agence.

2. Au siège central.

Les valeurs belges et étrangères sont traitées séparément. Le calcul éventuellement fait à l'agence est vérifié : la liste des tirages est consultée, le taux de précompte appliqué et l'échéance du titre contrôlés, etc Toute erreur rencontrée est signalée par note à l'agence concernée.

En théorie, l'opposition doit être vérifiée par la C.G.E.R., mais, en pratique, cela n'est pas fait. L'émetteur étant légalement contraint de vérifier les oppositions sur les titres qui lui sont remis, renseignera la C.G.E.R. d'une opposition éventuelle lorsqu'il recevra les titres en retour. Si la C.G.E.R. ne vérifie pas systématiquement s'il y a opposition sur un titre, c'est parce que cette opération requiert un temps relativement important par rapport au nombre d'oppositions décelées.

Après ces diverses vérifications, et si aucune erreur n'est présente, la C.G.E.R. paie éventuellement le client sur compte, s'il n'a pas été payé en liquide, ou s'il s'agit d'une devise autre que le franc belge. Ensuite, les titres sont déposés en compensation à la Banque Nationale. Ces titres sont remis à l'organisme bancaire où est domicilié l'émetteur. L'émetteur doit alors rembourser la C.G.E.R. qui a avancé l'argent au client.

Jusqu'à présent, toute procédure automatisée n'était pas possible étant donné le matériel en agence. En effet, le matériel Philips 6X00 présentait un écran trop petit pour permettre la saisie d'un nombre important d'informations. Le nouveau modèle de terminaux (Philips 9X00) progressivement installés en agence va pallier à ce problème. C'est ainsi que l'automatisation d'opérations telles que le remboursement de titres peut à présent être envisagée.

La solution à apporter à ce problème doit répondre à plusieurs exigences. La première volonté est de pouvoir saisir l'identification d'un maximum de titres en agence. Ainsi, à la réception des valeurs au siège, le service Titres peut vérifier que tous les titres encodés en agence sont bien arrivés.

Une deuxième exigence est la possibilité, à partir de l'identification de la valeur, de calculer le montant à payer au client par le système central. Ce montant calculé serait directement renvoyé au guichetier qui pourrait ensuite payer le client.

Une dernière exigence est d'avoir la possibilité d'encoder le numéro des valeurs et cela afin de pouvoir vérifier les oppositions et le tirage avant de calculer le montant à payer au client. L'encodage de ces numéros serait obligatoire pour toute valeur dont la codification est reconnue.

Section 3. Exposé de la solution retenue

L'opération de remboursement de titres et de paiement de coupons est une application incluse dans le projet d'automatisation de la gestion des comptes de Dépôt à Découvert (plus communément appelés comptes DAD). Ce projet, dénommé Projet "Caisse-Titres" va englober la gestion des valeurs en passage dans la banque, ainsi en est-il des valeurs faisant l'objet d'un remboursement.

La "Caisse-Titres" doit non seulement assurer le suivi du paiement au client jusqu'au remboursement de l'émetteur, mais aussi suivre le cheminement de la valeur en tant que "papier" (matière) depuis l'agence jusqu'en compensation à la Banque Nationale. C'est ainsi que nous voyons apparaître deux types de comptabilité : la comptabilité Espèces qui assurera le suivi du paiement, et la comptabilité Matières le suivi de la valeur elle-même.

La solution retenue va permettre de saisir l'identification du client et de la valeur. Si la codification du numéro-valeur identifiant la valeur n'a pu être reconnue, des informations supplémentaires seront saisies. Ces données seront traitées au siège. Un montant à payer sera éventuellement envoyé, sinon la saisie des numéros est demandée. Ces numéros seront vérifiés. En fonction du résultat, un montant à payer au client sera renvoyé accompagné du résultat de la vérification.

Chapitre 2. Proposition d'interface - Critique

Section 1. Proposition initiale d'interface - Critique

Sous-section 1.1. Exposé de l'interface proposée par la C.G.E.R.

Voici l'écran de saisie unique proposé :

REMB :	VENTE VIF :	DEPOT :	REINV :
NUM VALEUR :		CODE :	DEVISE :
DESCRIPTION :			
DATE EMISSION :		DATE ECHEANCE :	
TAUX :			
COUPON/MANTEAU :		N° COUPON/DATE ECH.COUP :	
NOMBRE DE COUPONS :			
FORME :			
NBRE :	COUPURE :		
NBRE :	COUPURE :		
NBRE :	COUPURE :		
.	.		
.	.		
.	.		
MESSAGE :			
CODE SURVEILLANCE :		CONFIRMATION BLOCAGE :	

COMPTE-LIQUIDITE : 005-_____ -__		REF. ORDRE DE BOURSE :	
COMPTE-LIQUIDITE : 001-_____ -__			
COMPTE-TITRE :			

Cet écran permet la saisie des informations de quatre opérations sur titre. Ces opérations sont le remboursement, la vente vif, le dépôt et le réinvestissement.

Mon but n'est pas de détailler toutes les opérations sur titres. Je vais uniquement exposer la procédure à suivre pour l'opération de remboursement.

Deux procédures sont possibles. Cela dépend de la valeur du Numéro_Valeur. La codification du Numéro-Valeur est reconnue ou non. Le Numéro-Valeur identifie l'émission d'un titre à une date donnée par tel émetteur. Le Numéro-Valeur peut être codifié par différents organismes. La codification la plus courante est celle du Secrétariat des Valeurs Mobilières. D'autres codifications sont Cédel, Euroclear, ... La codification doit être mentionnée dans l'item Code.

Si la codification n'est pas reconnue, le Numéro_Valeur est laissé nul. Les items suivants doivent alors être encodés : Devise, Date-Emission, Date-Echéance-Titre, le Taux, le dernier coupon attaché, la Description et la forme. Cet encodage permettra peut-être, malgré l'absence du Numéro-Valeur, de calculer le montant à payer au client, sinon d'enregistrer un maximum de renseignements à propos de la valeur à rembourser. Le but étant le contrôle de l'arrivée au siège des valeurs reçues en agence.

La Devise est la monnaie dans laquelle doit être payée la valeur.

Les items Nbre et Coupure permettent de saisir la quantité par coupure et la coupure.

Le paramètre COUPON ATTACHE ou DATE ECH du COUPON ATTACHE concerne la mise en dépôt de manteau d'obligation avec coupon(s) et ne concerne donc pas l'opération de remboursement. Le nombre de coupons déposés sera encodé à l'item NOMBRE DE COUPONS.

La FORME mentionne si le titre est nominal, provisoire, au porteur ou nominatif tiers.

On peut prévoir un MESSAGE indiquant, par exemple, que le titre est abîmé, livré sous forme de virement CIK (Caisse Interbancaire Kas), etc...

Le CODE SURVEILLANCE permet de demander un certain traitement du titre en dépôt : par exemple, un père peut déposer ses titres sur son compte-titres et demander que les dividendes soient payés sur le compte de ses enfants. Cet item ne sert donc pas pour l'opération de remboursement.

La CONFIRMATION BLOCAGE indiquerait, par exemple, que le client sait qu'il y a blocage sur les titres qu'il dépose mais qu'il veut néanmoins déposer. Cet item ne sert donc pas non plus pour l'opération de remboursement.

Le COMPTE-LIQUIDITE indique le compte sur lequel le montant doit être versé. Il peut s'agir d'un compte universel en francs belges, d'un compte en devises préalablement ouvert ou du compte de l'agence si le client veut rester anonyme.

Le COMPTE-TITRES et la Référence de l'Ordre de Bourse sont utiles pour l'opération de dépôt.

Suite à cette saisie, en supposant que le Num-Valeur a pu être reconnu, on affichera à l'écran les différents masques de numérotation de titres propres à l'émetteur parmi lesquels le guichetier choisira celui présenté par la valeur à rembourser. Il existe trente formats différents de numérotation de titres. Supposons, par exemple, qu'il a été saisi au premier écran trois coupures de 10.000. On afficherait alors les champs, selon le masque choisi, où le guichetier encodera les numéros figurant sur les titres :

3 COUPURES DE 10.000

DE	----	-	-----	-	---	A	----	-	-----	-	---
DE	----	-	-----	-	---	A	----	-	-----	-	---
DE	----	-	-----	-	---	A	----	-	-----	-	---

Ces encodages seraient envoyés au main frame où les différents traitements seraient exécutés. Ces traitements sont la vérification d'erreurs éventuelles d'encodage, le contrôle des oppositions, le contrôle des tirages et le calcul du montant à payer au client. L'idée serait de renvoyer les numéros des valeurs encodées en les mettant en évidence s'ils sont frappés d'opposition ou de tirage ainsi qu'en les accompagnant du montant à payer par valeur et du montant total à payer au client. Une référence de l'opération, dont le format est imposé, serait également envoyée à l'agence.

A l'agence, trois bordereaux seraient imprimés avec le détail de l'opération, les titres frappés d'opposition saisis, le client payé sur compte ou directement (uniquement pour des valeurs belges) et le paiement imputé dans la comptabilité de l'agence.

Illustrons à l'aide d'un exemple :

Nous devons effectuer une opération de remboursement de titres accompagnés de coupons. Nous constatons que nous avons cinq titres de même Numéro-Valeur, nous pouvons les rassembler dans la même opération. Tout d'abord, nous sélectionnons l'opération adéquate, c'est-à-dire ici REMB en plaçant une croix dans son champs de saisie.

REMB : X	VENTE VIF :	DEPOT :	REINV :
NUM VALEUR : 123456789 99		CODE : 0	DEVISE :
DESCRIPTION :			
DATE EMISSION :		DATE ECHEANCE :	
TAUX :			
COUPON/MANTEAU : MANT		N° COUPON/DATE ECH.COUP : 1-10-91	
NOMBRE DE COUPONS : 10			
FORME :			
NBRE : 2	COUPURE : 5000		
NBRE : 3	COUPURE : 10 000		
NBRE :	COUPURE :		
.	.		
.	.		
.	.		
MESSAGE :			
CODE SURVEILLANCE :		CONFIRMATION BLOCAGE :	

COMPTE-LIQUIDITE : 005-_____ -		REF. ORDRE DE BOURSE :	
COMPTE-LIQUIDITE : 001-0012345-24			
COMPTE-TITRE :			

Nous encodons ensuite le Numéro-Valeur car nous avons reconnu la codification employée pour ce numéro. Supposons qu'il s'agisse de la codification du Secrétariat des Valeurs Mobilières, il sera représenté par un "0". La devise n'est pas encodée : la codification étant reconnue, cette devise pourra être retrouvée par le système via le langage d'interrogation de la base de données.

Les items "DESCRIPTION", "DATE D'EMISSION", "DATE D'ECHEANCE" et "TAUX" ne sont eux aussi encodés que lorsque la codification du Numéro-valeur n'a pas été reconnue ou bien lorsqu'il n'y a pas de Numéro-Valeur figurant sur le titre.

Sous-section 1.2. Critique selon les règles ergonomiques et selon le Modèle de la Tâche

L'idée de n'avoir qu'un seul écran pour toutes les opérations sur titres (sauf l'achat) est venue du fait que ces diverses opérations ont en commun la saisie de la matière. De ce fait, elles ont besoin de saisir des données semblables. D'où l'idée d'un seul écran...

La critique passera en revue les différentes règles ergonomiques empiriques et, pour chacune d'elles, les différentes étapes pertinentes du Modèle de la Tâche.

La tâche à effectuer est définie comme étant l'exécution de l'opération sur titre. Pour réaliser cette tâche, plusieurs sous-objectifs devront être atteints. Ainsi va-t-on considérer comme sous-objectifs la sélection de l'opération et l'introduction d'une caractéristique demandée (fait qui sera itératif puisque plusieurs caractéristiques sont demandées).

1. La Cohérence.

a) Considérons d'abord la sélection de l'opération sur titre.

- Spécification du plan d'action :

Toutes les opérations sur titre possibles sont présentées en première ligne et sont à sélectionner de la même façon. La cohérence à ce niveau semble donc atteinte.

- Niveau Exécution :

La sélection se fait de façon identique pour toutes les opérations. Cependant, rien n'indique comment opérer pour sélectionner : faut-il seulement appuyer sur ENTER lorsque le curseur se trouve derrière le mot à sélectionner ? Ou faut-il mettre une lettre, et laquelle ? etc...

- Niveau Perception :

La seule perception du fait que la sélection a été correctement effectuée serait, par exemple, l'apparition à l'écran de la lettre introduite à côté du mot sélectionné et le positionnement du curseur à côté de la prochaine caractéristique à saisir. Une autre éventualité, dans l'hypothèse où la sélection s'est faite par ENTER, est le simple déplacement du curseur à l'endroit de la prochaine saisie.

- Niveau Interprétation :

Cette perception indiquerait à l'utilisateur que la sélection a été faite et qu'il peut commencer l'introduction des caractéristiques.

b) Introduction d'une caractéristique.

- Spécification de l'action :

Sachant que cet écran sert plusieurs opérations, le problème est de savoir quelles sont les données obligatoires à introduire pour telle opération et quelles sont les données facultatives. Deux possibilités se présentent : tout champs est libre d'être ou non rempli quelle que soit l'opération, ou bien, les champs à ne pas remplir pour l'opération sélectionnée sont bloqués et alors rien ne peut y être introduit.

Dès lors, s'il s'agit de la première possibilité, le guichetier remplira à chaque fois tous les champs de peur de faire une erreur. Sinon, il apprendra que pour telle opération le curseur ne s'arrête pas à telle caractéristique, ou encore qu'il s'y arrête mais qu'on ne peut rien introduire.

Au niveau de la spécification de l'action, l'utilisateur voudra introduire l'information requise où se trouve le curseur.

- Niveau Exécution :

L'utilisateur tentera d'introduire l'information demandée où est situé le curseur.

- Niveau Perception :

S'il s'agit de la première possibilité, l'utilisateur verra s'afficher ce qu'il a introduit, dans la deuxième également, mais seulement si ce champs n'était pas bloqué. Si le champs était bloqué, rien ne s'est affiché.

Le curseur s'est affiché dans le prochain champs à saisir.

- Niveau Interprétation :

L'utilisateur sait qu'il peut continuer la saisie grâce au curseur qui s'est déplacé.

Si la donnée introduite n'a pas été affichée, l'utilisateur apprend qu'il s'agissait là d'une caractéristique facultative pour l'opération choisie.

2. La Concision.

a) Sélection d'une opération.

- Niveau Spécification de l'action :

Le fait d'avoir les champs non obligatoires bloqués serait considéré comme une concision par défaut, mais il manque le fait, au niveau de la spécification de l'action, de prévenir l'utilisateur par, par exemple, le noircissement de la caractéristique bloquée.

- Niveau Exécution :

Considérons le temps d'exécution. Le fait qu'il n'y a aucune indication sur la manière de sélectionner l'opération ne tend certainement pas à minimiser ce temps d'exécution. En effet, le risque d'erreur est important, ce qui peut engendrer plusieurs essais de la part de l'utilisateur.

- Niveau Perception :

On ne peut être plus bref au niveau de la perception : ou bien une lettre (ou un caractère quelconque) apparaît derrière l'opération sélectionnée et le curseur s'est déplacé au champs suivant à introduire, ou bien seul le curseur a bougé (si sélection uniquement par ENTER) ou encore, rien n'a changé, le curseur est resté en première position derrière le champs à sélectionner.

- Niveau Interprétation :

Voyons si ce fait perçu est assez expressif que pour informer l'utilisateur. Si le curseur s'est déplacé, l'utilisateur réalise que la sélection est faite. Cependant, dans le cas où la sélection ne s'opère qu'avec un ENTER, l'utilisateur peut avoir quelque doute sur la sélection si il ne sait pas que, lorsque la sélection est erronée, le curseur ne se déplace pas.

b) Introduction d'une caractéristique.

- Niveau Spécification de l'action :

Certaines caractéristiques sont à introduire complètement, d'autres à partir d'un code. Ainsi en est-il du Code du Numéro Valeur, le zéro indique que le code est à introduire selon une codification chiffrée et non en un mot. Par exemple, nous devons introduire le chiffre "1" et non le mot que ce chiffre représente ("Cedel").

- Niveau Exécution :

Si le champs est "blanc", l'utilisateur introduit ce qui est demandé en entier, sinon le code employé.

- Niveau Perception :

L'utilisateur voit apparaître ce qu'il a introduit et le mouvement du curseur vers la caractéristique suivante à saisir.

- Niveau Interprétation :

Le fait perçu est assez expressif pour que l'utilisateur réalise que l'introduction a bien eu lieu et qu'il doit en effectuer une suivante.

3. Structuration des activités.

a) Sélection d'une opération.

- Structuration des commandes par classes de fréquence d'utilisation et par couche de complexité croissante :

N'ayant pas fait de recherche sur la fréquence des diverses opérations en question, ce qui dépend en très grande partie de l'importance de l'agence ainsi que de son emplacement, je ne pourrais dire ici si elles sont présentées selon leur fréquence d'utilisation. De même, je ne puis dire qu'elles sont présentées selon une complexité croissante car m'étant beaucoup plus concentrée sur l'opération de remboursement de titres et de paiement de coupons, je connais peu les autres opérations.

- Valeurs par défaut :

Nous pouvons remarquer qu'aucune valeur par défaut ne permet d'aider l'utilisateur dans sa sélection, surtout quant à la manière de sélectionner.

- Disposition spatiale des commandes :

On peut apprécier le fait que les différentes opérations sélectionnables de la même façon se présentent toutes sur la première ligne et, ainsi, se distinguent des caractéristiques à introduire.

b) Introduction des caractéristiques :

Pour cette sous-tâche qui consiste à remplir des champs, nous considérerons uniquement l'existence de valeurs par défaut et la disposition spatiale.

- Valeurs par défaut :

Seul le CODE dispose d'une valeur par défaut ("0") qui représente la codification du Secrétariat des Valeurs Mobilières et qui est la plus fréquente.

Les autres caractéristiques auraient difficilement pu disposer de valeurs par défaut étant donné leur trop grande diversité.

- Disposition spatiale :

Cette disposition spatiale a été guidée par l'importance des différentes caractéristiques. En effet, le Numéro Valeur est demandé en premier lieu car, à lui seul, il identifie la valeur. Le Code est mis sur la même ligne que le Numéro Valeur car il le caractérise. Les caractéristiques qui suivent sont demandées seulement si la codification du Numéro Valeur n'a pu être reconnue. Elles sont donc de première importance dans ce cas et apparaissent après la ligne "Numéro Valeur-Code". Les lignes "Nbre-Coupure" doivent être remplies dans les deux cas et apparaissent après les deux possibilités d'identification. Nous pouvons cependant critiquer le fait que les trois champs suivants (Message, Code Surveillance et Confirmation Blocage) apparaissent à cet endroit. Le premier est facultatif et les deux autres obligatoires pour le dépôt (ce qui n'est d'ailleurs pas indiqué).

De même, Réf. Ordre de Bourse ne semble pas à un endroit adéquat. Rien ne justifie qu'il se trouve à côté du numéro de compte.

Nous pouvons tout de même apprécier le regroupement en un seul endroit des différents numéros de compte.

4. Retour d'information.

a) Sélection d'une opération .

- Niveau Perception :

Seuls l'affichage d'une lettre et le mouvement du curseur indiquent que la sélection s'est correctement effectuée. Si la sélection a été incorrecte, le curseur est resté en première position du champs à sélectionner, éventuellement après avoir effacé ce que l'utilisateur avait introduit.

- Niveau Interprétation :

L'utilisateur réalise facilement qu'il a bien ou mal sélectionné l'opération désirée. Cependant, en cas de non-sélection, rien ne lui indique l'origine de cet incident.

b) Introduction d'une caractéristique.

- Niveau Perception :

Il y a affichage de ce qu'il a introduit et le mouvement du curseur au champs suivant. Si ce champs était bloqué, rien ne s'est affiché, le curseur a bougé.

- Niveau Interprétation :

L'utilisateur verra s'il n'a pas fait d'erreur dans ce qu'il a introduit ainsi que la caractéristique suivante à donner.

5. Gestion des erreurs.

a) Sélection de l'opération.

- Niveau Intention :

Peu d'erreurs d'intention peuvent apparaître. L'utilisateur veut effectuer une opération et elle lui est présentée.

- Niveau Spécification de l'action :

Rien n'indique à l'utilisateur comment procéder. Ceci peut lui procurer un sentiment de malaise, par crainte d'effectuer une erreur dont il ignore les conséquences. Afin d'éviter toute erreur, nous pourrions disposer d'une aide préventive qui s'afficherait en bas de l'écran dans une zone prévue à cet effet. Cette aide indiquerait clairement et brièvement à l'utilisateur comment sélectionner correctement.

- Niveau Exécution :

L'utilisateur va "essayer" de sélectionner en imaginant différentes possibilités de procéder. S'il disposait d'une aide, il lui suffirait de suivre les instructions données par le message d'aide.

- Niveau Perception :

L'utilisateur percevra le changement d'état ou non de son écran selon qu'il a "trouvé" ou non la bonne façon de procéder pour sélectionner. Un message d'aide pourrait s'afficher en disant à l'utilisateur s'il a bien ou mal agit. S'il a fait une erreur malgré l'aide en ligne, le message d'erreur devrait lui indiquer précisément son erreur. Cela lui éviterait de la recommencer et lui permettrait de mieux la comprendre.

- Niveau Interprétation :

L'utilisateur s'apercevra qu'il a effectivement une caractéristique à entrer. Ou bien il s'apercevra du non changement de son écran et se trouvera dans la même situation d'incertitude qu'au départ devant le manque d'information quant à l'origine de l'erreur et quant à la façon de sélectionner.

b) Introduction d'une donnée.

- Niveau Intention :

L'utilisateur voudra entrer la caractéristique demandée par la position du curseur. Ici, une aide préventive pourrait renseigner l'utilisateur des cas où l'encodage de la valeur de l'item où se trouve le curseur est nécessaire.

- Niveau Spécification de l'action :

Il saura dans quel format entrer la donnée demandée selon que le champs est "blanc" ou agrémenté d'une valeur par défaut. Une aide en ligne pourrait fournir un exemple d'encodage, cela serait beaucoup plus explicite.

- Niveau Exécution :

Des erreurs peuvent survenir à ce niveau lorsqu'il s'agit d'introduire de longs numéros. Il s'agit principalement d'erreurs de distraction. Des erreurs au niveau du type de la valeur à encoder sont également possibles.

- Niveau Perception :

Seule une erreur dans le numéro de compte peut être détectée lors de la saisie. Cette erreur est renseignée par un message apparaissant en bas d'écran. Les erreurs au niveau du type sont éventuellement détectables.

- Niveau Interprétation :

Le message demande la correction du numéro à l'utilisateur. Les autres erreurs peuvent être détectées par l'utilisateur lui-même (s'il n'est pas distrait !). Un message d'aide pourrait ici être aussi d'un grand secours en indiquant la cause précise de l'erreur (le type de la valeur encodée, la longueur en nombre de caractères, la valeur différente du domaine de valeurs possibles.

6. La Flexibilité.

Outre le fait que cet écran doit exister en langue française et en langue néerlandaise, l'interface est toujours la même : aucune adaptation n'a été pensée pour les débutants en agence, par exemple, pour les non-initiés en matière d'opérations sur titres, etc...

Analyse des Qualités de l'Interface.

=====

Pour terminer cette analyse, nous pouvons considérer à présent l'interface en fonction des diverses qualités que doit présenter une bonne interface.

Passons les en revue.

- 1) **Le Temps d'Apprentissage :**
Celui-ci doit être relativement long du fait du peu de guidage apporté à l'utilisateur et du manque de clarté.
- 2) **Rapidité d'Exécution :**
L'exécution est très lente et cela, parce que l'utilisateur n'est pas du tout guidé.
- 3) **Taux d'erreurs :**
Plus il y a à encoder et plus le taux d'erreur croît. Le risque est élevé étant donné la quantité et la longueur des nombres à introduire.
- 4) **Période de Rémanence :**
Cette période dépendra de la fréquence des opérations sur titres que l'utilisateur aura à effectuer et donc de la localisation et de l'importance de l'agence. Elle sera relativement longue si l'effort cognitif fourni est grand et le temps d'apprentissage long.
- 5) **Satisfaction Subjective :**
La satisfaction subjective doit être faible au vu du peu de compréhension qu'a l'utilisateur du système.

Sous-section 1.3. Critique d'après les défauts de l'application

Le projet Caisse-Titres englobe toutes les opérations sur titres. Les personnes qui ont conçu cette interface avaient à l'esprit le projet global. Elles ont dès lors considéré tous les items résultats d'encodage de ces diverses opérations. Elles se sont davantage concentrées sur la nature des items que le système allait recevoir pour chaque opération que sur le travail d'encodage de l'opérateur.

Un problème issu de l'application est la venue du Numéro-Valeur à identifier. En effet, ce numéro est la clé d'accès à la base de données. Il permet de consulter tous les renseignements concernant la valeur qu'il identifie. Dès lors, ce numéro, concept nécessaire à l'application, n'a à priori rien à voir avec le travail du guichetier. Celui-ci n'a pas besoin de savoir comment fonctionne la base de données, cela sort du modèle de la tâche à exécuter. Or, étant donné l'application, il doit être demandé au guichetier de l'encoder. Dans l'opération manuelle, le guichetier ne devait jamais noter ce numéro. La nouvelle procédure lui demande non seulement de l'encoder mais également de reconnaître la codification utilisée. Il s'agit dès lors d'insérer un nouveau concept dans la tâche de l'utilisateur. Cette codification lui étant complètement inconnue, l'interface doit ici être utilisée pour l'aider à encoder ce concept. Mais l'interface initialement proposée ne présente aucune aide en ce sens.

Un autre problème est de tenter d'éviter les erreurs d'encodage des coupures et de leur quantité respective. Dans ce but, il est demandé d'encoder la valeur nominale totale. Pour cela, l'utilisateur doit trier les valeurs reçues, compter le nombre de valeurs par coupure, multiplier la quantité par coupure par la valeur de la coupure et ensuite additionner les résultats de ces multiplications. Lors de la deuxième saisie, il devra à nouveau manipuler les valeurs pour encoder leur numéro. Cette procédure demande ainsi à l'opérateur de faire face à deux exigences contradictoires. En effet, celui-ci doit veiller à une sécurité maximale par l'encodage des numéros et, en même temps, il doit veiller à ce que l'opération ne prenne pas trop de temps.

Cette solution, on le voit, ne lui facilite pas la tâche, elle prend plus de temps, et risque de créer de longues files au guichet en exigeant deux manipulations au lieu d'une, avec les risques d'erreur que cela peut créer.

Nous voyons ainsi que l'application impose des contraintes au guichetier. Il lui est ajouté un concept étranger à sa tâche auquel il devra s'habituer. Si une bonne aide ne lui est pas fournie, le risque d'erreur est grand.

De plus, la double manipulation des valeurs augmente la durée d'une opération par rapport à l'ancienne procédure. Au lieu de décharger le guichetier, la solution automatisée lui donne davantage de travail. Ces différentes critiques ne sont pas pour faciliter l'acceptation de la nouvelle procédure par les guichetiers : elle ne leur laisse pas la possibilité de mieux servir le client.

Sous-Section 1.4. Synthèse de la critique de l'interface initialement présentée sous l'angle des règles de Scapin et Schneiderman

Avant toute conception, Scapin (SCA, 87) recommande l'étude de pré-requis. Il s'agit d'étudier le poste de travail de l'utilisateur, de définir précisément la tâche et d'étudier les niveaux d'expérience possibles de l'utilisateur.

L'absence de cette étude engendre, lors de la conception, divers choix, dès lors arbitraires, car non-guidés par des critères d'ergonomie. Ces choix arbitraires se feront à différents niveaux de la conception : au niveau de la tâche, lors de la décomposition en sous-tâches et sous-objectifs; au niveau sémantique, par exemple lors du choix du style de dialogue; au niveau syntaxique, pour le choix des objets interactifs abstraits, entre autres; au niveau de l'interaction, lors des choix des opérations physiques élémentaires.

Dès lors, la seule base de conception de l'interface se constitue du résultat de l'analyse conceptuelle et de l'analyse fonctionnelle. Il en résulte une interface guidée par l'application et non par les caractéristiques propres de l'utilisateur et de son environnement de travail, c'est-à-dire une interface dénuée de toute considération ergonomique.

Voyons quel était le but des concepteurs de l'interface initiale.

Leur but était de proposer comme interface un écran polyvalent, valable pour toutes les opérations sur titres excepté le remboursement des titres C.G.E.R.. Etant donné que plusieurs items sont communs à ces diverses opérations, et étant donné que la sélection d'un seul écran est beaucoup plus rapide qu'un enchaînement de plusieurs écrans nécessitant à chaque fois la sélection, l'idée d'un seul écran leur est naturellement venue. L'écran unique avait également pour but de faciliter la tâche de l'utilisateur; il n'avait pas à s'empêtrer dans différents écrans selon l'opération sur titre choisie.

Au départ, l'idée d'un seul écran part de la volonté de gagner du temps mais la confusion générée par la présence constante d'items à choisir ou non selon l'opération va dans le sens contraire.

L'idée d'écran unique aurait été valable si toutes les opérations nécessitaient la même saisie. Dès lors, il est nécessaire de différencier ces opérations et d'offrir pour chacune d'elle une interface qui lui est propre.

Reprenons les conséquences d'une telle conception sous l'angle des règles d'évaluation d'une interface de Schneiderman, appelées aussi Règles d'Or.

1. La règle de la Cohérence doit être vérifiée sous différents points de vue :

La cohérence est à vérifier entre écrans. Or, nous n'avons ici qu'un seul écran. Nous allons donc considérer si cette interface est cohérente dans le sens où deux éléments pouvant être sélectionnés de la même façon sont effectivement sélectionnables de façon

identique.

La cohérence inter-applications n'est pas respectée par l'écran de saisie, unique pour les différentes opérations, étant donné que les mêmes items n'ont pas toujours à être complétés.

La cohérence inter-référentielle est impossible à vérifier étant donné l'absence totale de renseignement quant au feed-back dans la présentation de l'interface.

De même, la cohérence lexicale ne peut être appréciée puisque l'utilisateur n'a pas été consulté.

La cohérence syntaxique semble respectée dans le sens où chaque objet possède un nom distinct, la cohérence spatiale aussi dans le sens où l'objet le plus important (pour le programmeur!), c'est-à-dire le Numéro Valeur, respecte le critère ergonomique recommandant de le placer en haut à gauche de l'écran. Les règles régissant ces cohérences lexicale, syntaxique et spatiale sont les règles de localisation, de structuration et de nomination.

2. La règle de la Concision, étant donnée la tâche, est à considérer au niveau des macro-commandes et des valeurs par défaut. En effet, au niveau des abréviations, l'utilisateur est contraint à introduire l'entièreté des informations demandées, il ne semble donc pas pertinent d'envisager l'abréviation en ce cas. Cependant des abréviations peuvent apparaître dans l'étiquette d'un item apparaissant à l'écran, dès lors doit-on s'assurer de la facilité de compréhension de ces étiquettes par l'utilisateur.

Aucune macro-commande n'a été définie, ce qui aurait par contre été le cas si une étude des niveaux d'expérience de l'utilisateur avait été faite.

Une seule valeur par défaut a été définie, pour l'item Code. Elle représente le code le plus souvent employé (code du Secrétariat des Valeurs Mobilières) pour l'identification d'une valeur. Nous pouvons ici critiquer l'introduction d'un code, jamais employé par l'utilisateur lors de la procédure manuelle, et qui vient donc perturber le modèle de la tâche de l'utilisateur. Cela découle encore une fois de l'absence d'étude de la tâche telle qu'elle est vue par l'utilisateur.

3. Règle de la Structuration des activités.

Seule l'activité de saisie est décrite. Elle se passe toujours en trois étapes : saisie de l'identification du client et de la valeur, saisie des quantités par coupure et des coupures et enfin, saisie des numéros. La saisie est donc présentée comme une tâche fermée.

Par contre, aucune commande n'est décrite, qu'il s'agisse de la sélection d'opération, de l'introduction de données au sein d'un écran, de la terminaison lorsqu'un écran est complété, du passage à l'écran suivant (automatique ou pas?)... Il est dès lors impossible de vérifier si la structuration des commandes est faite par complexité croissante ou par fréquences d'utilisation de l'utilisateur...

4. Le Retour d'informations est également complètement absent, aucune description n'en est faite. Il a fallu imaginer les diverses alternatives possibles de feed-back lors de la critique. Cela découle

également du manque d'étude au niveau ergonomique : si le concepteur avait un temps soit peu pensé à la tâche de l'utilisateur, il aurait pu remarquer l'"oubli" découlant de la non-prise en compte de l'utilisateur.

5. La Gestion des Erreurs est également entièrement passée sous silence. Aucun signalement, de quel type que ce soit (son, clignotement, message d'erreur...) n'est décrit. De plus, aucune aide n'a été prévue, ni aucune commande permettant le "défaire et refaire".
6. La Flexibilité n'a pas été envisagée, cela se déduit directement du manque d'information quant au niveau d'expérience de l'utilisateur.

Outre l'absence d'étude au niveau des pré-requis recommandée par Scapin, les règles empiriques décrites par Scapin et Dumas (DUM, 88) devant être considérées lors de toute conception d'interface, dont la prise en compte, en fonction du poste de travail, de la tâche, et de l'utilisateur, encourage l'apprentissage, la rémanence, le faible taux d'erreurs, et la satisfaction subjective à utiliser l'interface, ont été laissées véritablement de côté.

De cette critique, il ressort très nettement qu'aucune prise en compte de l'ergonomie n'a été faite, et cela même au niveau le plus élémentaire tel le choix de la fonction de sélection.

Dès lors, cette critique nous enseigne qu'il est nécessaire non seulement de définir précisément la tâche, d'étudier les caractéristiques de l'utilisateur, du poste de travail, mais aussi qu'il est nécessaire d'envisager les règles ergonomiques élémentaires pertinentes permettant d'ajuster le plus adéquatement possible l'interface à la définition de la tâche, aux caractéristiques du poste de travail et à l'expérience de l'utilisateur.

Sous-section 1.5. Enseignements

Nous pouvons tirer quelques leçons de l'interface initialement présentée et de sa critique.

Tout d'abord, nous voyons qu'il ne faut pas tenter de définir les informations à saisir pour l'application en concevant l'interface en même temps. Trop d'éléments sont alors négligés ou ignorés. Plusieurs opérations se mêlent, l'utilisateur n'est pas considéré, etc ... Cette façon de concevoir est totalement à proscrire.

La critique permet de mettre en évidence l'importance de la prise en compte de l'utilisateur, de son niveau d'expérience, mais aussi de son environnement de travail.

La critique souligne surtout le caractère primordial de la décomposition de la tâche dans une démarche de conception d'interface. Le modèle de la tâche de D.A. Norman et S.W. Draper permet en effet la mise en évidence de la pensée de l'utilisateur manipulant une interface lors des étapes de spécification de l'action et d'interprétation.

Ainsi, toute conception d'interface, par définition au service de l'utilisateur dans l'exécution de sa tâche, doit se baser sur la définition et la décomposition de cette tâche.

Section 2. Définition des contraintes

Cette section va nous permettre de définir les différentes contraintes à prendre en compte avant de concevoir l'interface. Nous définirons les contraintes contextuelles, les contraintes techniques et enfin les contraintes dûes à la standardisation de la conception d'interface au niveau des agences.

Sous-section 2.1. Contraintes contextuelles

Les contraintes contextuelles concernent le poste de travail et son environnement.

Le poste de travail est caractérisé par un bureau avec guichet où le guichetier dispose d'un terminal relié à une imprimante et au mini-ordinateur de l'agence.

La nouvelle organisation des agences C.G.E.R. va réduire le nombre de guichets. Ceux-ci subsisteront uniquement pour les opérations dites rapides. Pour les autres opérations, le guichetier et le client s'assoieront autour d'un bureau. Il n'y aura plus d'argent au guichet, chaque bureau enverra l'argent via pneumatique au bunker de l'agence. L'objectif de l'installation de ce type d'agence est la volonté d'améliorer le contact avec client. Nous disposons déjà de ce style d'agence à Namur, rue de Bruxelles.

Une grande contrainte ressentie au guichet est le temps de réponse fort lent du système IMS.

L'environnement de travail du guichetier est régi par le temps. En effet, quand une file de personnes s'étend devant vous, il faut pouvoir gérer un certain équilibre entre le service au client et la durée d'une opération par client. L'interface aura ici un important rôle : elle devra aider à l'encodage au mieux pour permettre au guichetier de s'investir davantage dans le conseil et l'information au client.

Sous-section 2.2. Contraintes techniques

La contrainte technique à la conception d'interface est principalement la relative pauvreté en objets interactifs du matériel Philips 9X00 même si celui-ci représente un net progrès par rapport au matériel précédent (usage de huit touches fonction, pas de menu déroulant ...).

Au niveau du matériel, une autre contrainte est le manque de place mémoire du mini-ordinateur permettant l'accès à une base de données locale complète pour les opérations sur titres.

Une autre contrainte technique est due à l'application de remboursement de titres elle-même. L'utilisateur est habitué, manuellement, d'établir un bordereau par client pour différents types de coupons et un bordereau par client et par type de titre. Or, l'application est organisée de telle façon qu'à une opération de remboursement ou de paiement de coupons ne correspondra qu'un seul type de valeur. Ce changement imposé à l'utilisateur devra être pris en compte lors de la conception d'interface afin de faciliter au mieux le passage de l'ancienne procédure à la nouvelle.

Une seconde contrainte due à l'application est l'insertion de la codification dans la nouvelle procédure de l'opération de remboursement. Cette codification du numéro-valeur est complètement inconnue du guichetier, n'ayant jusqu'à présent jamais dû la reconnaître. Cette contrainte devra elle aussi être prise en compte lors de la conception de l'interface.

Sous-section 2.3. Contraintes dues à la standardisation de la conception des interfaces des agences

La liste des règles présentées ici n'est pas exhaustive. Il s'agit des règles principales de standardisation :

Deux champs de saisie maximum par ligne sont acceptés sauf si ces éléments sont interdépendants et que plusieurs lignes identiques se suivent.

Pour les items à domaine de valeurs inférieur ou égal à cinq, les touches fonction doivent être utilisées. Lorsque le nombre de valeurs est supérieur à cinq, la liste doit être utilisée.

Tous les codes doivent être associés à un texte les expliquant.

Les listes ne sont jamais montrées automatiquement à l'utilisateur mais seulement sur demande.

Quand un écran est complété, il doit être confirmé à l'aide de la touche OK.

Chaque champ de saisie doit disposer d'une aide associée.

Il existe une standardisation au niveau des fonctions attribuées à certaines touches fonction réservées.

Chapitre 3. Application de la démarche de conception d'interface Homme-Machine proposée par B. Sacré et I. Provot

Section 1. Exposé de la démarche et du formalisme employé

Disposant des modèles conceptuels résultant de l'analyse fonctionnelle, cette démarche se compose de deux grandes étapes :

- l'affinement des résultats de l'analyse fonctionnelle,
- la conception de l'interface Homme-Machine.

1. L'analyse fonctionnelle détaillée.

1.1 Décomposition de la tâche.

Disposant du modèle E.R.A. (Voir annexe), des contraintes définies par le problème et par la solution choisie, nous commençons par une décomposition de la tâche selon le modèle proposé par D.A. Norman et S.W. Draper (NORMAN). Cette décomposition nous permettra de dégager les fonctions interactives de l'application.

Une fonction interactive correspond à un service de l'application mis à la disposition de l'utilisateur. Toute information reçue par ce type de fonction a été saisie à l'écran. Nous appellerons ce type d'information un message externe de saisie. Toute information générée par une fonction interactive est une information à destination de l'écran. Nous appellerons cette information message externe d'affichage. Ces définitions constituent un affinement de la notion de message externe telle qu'elle est définie dans (BODART-PIGNEUR, 89).

Un message externe y est défini comme un événement extérieur au système mais qui est supposé provoquer une réaction de la part de ce dernier. Les termes message externe de saisie et message externe d'affichage sont bien un affinement de cette définition car ils précisent l'origine ou la destination du message externe. Ainsi, nous saurons qu'un message externe de saisie provient d'un processus de saisie d'informations. De même, nous saurons qu'un message externe d'affichage est un message généré par la fonction interactive considérée et qui est destiné à être affiché.

Pour chaque fonction interactive, nous identifierons et définirons ces deux types de message.

1.2 Structuration des traitements.

A partir des fonctions interactives identifiées à l'étape précédente et de leurs messages externes de saisie et d'affichage, cette étape va permettre de dégager les fonctions composant la fonction interactive. Nous procéderons de manière déductive en partant des messages externes d'affichage et en déduisant au fur et à mesure les fonctions atomiques jusqu'à ce que tous les messages externes de saisie soient reçus par ces fonctions. Des messages internes sont ainsi découverts. Un message interne est un message généré et reçu par une fonction atomique. Cette démarche déductive a été définie dans (BODART-PIGNEUR, 89).

Ensuite, l'application peut être illustrée par un graphe présenté de manière hiérarchique où l'application est décomposée en sous-fonctions reliées entre elles par la relation "Composé De" avec trois repères privilégiés :

- la racine = l'application interactive,
- les feuilles = les fonctions atomiques ou élémentaires,
- le niveau intermédiaire = les fonctions interactives.

1.3 La statique des traitements.

Ce modèle décrit dans (Bodart-Pigneur, 89) a pour objectif de définir pour chaque fonction interactive et pour chaque fonction atomique l'objectif de la fonction, les performances attendues, les messages-données, les consultations de la mémoire du S.I. faites par la fonction, les messages-résultats et les actions faites sur la mémoire du S.I.

Les fonctions atomiques sont présentées dans l'ordre de leur découverte lors de la démarche déductive de la structuration des traitements.

Ceci se fait sans distinguer les messages internes et les messages externes.

1.4 Graphes d'enchaînement des fonctions atomiques des fonctions interactives.

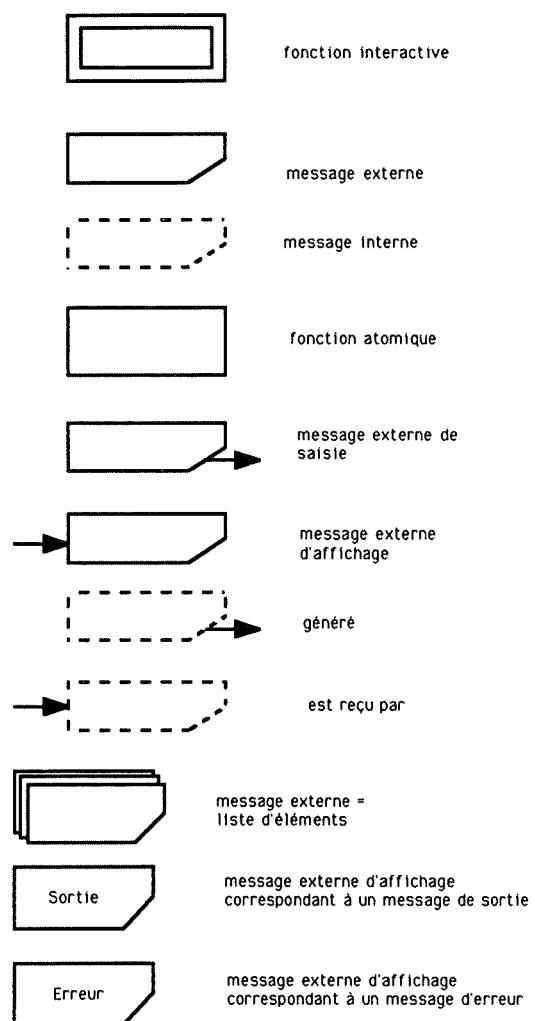
Ces graphes viennent illustrer l'enchaînement des fonctions atomiques tel qu'il a été découvert lors de la structuration des traitements. Ils ont l'avantage de représenter d'une autre façon ce qui est défini dans la structuration des traitements.

Nous reprendrons également les contraintes à vérifier et les règles à appliquer par la fonction.

Nous définirons enfin les messages internes.

Pour chaque fonction interactive, deux graphes sont présentés. Le premier met en évidence les messages externes de saisie et d'affichage en présentant la fonction interactive sous le modèle de la "Boîte noire". Le deuxième ouvre la boîte et montre l'enchaînement des fonctions atomiques.

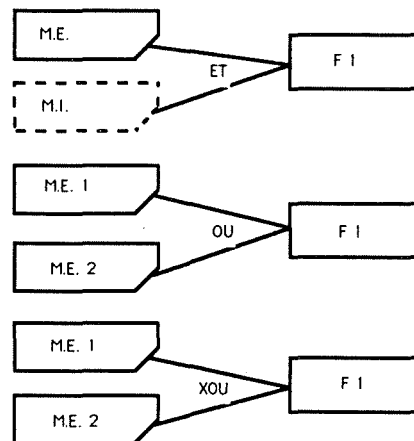
Présentation du formalisme :



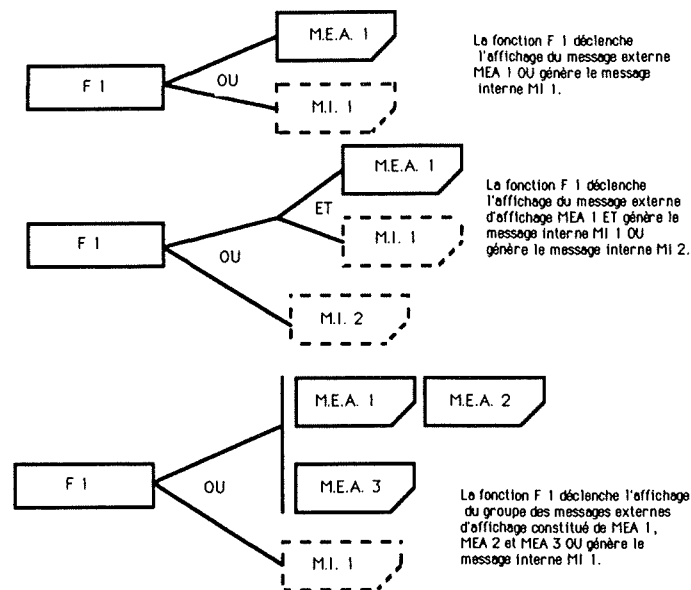
Déclenchement d'une fonction :

Une fonction est déclenchable soit :

- à la réception conjointe (ET) d'au moins deux messages externes ou internes,
- à la réception d'un et d'un seul message parmi plusieurs messages possibles (XOU = OU exclusif),
- à la réception d'un ou plusieurs messages parmi un ensemble de messages possibles (OU).



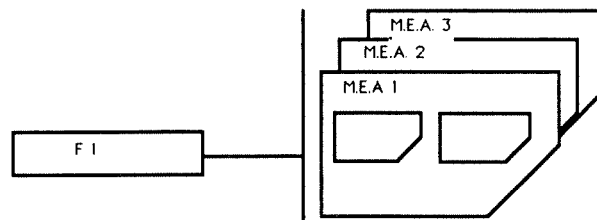
Terminaison d'une fonction :



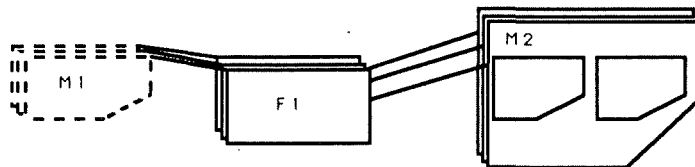
Ces différents formalismes peuvent se combiner de différentes façons et même se complexifier d'après les contraintes et les règles appliquées par la fonction.

Cas particuliers :

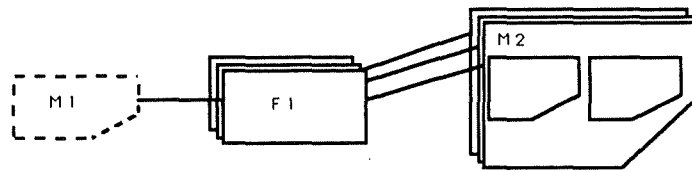
a. Constitution d'une liste d'informations à afficher.



b. Chaque réception d'un message M1 déclenche une occurrence de F1 qui génère à chaque fois un nouveau groupe de message M2.



- c. La réception d'un message M1 provoque le déclenchement de la première occurrence de F1. La terminaison de F1 rend possible le déclenchement d'une autre occurrence de F1, M1 étant reçu par chacune des fonctions F1. Chaque terminaison de F1 génère un message M2.



2. Conception de l'interface Homme-Machine.

A cette étape, nous disposons :

- des fonctions de l'application,
- des messages à saisir et à afficher,
- des conditions de déclenchement des fonctions atomiques, des contraintes techniques et des contraintes liées au poste de travail.

Cette méthode générale de conception d'interface Homme-Machine se décompose en trois étapes:

2.1 Choix du style d'interaction.

Ce choix est guidé :

- par les contraintes techniques liées au gestionnaire d'interaction cible,
- par les contraintes issues de la complexité de l'application (nombre de fonctions interactives, volume d'informations à saisir, à afficher, ...),
- par l'environnement de travail,
- par le profil de l'utilisateur.

2.2 Description du Macro-Dialogue.

Le Macro-Dialogue permet de représenter l'enchaînement des micro-dialogues. C'est à cette étape que sont définis les ordres de succession possibles des micro-dialogues

Un micro-dialogue est défini comme une unité de dialogue autonome ayant un objectif et composée d'un ensemble d'objets interactifs supportant le dialogue avec l'utilisateur final. L'utilisateur communique avec l'application par l'intermédiaire du micro-dialogue. (Exemple : gestion d'un menu, gestion d'un formulaire et de ses composantes.)

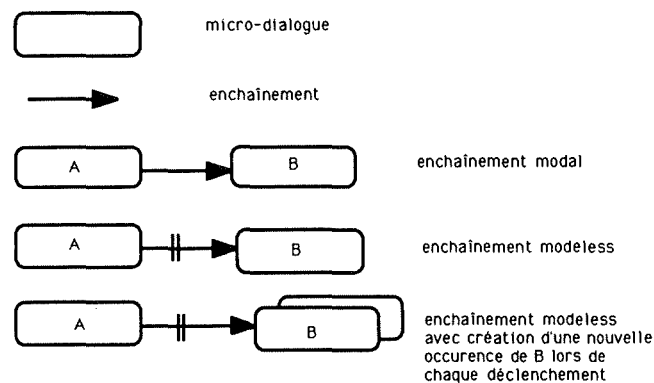
Il existe deux sortes d'enchaînement de micro-dialogues :

- l'enchaînement modal qui oblige l'utilisateur à clôturer le micro-dialogue déclenché et retourne ensuite à l'unité de dialogue "déclencheur".

- l'enchaînement modeless qui permet à l'utilisateur de converser avec une unité de dialogue autre que celle qui vient d'être déclenchée et qui est indépendante de l'unité de dialogue "déclencheur". Ce type d'enchaînement requiert que l'on précise si l'unité de dialogue est unique ou si elle peut exister en plusieurs occurrences.

Le Macro-Dialogue peut être représenté à l'aide d'un graphe d'enchaînement où les sommets sont les micro-dialogues et les arcs les enchaînements possibles.

Formalisme employé :



3. Description des micros-dialogues.

Un micro-dialogue peut être illustré par plusieurs styles d'interaction possibles. Il s'agit ici de choisir le style adéquat en fonction de l'objectif du micro-dialogue.

Quand le style est choisi, il faut faire des choix quant à la représentation des informations à afficher et à saisir, quant à leur regroupement, à leur disposition spatiale, etc...

Pour éviter l'arbitraire au niveau de la représentation des informations, la démarche de conception d'interface Homme-Machine fera appel à l'usage d'un arbre de sélection des objets interactifs abstraits.

Cet arbre permettra, en fonction du type de l'information, de son domaine de valeurs, de son caractère critique ou non, important ou non, ... d'appliquer les règles ergonomiques de structuration, de présentation, de complétude et de cohérence.

L'application de ces diverses règles va permettre de proposer l'objet interactif le plus adéquat pour représenter l'information considérée. (Voir Vanderdonckt, 91).

Conclusion.

Cette démarche est purement empirique. Elle doit être testée avec d'autres styles d'interaction.

Elle va être appliquée au problème de remboursement de titres et de paiement de coupons. Toutes les étapes vont être suivies.

Cependant, étant donné les contraintes techniques déterminées par le matériel Philips 9X00, il serait intéressant de concevoir une interface qui ne tienne pas compte de ces contraintes et qui considère comme étant possible l'emploi de tous les objets interactifs existants dans les interfaces modernes. Il supposerait également la mise à la disposition des agences de la base de données. Ces hypothèses nous permettront ainsi de tester plus en avant cette démarche générale de conception d'interface.

Section 2. Conception de l'interface réelle

Sous-section 2.1. Etapes préalables à la conception de l'interface homme-machine

A. La décomposition de la tâche.

Nous allons décomposer la tâche à accomplir d'après le modèle de la tâche de D.A. Norman et S.W. Draper.

Ce modèle décompose la tâche à accomplir en sept étapes :

1. Etablir l'objectif (état qu'une personne désire atteindre).
2. Définir l'intention (décision d'agir en vue d'atteindre l'objectif).
3. Spécifier la séquence d'actions :
 - Traduire objectif et intention en états désirés du système ;
 - Déterminer les dispositions et mécanismes de contrôle qui produiront cet état ;
 - Déterminer les manipulations requises de ces mécanismes.
- ==> spécification mentale des actions qui doivent être exécutées.
4. Exécuter la séquence d'actions.
5. Percevoir l'état du système.
6. Interpréter l'état du système.
7. Evaluer l'état du système.

Ce modèle est itératif : la tâche peut être exprimée sous la forme d'une hiérarchie d'objectifs et d'intentions.

Certaines étapes peuvent être ignorées et d'autres répétées : le modèle n'est pas séquentiel.

Il permet de mettre en évidence deux types d'écarts entre variables psychologiques et variables physiques :

- écart lié au processus d'exécution,
- écart lié au processus d'évaluation.

Cette décomposition de la tâche nous guidera ensuite pour effectuer la structuration des traitements.

Appliquons ce modèle à la tâche d'exécution de l'opération de remboursement/paiement de coupons. Seules les étapes d'intention et de spécification de l'action sont abordées. En effet, ces étapes sont les seules pertinentes permettant la décomposition de la tâche en sous-tâches et sous-objectifs (voir les notes du cours d'Interface Homme-Machine, deuxième licence).

Cette analyse est faite pour l'interface réelle, en utilisant les objets mis à la disposition de l'outil TFM de définition d'interface pour le matériel PHILIPS 9X00.

OBJECTIF : Exécuter correctement l'opération de paiement/remboursement de coupons.

INTENTION 1 : Choisir l'opération correcte.

Spécification Action 1 :

- a) Obtenir le noircissement du nom de l'opération.
- b) A l'aide des touches fléchées et de F1 = OK.
- c) En se positionnant à l'aide des touches fléchées sur le nom de l'opération et en appuyant alors sur F1.

INTENTION 2 : Encoder correctement les caractéristiques Client.

Spécification Action 2 :

- a) Encoder les items.
- b) A l'aide des touches du clavier en confirmant la fin de l'encodage d'un item par ENTER lorsque l'item n'a pas une longueur maximale à atteindre.
- c) Confirmer la fin de l'encodage des caractéristiques Client par ENTER.

INTENTION 3 : Encoder correctement les caractéristiques de la valeur.

Spécification Action 3 :

- a) Obtenir le curseur sur le champs adéquat et encoder la valeur.
- b) A l'aide des touches clavier et des touches fonctions en confirmant la fin de l'encodage par ENTER.
- c) En confirmant l'encodage total par F1 = OK.

INTENTION 4 : Encoder des caractéristiques supplémentaires concernant la valeur si le Num-Valeur n'est pas reconnu.

Spécification Action 4 :

- a) Encoder les items supplémentaires propres à la valeur.
- b) A l'aide des touches du clavier, des touches fonctions et des touches fléchées.
- c) En confirmant par ENTER les items de longueur non obligatoire et en confirmant l'encodage total par F1 = OK.

INTENTION 5 : Encoder des caractéristiques supplémentaires si la valeur est une obligation.

Spécification Action 5 :

- a) Obtenir le curseur sur le champs d'édition à saisir, encoder l'item et passer à l'item suivant jusqu'à ce que tous les items pertinents aient été encodés.

- b) A l'aide des touches du clavier en confirmant la fin de l'encodage par ENTER si la longueur maximale n'est pas obligatoire.
- c) En confirmant la fin de l'encodage total par F1 = OK.

INTENTION 6 : Introduire les Qtés/Coupure et Coupures.

- Spécification Action 6 :**
- a) Introduire les Qtés/coupon et Coupon.
 - b) A l'aide des touches fléchées pour sélectionner la bonne coupure. L'encodage se fera à partir des touches clavier et sera confirmé par ENTER si la longueur maximale n'est pas obligatoire.
 - c) En confirmant l'encodage total par F1 = OK.

INTENTION 7 : Introduire le format de numérotation.

- Spécification Action 7 :**
- a) Donner le format de numérotation.
 - b) En encodant le format adéquat choisi parmi ceux présentés à l'utilisateur.
 - c) A l'aide du clavier en confirmant par ENTER.

INTENTION 8 : Introduire un numéro.

- Spécification Action 8 :**
- a) Encoder les numéros sous forme de séries De ...
A ...
 - b) A l'aide du clavier. La confirmation de l'encodage d'un numéro est implicite lorsque tout le champs de saisie du numéro est complété.

INTENTION 9 : Indiquer le type de paiement voulu.

- Spécification Action 9 :**
- a) Encoder le Type de paiement.
 - b) A l'aide des touches fonctions.
 - c) En confirmant par ENTER.

INTENTION 10 : Corriger la valeur donnée d'un item.

- Spécification Action 10 :**
- a) Correction de l'item.
 - b) En demandant la possibilité de correction attribuée à une touche fonction intitulée CORR.

- c) En appuyant sur la touche Fonction CORR, en se positionnant sur l'item à corriger, en corrigeant à l'aide du clavier et en confirmant la correction à l'aide de ENTER. Ceci a pour effet de positionner le curseur sur le prochain champs de saisie ou, à défaut, sur la fin du champs de saisie à confirmer ou nulle part si tous les champs de saisie sont encodés et confirmés.

REMARQUE : L'encodage d'un formulaire demande la répétition des étapes de spécification de l'action, d'exécution, de perception et d'interprétation autant de fois qu'il y a d'items à encoder.

De cette décomposition de la tâche, il ressort différents types d'encodages :

- du client
- de la valeur
- de caractéristiques supplémentaires de la valeur
- de caractéristiques supplémentaires si la valeur est une obligation
- des Qtés/Coupure et Coupures
- du format de numérotation
- des numéros sous forme de séries De ... A ...
- du type de paiement.

Nous savons que la solution au problème suppose l'absence du support de la Base de données des valeurs mobilières au niveau de l'agence.

Cette contrainte engendre dès lors un certain ordre au niveau de l'encodage. Par exemple, il n'est pas possible d'encoder les numéros sans avoir encodé au préalable l'identité de la valeur.

Départageons les divers encodages :

- L'encodage Client est indépendant des autres et peut donc être indifféremment saisi au départ de l'opération, au milieu ou à la fin. Cependant, nous préférons le placer dès le début car lorsque les champs à saisir pour le client sont encodés, le guichetier n'a plus qu'à se concentrer sur la saisie de la valeur.

- L'encodage de la valeur et, éventuellement de ses caractéristiques supplémentaires, devra être préalable à l'encodage du format, des coupures et des numéros. Ces différents encodages ne pourront permettre la vérification de l'opposition et du tirage que si la codification de la valeur a pu être reconnue par le guichetier. Dans le cas contraire, ces différents encodages sont inutiles. De plus, le fait que l'encodage de l'identité de la valeur est préalable à tout autre permettra de ne faire apparaître à l'utilisateur que les formats émis par l'émetteur du titre et uniquement les coupures émises pour ce titre. Ainsi, le choix de l'utilisateur est réduit et les erreurs possibles d'encodage limitées.

- L'encodage du type de paiement désiré vient en dernier lieu. En effet, rien ne sert de le demander si, par après, l'application nous apprend que le paiement est impossible. Dès lors, ce n'est qu'après l'affichage du montant à payer au client qu'il est préférable de demander le type de paiement désiré.

Nous voyons donc apparaître trois fonctions interactives composant l'application :

- La première que nous appellerons `Traitement_Saisie` reçoit l'encodage Client, l'encodage Valeur, l'encodage Suppl. Valeur éventuellement, et l'encodage Suppl. Obligation éventuellement.
- La deuxième que nous appellerons `Traitement_Numéros` reçoit le format de numérotation choisi, les Qtés/Coupure et Coupures choisies ainsi que les séries de numéros.
- La troisième que nous appellerons `Enregistrement_Paiement` reçoit le type de paiement demandé.

Définissons à présent le type et le domaine de valeurs des messages externes.

<u>Message</u>	<u>Type</u>	<u>Domaine</u>
Nom_Cli	chaîne de 1 à 20 car.	
Pré_Cli	chaîne de 1 à 12 car.	
Num_Cpte	chaîne de 12 car.	
Num_Id	chaîne de 12 car.	
Type_Val	chaîne de 2 car.	(Coupon, Manteau, Manteau+Coupon)
Num_Valeur	chaîne de 14 car.	
Code	chaîne d'un car.	(0, M, C, E, O, F, V)
Num_Coupon	entier(2)	
Date_Ech_Coupon	chaîne de 8 car.	
Val_Nom_Tot	réel(12.3)	
Nb_Coupon	entier(3)	
Type_Titre	chaîne de un car.	(Action, Obligation)
Devise	chaîne de 3 car.	(BEF, FRF, LUF, DMA,...)
Date_Em	chaîne de 8 car.	
Date_Ech	chaîne de 8 car.	
Taux	entier(4)	
Libellé	chaîne de 1 à 30 car.	
Prime_Em	entier(4)	
Prime_Ech	entier(4)	
Structure	tableau d'entiers (10,2)	
N_Structure	entier(2)	
Coupure	tableau de réels (10,10)	
Coupure_Saisie	entier(10)	
Qté/Coupure	entier(3)	
Série_Numéros	chaîne de 9 car.	
Num_Oppo	tableau de car. (999,9)	
Num_Tira	tableau de car. (999,9)	
Montant_Unit_Tira	chaîne de 10 car.	
Numéro_Ok	tableau de car. (999,9)	
Montant_A_Payer_Mémorisé	entier(10)	
Taux_Précpte_Num	entier(4)	(0,5,10,12,20,25)
Type_Paiement	chaîne de 4 car.	(cash,cpte)
Taux_De_Précompte	entier(4)	(0,5,10,12,20,25)
Montant_Mémorisé	entier(10)	
Référence_Op	chaîne de 14 car.	
Montant_A_Payer	entier(10)	

B. La structuration des traitements

Nous disposons à présent de l'identification des fonctions interactives et de leurs messages externes de saisie et d'affichage.

Pour chaque fonction interactive, nous adopterons une démarche déductive qui nous permettra de découvrir des fonctions atomiques. Cette démarche part de la définition des messages externes d'affichage. Pour chacun de ces messages, nous découvrons une fonction le générant. Ces fonctions nécessitent des messages en entrée. Nous regardons alors si ces messages en entrée correspondent aux messages externes de saisie reçus par la fonction interactive considérée. Si ce n'est pas le cas, il faut à nouveau identifier des fonctions atomiques générant ces messages. Nous dégageons ainsi les messages internes.

Conformément à (BODART-PIGNEUR, 89), chaque traitement doit être défini en langage DSL. Cependant, étant donné que la statique des traitements reprendra la définition de tous les traitements, nous nous contenterons ici de dégager les fonctions atomiques, leurs messages générés et reçus.

L'application porte le nom d'Opération de Remboursement/Paiement Coupons. Elle est divisée, d'après la décomposition de la tâche, en trois fonctions interactives : Traitement_Saisie, Traitement_Numéros et Enregistrement_Paiement.

1. Structuration de Traitement_Saisie.

Cette fonction a pour objectifs :

- enregistrer l'identification du client,
- rechercher la valeur identifiée par le Num_Valeur et renvoyer le ou les formats possibles de numérotation, la ou les coupure(s) possible(s) émise(s) pour cette valeur,
- calculer le montant à payer au client si le Num_Valeur est absent et si l'échéance est arrivée, si le titre est belge et si tous les éléments nécessaires au calcul sont présents.

Les informations reçues par cette fonction sont les suivantes :

Nom_Cli
Pré_Cli
Num_Cpte
Num_Id

ET Num_Valeur ≠ 0
 Type_Val
 Code
 Num_Coupon
 Date_Ech_Coupon
 Val_Nom_Tot
 Nb_Coupon

OU	Num_Valeur = 0	ET	Type_Titre
	Type_Val		Devise
	Code		Date_Em
	Num_Coupon		Date_Ech
	Date_Ech_Coupon		Taux
	Val_Nom_Tot		Libellé
	Nb_Coupon		Prime_Em
			Prime_Ech

Les informations générées par cette fonction interactive sont les suivantes :

	Référence_Op
	ET 1 à N Structure
	ET 1à N Coupure
OU	Référence_Op
	ET Montant_Mémorisé
	ET Taux_De_Précompte
OU	Sortie_1 : "Impossible de payer : éléments manquants.La référence de l'opération est "Référence_Op.
OU	Erreur_1 : "Le Num_Valeur donné est incorrect : "Num_Valeur.

En fonction des objectifs de la fonction, des messages qu'elle reçoit et qu'elle génère, nous allons à présent identifier les fonctions atomiques qui la composent :

Mémo-Montant :	génère Montant-Mémorisé reçoit Montant_Calculé
Calcul_Montant :	génère Taux_De_Précompte et Montant_Calculé reçoit Taux_De_Précompte et Op_Mémorisée
Recherche_Coupures :	génère 1 à N Coupure reçoit Op_Mémorisée
Recherche_Structures :	génère 1 à N Structure reçoit Op_Mémorisée
Choix_Taux_Précpte_Saisie :	génère Taux_De_Précompte reçoit Paiement_Permis et Op_Mémorisée

Mémorisation_Opération : génère Op_Mémorisée
 et Référence_Op
 reçoit Valeur_valide_Num_Val_Nul
 ou (exclusif) Valeur_Valide
 et Client_Mémorisé
 ou (exclusif) Client_Valide

Mémorisation_Client : génère Client_Mémorisé
 reçoit Nouveau_Client_Valide

Validation_Client : génère Nouveau_Client_Valide
 ou (exclusif) Client_Valide
 reçoit Nom_Cli
 Pré_Cli
 Num_Cpte
 Num_Id

Vérification_Paiement_Permis_Saisie :
 génère Paiement_Permis
 ou (exclusif) Sortie_1
 reçoit Valeur_Valide_Num_Val_Nul

Validation_Valeur : génère Valeur_Valide
 ou (exclusif) Valeur_Valide_Num_Val_Nul
 ou (exclusif) Erreur_1
 reçoit Num_Valeur $\neq$ 0
 Type_Val
 Code
 Num_Coupon
 Date_Ech_Coupon
 Val_Nom_Tot
 Nb_Coupon

OU	Num_Valeur = 0	ET	Type_Titre
	Type_Val		Devise
	Code		Date_Em
	Num_Coupon		Date_Ech
	Date_Ech_Coupon		Taux
	Val_Nom_Tot		Libellé
	Nb_Coupon		Prime_Em
			Prime_Ech

2. Structuration de Traitement_Numéros.

Cette fonction a pour objectifs :

de vérifier si les numéros ne sont pas soumis à une opposition
ou au tirage et calculer le montant à payer au client.

Les informations reçues par cette fonction sont les suivantes :

Référence_Op
ET Qté/Coupure
ET Coupure_Saisie
ET 1 à N Série_Numéros
ET N_Structure.

Les informations générées par cette fonction sont les suivantes :

Sortie_6 : "Le numéro de référence est
"Référence_Op.

ET

Sortie_1 : "Les numéros "Numéro_Ok" ne sont
soumis ni au tirage, ni à l'opposition".

OU

Sortie_2 : "Les numéros "Num_Oppo" subissent
l'opposition."

OU

Sortie_3 : "Les numéros "N_Tira" sont sujets au
tirage, le montant unitaire s'élève à
"Montant_Unit_Tira.

ET

Sortie_4 : "Le montant net à payer
="Montant_A_Payer_Mémorisé" et le taux de
précompte appliqué = "Taux_De-Précompte.

OU

Sortie_5 : "Le paiement n'est pas permis :
l'échéance n'est pas atteinte."

OU

Erreur_1 : "La quantité de numéros encodés ne
correspond pas à la valeur nominale totale
encodée à la saisie."

OU

Erreur_2 : "La Référence_Op est absente du
S.I.."

Identifions les fonctions atomiques :

Mémorisation_Montant :

génère Sortie_4
reçoit Montant_A_Payer
et Réf_Op_Valide

Calcul_Après_Vérif :

génère Montant_A_Payer
reçoit Paiement_Permis_Num
et Réf_Op_Valide
et Taux_Précpte
et Num_Tira
ou Num_Ok_Tira
ou Num_Oppo

Vérif_Paiement_Paiement_Permis_Num :

génère Paiement_Permis_Num
ou (exclusif) Sortie_5
et Sortie_1
ou Sortie_2
ou Sortie_3

Vérification_Tira :

génère Num_Tira
ou Num_Ok_Tira
reçoit Num_Ok_Oppo

Vérification_Oppo :

génère Num_Oppo
ou Num_Ok_Oppo
reçoit Saisie_Num_Mémorisés

Mémorisation_Num_Valides :

génère Saisie_Num_Mémorisés
reçoit Réf_Op_Valide
et Num_Saisis_Valides

Validation_Quantités :

génère Num_Saisis_Valides
ou (exclusif) Erreur_1
reçoit Qté/Coupure
Coupure_Saisie
1 à N Série_Numéros
N_Structure.

Choix_Taux_Précpte :

génère Taux_Précpte
reçoit Réf_Op_Valide

Validation_Référence :

génère Réf_Op_Valide
ou (exclusif) Erreur_2
reçoit Référence_Op

3. Structuration d'Enregistrement_Paiement.

Cette fonction interactive a pour objectif :
de valider et enregistrer le paiement.

Cette fonction reçoit en entrée :

Référence_Op
Montant_A_Payer
Type_Paiement.

Cette fonction génère les messages suivants :

Sortie_1 : "Le paiement de l'opération identifiée par
"Référence_Op" est enregistré."

OU (exclusif)	Erreur_1 : "La Référence_Op est inconnue du S.I.."
OU	Erreur_2 : " Le type de paiement = Sur Compte MAIS aucun numéro de compte n'a été donné."
OU	Erreur_3 : " Il n'est pas possible de payer Cash en devises."

Identifions les fonctions atomiques :

Mémorisation_Paiement :

génère Sortie_1
reçoit Paiement_Valide
et Montant_A_Payer
et Mode_Paiement_Accepté

Validation_Paiement :

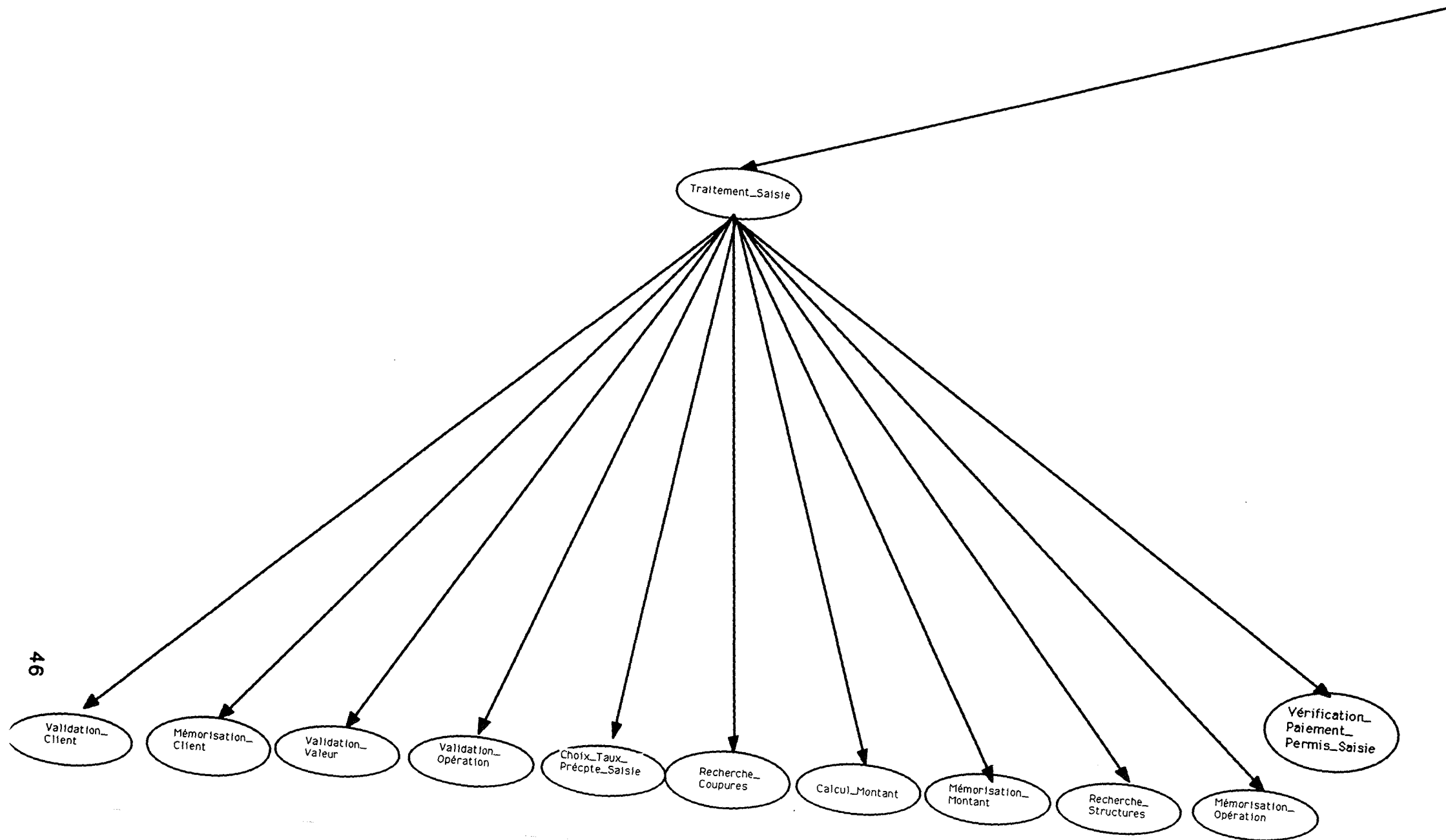
génère Paiement_Valide
ou (exclusif) Erreur_1
reçoit : Référence_Op

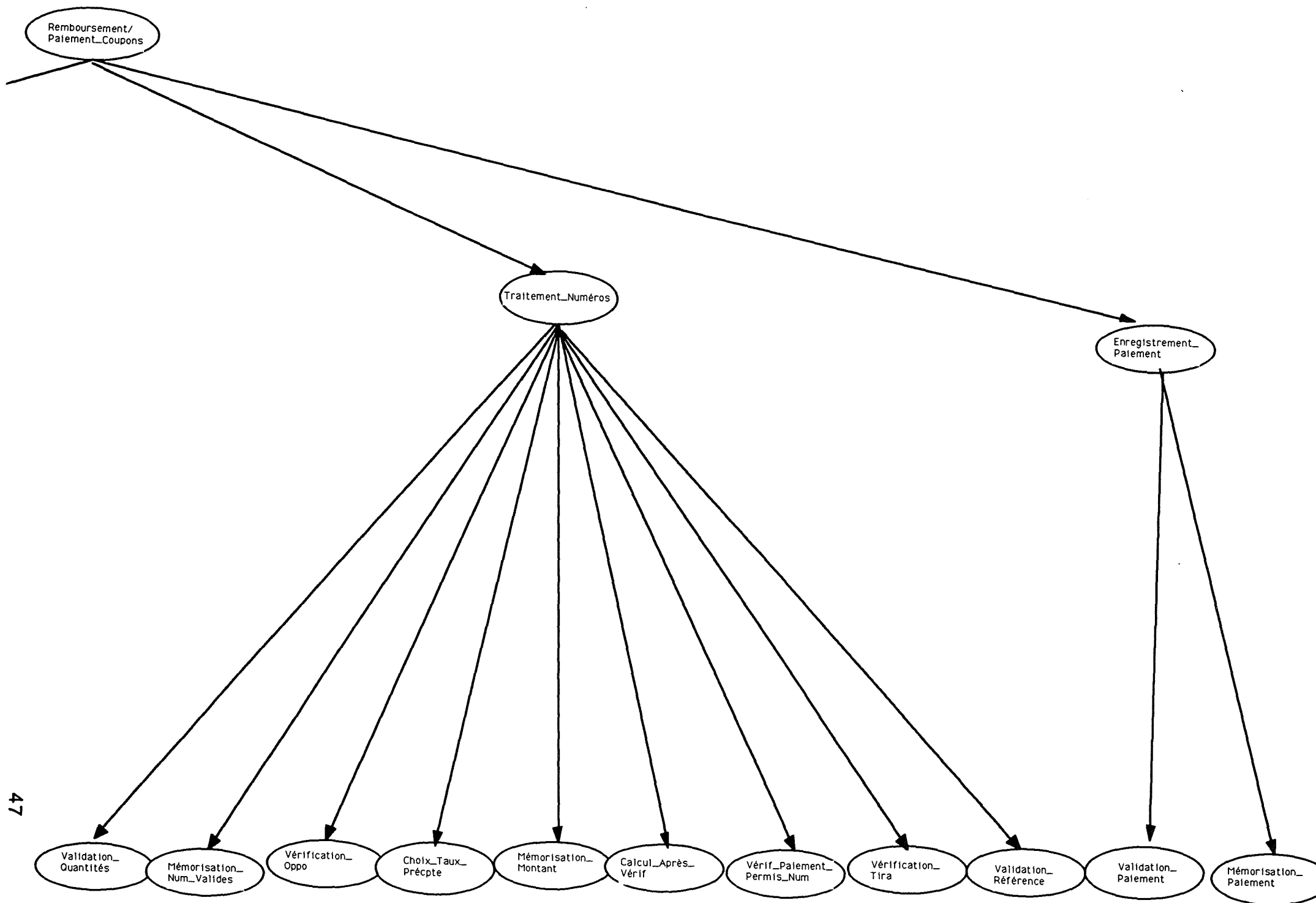
Vérification_Mode_Paiement :

génère Mode_Paiement_Accepté
ou (exclusif) Erreur_2
ou (exclusif) Erreur_3
reçoit Type_Paiement

Suite à cette démarche déductive, nous pouvons illustrer graphiquement la structuration des traitements.

Cette structuration des traitements est illustrée par le graphe de la page suivante. Il est composé des trois niveaux : application, fonction interactive et fonction atomique. La flèche ----> représente la relation Composée-De.





C. La statique des traitements.

Cette étape permet de définir les fonctions atomiques découvertes lors de la structuration des traitements. Nous les présentons dans l'ordre où elles ont été dégagées de manière déductive.

1. Phase Traitement_Saisie :

Objectif : - enregistrer l'identification du client,
- rechercher la valeur identifiée par le Num_Valeur et renvoyer le ou les formats possibles de numérotation, la ou les coupure(s) possible(s) émise(s) pour cette valeur,
- calculer le montant à payer au client si le Num_Valeur est absent et si l'échéance est arrivée, si le titre est belge et si tous les éléments nécessaires au calcul sont présents.

Performances : saisir le plus de valeurs possibles en agence et vérifier le maximum de valeurs.

Messages-Données : Nom_Cli
Pré_Cli
Num_Cpte
Num_Id

ET Num_Valeur ≠ 0
 Type_Val
 Code
 Num_Coupon
 Date_Ech_Coupon
 Val_Nom_Tot
 Nb_Coupon

OU	Num_Valeur = 0	ET	Type_Titre
	Type_Val		Devise
	Code		Date_Em
	Num_Coupon		Date_Ech
	Date_Ech_Coupon		Taux
	Val_Nom_Tot		Libellé
	Nb_Coupon		Prime_Em
			Prime_Ech

Consultation de la mémoire du S.I. : Vamo, Coupure_Emise,
Emetteur, Structure, Coupon,
Code_Ext.

Messages_Résultats : Référence_Op
 ET 1 à N Structure
 ET 1à N Coupure

OU Référence_Op
 ET Montant_Mémorisé
 ET Taux_De_Précompte

OU Sortie_1 : "Impossible de payer : éléments
 manquants.La référence de l'opération est
 "Référence_Op.

OU Erreur_1 : "Le Num_Valeur donné est
 incorrect : "Num_Valeur.

Actions sur la mémoire du S.I. : Ajout d'une Opération
 Ajout d'un Client
 Ajout d'un Traitement
 Ajout d'un Déclenchement.

Fonction Mémorisation_Montant :

Objectif : Mémoriser le montant à payer pour cette opération.

Performances : aucune.

Message-Donnée : Montant_Calculé.

Consultation de la mémoire du S.I. : aucune.

Message-Résultat : Montant_Mémorisé.

Action sur la mémoire du S.I. : Ajout d'un Montant dans Opération.

Fonction Calcul_Montant :

Objectif : Calculer le montant à payer à partir du taux de précompte et
 des éléments de Opération.

Performances : aucune.

Messages-Données : Op_Mémorisée
 ET Taux_De_Précompte.

Consultation de la mémoire du S.I. : aucune.

Messages-Résultats : Montant_Calculé
 ET Taux_De_Précompte.

Action sur la mémoire du S.I. : aucune.

Applique les règles suivantes :

	R1	R2	R3	R4	R5
Type_Valeur = C	-	Y	N	N	-
Type_Valeur = M	-	N	Y	N	-
Type_Valeur = M + C	-	N	N	Y	-
Num_Valeur = 0	N	Y	Y	Y	-
Val_Nom_Tot <> 0	-	N	Y	Y	-
Montant_Coupon <> 0	-	Y	N	Y	-
Valeur = Action	-	N	N	N	Y
Valeur = Obligation	-	Y	Y	Y	N

Si R1 alors Pas-de-Calcul.

Si R2 alors $\text{Montant_Calculé} = (\text{Montant_Coupon} * \text{Nbre_Coupon}) - ((\text{Montant_Coupon} * \text{Nbre_Coupon}) * \text{Taux_De_Précompte})$

Si R3 alors $\text{Montant_Calculé} = (\text{Val_nom_Tot} * \text{Taux}) + (\text{Prime_Ech} * \text{Val_Nom_Tot}) - ((\text{Prime_Ech} * \text{Val_Nom_Tot}) * \text{Taux_De_Précompte})$

Si R4 alors $\text{Montant_Calculé} = \text{R2} + \text{R3}$

Si R5 alors Pas-de-Calcul.

Fonction Recherche_Coupures :

Objectif : Rechercher les coupures émises pour le Num_Valeur donné.

Performances : aucune.

Message-Donnée : Op_Mémorisée.

Consultation de la mémoire du S.I. : Vamo, Coupure_Emise.

Message-Résultat : 1 à N Coupure.

Action sur la mémoire du S.I. : aucune.

Fonction Recherche_Structures :

Objectif : Rechercher le ou les format(s) d'encodage de numéros pour le Num_Valeur donné.

Performances : aucune.

Message-Donnée : Op_Mémorisée.

Consultation de la mémoire du S.I. : Vamo, Emetteur, Structure.

Message-Résultat : 1 à N Structure.

Action sur la mémoire du S.I. : aucune.

Fonction Choix_Taux_Précompte_Saisie :

Objectif : Rechercher le taux de précompte à appliquer au calcul du montant.

Performances : aucune.

Messages-Données : Op_Mémorisée
ET Paiement_Permis.

Consultation de la mémoire du S.I. : aucune.

Message-Résultat : Taux_De_Précompte.

Actions sur la mémoire du S.I. : aucune.

Applique les règles suivantes :

SI	R1	R2	R3	R4	R5	R6	R7	R8	R9
Date_Emission > 01-03-9	Y	N	N	N	N	N	-	-	-
Année de Date_Emission <= 1962	N	Y	Y	N	N	N	-	-	-
Année de Date_Emission > 1962	N	N	N	Y	Y	Y	-	-	-
Année de Date_Echéance < 1984	N	Y	N	Y	N	N	-	Y	N
Année de Date_Echéance >= 1984	N	N	Y	N	Y	Y	-	N	Y
Valeur = Valeur_Exception	N	N	N	N	N	N	Y	N	N
Valeur = Villes 1919-62 ou Eaux 1937 ou Grands Lacs ou CCB 1ère, 2ème, 3ème et 4ème	N	N	N	N	N	N	N	Y	Y
Valeur = Emprunt spécial 81-91 Tranche 1	N	N	N	N	N	Y	N	N	N

ALORS

Taux_De_Précompte = 0 %	-	-	-	-	-	-	X	-	-
8 %	-	-	-	-	-	-	-	X	-
10 %	X	X	-	-	-	-	-	-	-
10,5 %	-	-	-	-	-	-	-	-	X
12,5 %	-	-	X	-	-	-	-	-	-
20 %	-	-	-	X	-	X	-	-	-
25 %	-	-	-	-	X	-	-	-	-

où Valeur_Exception = SNCB Actions Privilégiées
ou Lots 1932 et 1933
ou Constructions et Soutien Ecoles
Catholiques de Liège
ou Centre Médico-Social de la Thudinie à
Lobbès
ou Fobeco
ou Santé par la Mutualité.

Fonction Mémorisation_Opération :

Objectif : Mémorise l'opération à partir de la valeur et du client valides.

Performances : aucune.

Messages-Données : Client_Mémorisé
 OU Client_Valide
 ET Valeur_Valide
 OU Valeur_Valide_Num_Val_Nul

Consultation de la mémoire du S.I. : aucune.

Message_Résultat : Op_Mémorisée.

Actions sur la mémoire du S.I. : Ajout d'une Opération
 Ajout d'un Encodage
 Ajout d'un Crédit
 Ajout d'un Déclenchement
 Ajout d'un Traitement
 Ajout d'un Paiement

Fonction Mémorisation_Client :

Objectif : Mémoriser le nouveau client.

Performances : aucune.

Message_Donnée : Nouveau_Client_Valide

Consultation de la mémoire du S.I. : aucune.

Message-Résultat : Client_Mémorisé

Action sur la mémoire du S.I. : Ajout d'un Client.

Fonction Validation_Client :

Objectif : Vérifier si le client est connu ou non.

Performances : connaître tous les clients qui font des opérations sur
 titres.

Messages_Données : Nom_Cli
ET Pré_Cli
ET Num_Cpte
ET Num_Id.

Consultation de la mémoire du S.I. : Client.

Messages_Résultats : Client_Valide
OU Nouveau_Client_Valide.

Actions sur la mémoire de S.I. : aucune.

Fonction Vérification_Paiement_Permis_Saisie :

Objectif : Vérifie s'il est possible de calculer le montant à payer au client étant donné les éléments saisis.

Performances : pouvoir calculer le plus possible de montants.

Message-Donnée : Valeur_Valide_Num_Val_Nul

Consultation de la mémoire du S.I. : aucune.

Messages-Résultats : Paiement_Permis
OU Sortie_1.

Action sur la mémoire du S.I. : aucune.

Fonction Validation_Valeur :

Objectif : Vérifier que le Num_Valeur donné existe dans Vamo s'il est différent de zéro.

Performances : retrouver un maximum de Num_Valeur dans le Vamo.

Messages_Données : Num_Valeur $\neq$ 0
Type_Val
Code
Num_Coupon
Date_Ech_Coupon
Val_Nom_Tot
Nb_Coupon

OU Num_Valeur = 0 ET Type_Titre
Type_Val Devise
Code Date_Em
Num_Coupon Date_Ech
Date_Ech_Coupon Taux
Val_Nom_Tot Libellé
Nb_Coupon Prime_Em
Prime_Ech

Consultation de la mémoire du S.I. : Code_Ext, Vamo.

Messages_Résultats : Valeur_Valide
OU Valeur_Valide_Num_Val_Nul
OU Erreur_1, Num_Valeur.

Actions sur la mémoire du S.I. : aucune.

2. Phase Traitement_Numéros :

Objectif : Vérifier si les numéros ne sont pas soumis à une opposition ou au tirage et calculer le montant à payer au client.

Performances : Vérifier l'opposition et le tirage pour le maximum de titres.

Messages-Données : Référence_Op
ET Qté/Coupure
ET Coupure_Saisie
ET 1 à N Série_Numéros
ET N_Structure.

Consultation de la mémoire du S.I. : Opération.

Messages-Résultats : Sortie_6 : "Le numéro de référence est
"Référence_Op.

ET Sortie_1 : "Les numéros "Numéro_Ok" ne sont
soumis ni au tirage, ni à l'opposition".

OU Sortie_2 : "Les numéros "Num_Oppo" subissent
l'opposition."

OU Sortie_3 : "Les numéros "N_Tira" sont sujets au
tirage, le montant unitaire s'élève à
"Montant_Unit_Tira.

ET Sortie_4 : "Le montant net à payer
="Montant_A_Payer_Mémorisé" et le taux de
précompte appliqué = "Taux_De-Précompte.

OU Sortie_5 : "Le paiement n'est pas permis :
l'échéance n'est pas atteinte."

OU Erreur_1 : "La quantité de numéros encodés ne
correspond pas à la valeur nominale totale
encodée à la saisie."

OU Erreur_2 : "La Référence_Op est absente du
S.I.."

*Actions sur la mémoire du S.I. : Ajout de 1 à N Coupure_Saisie
Ajout de 1 à N Numéro.*

Fonction Mémorisation_Montant :

Objectif : Mémoriser le montant à payer pour l'opération identifiée par la Référence_Opération.

Performances : aucune.

Messages-Données : Réf_Op_Valide
ET Montant_A_Payer.

Consultation de la mémoire du S.I. : aucune.

Message-Résultat : Sortie_4.

Action sur la mémoire du S.I. : Mise à jour de Montant de Opération.

Fonction Calcul_Après_Vérif :

Objectif : Calculer le montant à payer au client étant donné les numéros soumis au tirage et à l'opposition.

Performances : aucune.

Messages-Données : Réf_Op_Valide

ET Num_Oppo
OU Num_Tira
OU Num_Ok_Tira

ET Taux_Précpte_Num.

Consultation de la mémoire du S.I. : Devise, Cours, Coupon.

Messages-Résultats : Montant_A_Payer.

Action sur la mémoire du S.I. : aucune.

Applique les règles suivantes :

	R1	R2	R3	R4	R5	R6	R7	R8	R9
Num_Tira <> 0	Y	Y	Y	Y	Y	Y	N	N	N
Type_Valeur = C	N	Y	N	Y	N	N	Y	N	N
= M	Y	N	N	N	Y	N	N	Y	N
= M + C	N	N	Y	N	N	Y	N	N	Y
Num_Oppo <> 0	-	-	-	-	-	-	-	-	-
Num_Ok_Tira <> 0	N	N	N	Y	Y	Y	Y	Y	Y

Si R1 alors Montant_A_Payer = (Nbre_Num_Tira * Montant_Unit_Tira)
 = Val_Tot_Tira

Si R2 alors $\text{Montant_A_Payer} = \text{Val_Tot_Tira} +$
 $(\text{Montant_Coupon_Unit de Coupon} -$
 $(\text{Montant_Coupon_Unit de Coupon} * \text{Taux_Précpte_Num}))$
 $* \text{Nbre_Coupon}$
 $= \text{Val_Tot_Tira} + \text{Tot_Coupon_Net}$

Si R3 alors $\text{Montant_A_Payer} = \text{Montant_A_Payer de R2.}$

Si R4 alors $\text{Montant_A_Payer} = \text{Montant_A_Payer de R2.}$

Si R5 alors $\text{Montant_A_Payer} = \text{Val_Tot_Tira} + \text{Tot_Val_Nette_Titre}$

Si R6 alors $\text{Montant_A_Payer} = \text{Val_Tot_Tira} + \text{Tot_Coupon_Net} +$
 $\text{Tot_Val_Nette_Titre}$

Si R7 alors $\text{Montant_A_Payer} = \text{Tot_Coupon_Net}$

Si R8 alors $\text{Montant_A_Payer} = \text{Tot_Val_Nette_Titre}$

Si R9 alors $\text{Montant_A_Payer} = \text{Tot_Coupon_Net} +$
 $\text{Tot_Val_Nette_Titre}$

où $\text{Tot_Val_Nette_Titre} = ((\text{Val_Remb_Unit de Coupon} * \text{Taux_Remb}$
 $\text{de Coupon}) + (\text{Prime_Ech} - (\text{Prime_Ech} * \text{Taux_Précpte_Num}))$
 $+ (\text{Prime_Em} * \text{Taux_Précpte_Num})) * \text{Nbre_Num_Ok}$

Fonction Vérif_Paiement_Permis_Num :

Objectif : Vérifier si le titre est arrivé à échéance.

Performances : aucune.

Message-Donnée : Réf_Op_Valide

ET Num_Oppo
 OU Num_tira
 OU Num_Ok_Tira.

Consultation de la mémoire du S.I. : Opération, Vamo, Coupon.

Messages-Résultats : Sortie_1

 OU Sortie_2
 OU Sortie_3
 ET Sortie_5

 OU(exclusif) Paiement_Permis_Num.

Actions sur la mémoire du S.I. : aucune.

Fonction Vérification_Tira :

Objectif : Vérifier le tirage pour les numéros de l'opération identifiée par Référence_Op.

Performances : Vérifier le tirage pour tous les numéros non soumis à l'opposition.

Message-Donnée : Num_Ok_Oppo

Consultation de la mémoire du S.I. : Vamo, Tira

Messages-Résultats : Num_Tira
OU Num_Ok_Tira

Action sur la mémoire du S.I. : Mise à jour de Tira dans Opération.

Fonction Vérification_Oppo :

Objectif : Vérifier l'opposition pour les numéros de l'opération identifiée par Référence_Op.

Performances : Tous les numéros encodés doivent pouvoir être vérifiés.

Message-Donnée : Saisie_Num_Mémorisés.

Consultation de la mémoire du S.I. : Vamo, Oppo.

Messages-Résultats : Num_Oppo
OU Num_Ok_Oppo.

Action sur la mémoire du S.I. : Mise à jour de Oppo dans Opération.

Fonction Mémorisation_Num_Valides :

Objectif : Mémoriser les coupures, la structure et les numéros dans la mémoire du S.I..

Performances : aucune.

Messages-Données : Réf_Op_Valide
ET Num_Saisis_Valides.

Consultation de la mémoire du S.I. : aucune.

Message-Résultat : Saisie_Num_Mémorisés.

Fonction Validation_Quantités :

Objectif : Vérifier si les quantités encodées correspondent à la valeur nominale totale encodée lors de la phase Traitement_Saisie.

Performances : à vérifier pour toutes les valeurs.

Messages-Données : 1 à N fois Qté/Coupure
ET Coupure_Saisie
ET 1 à N Série_Numéro
ET N_Structure

Consultation de la mémoire du S.I. : Opération.

Messages-Résultats : Num_Saisis_Valides
OU Erreur_1.

Action sur la mémoire du S.I. : aucune.

Fonction Choix_Taux_Précpte :

Objectif : Choisir le taux de précompte en fonction des dates d'émission et d'échéance.

Performances : aucune.

Message-Donnée : Réf_Op_Valide.

Consultation de la mémoire du S.I. : Vamo, Client, Date_Echéance,
Date_Emission.

Message-Résultat : Taux_Précpte_Num.

Action sur la mémoire du S.I. : Mise à jour de Taux_Précpte de
Opération.

Applique les règles vues dans Traitement_Saisie pour choisir le taux de précompte, si la valeur est belge. Si la valeur est étrangère, les règles suivants la grille en annexe intitulée "Retenues fiscales sur Dividendes et Intérêts de Sociétés Etrangères" sont appliquées.

Fonction Validation_Référence :

Objectif : Vérifier la présence de la référence dans le S.I..

Performances : aucune.

Messages-Données : Référence_Op

Consultation de la mémoire du S.I.: Opération.

Messages-Résultats : Réf_Op_Valide
OU Erreur_2.

Actions sur la mémoire du S.I. : aucune.

Actions sur la mémoire du S.I. : Ajout de 1 à N Coupure
Ajout de 1 à N Numéro
Ajout de 1 à N Saisie
Ajout de 1 à N Découpe
Mise à jour de Structure dans
Opération.

3. Phase Enregistrement_Paiement.

Objectif : Enregistrer le paiement.

Performances : aucune.

Messages-Données : Référence_Op
ET Montant_A_Payer
ET Type_Paiement.

Consultation de la mémoire du S.I. : Opération.

Messages-Résultats : Sortie_1 : "Le paiement de l'opération identifiée
par "Référence_Op" est enregistré."

OU (exclusif) Erreur_1 : "La Référence_Op est inconnue du
S.I.."
OU Erreur_2 : " Le type de paiement = Sur Compte
MAIS aucun numéro de compte n'a été donné."
OU Erreur_3 : " Il n'est pas possible de payer Cash
en devises."

Action sur la mémoire du S.I. : Mise à jour de Paiement dans Opération.

Fonction Mémorisation_Paiement :

Objectif : Mémoriser le paiement.

Performances : aucune.

Messages-Données : Montant_A_Payer
ET Paiement_Valide
ET Mode_Paiement_Accepté.

Consultation de la mémoire du S.I. : aucune.

Message_Résultat : Sortie_1.

Action sur la mémoire du S.I. : Mise à jour de Paiement de Opération

Fonction Validation_Paiement :

Objectif : Vérifier la présence de la Référence_Opération dans le S.I..

Performances : A vérifier pour toutes les opérations.

Message-Donnée : Référence_Op.

Consultation de la mémoire du S.I. : Opération.

*Messages-Résultats : Erreur_1
 OU Paiement_Valide.*

Action sur la mémoire du S.I. : aucune.

Fonction Vérification_Mode_Paiement :

*Objectif : Vérifier que le mode de paiement est compatible avec les
 renseignements à notre disposition, et que l'on n'enfreint pas
 le principe de payer en devises uniquement sur compte-
 devise.*

Performances : aucune.

Message-Donnée : Type_Paiement.

Consultation de la mémoire du S.I. : Opération, Num_Cpte.

*Messages-Résultats : Mode_Paiement_Vérifié
 OU (exclusif) Erreur_2
 OU (exclusif) Erreur_3.*

Actions sur la mémoire du S.I. : aucune.

D. Graphes d'enchaînement des fonctions atomiques composant les fonctions interactives.

Trois fonctions interactives découlent de la décomposition de la tâche :

- Traitement_Saisie
- Traitement_Numéros
- Enregistrement_Paiement

Nous les présenteront une à une. Pour chacune, nous donnerons son objectif, les messages externes de saisie, les messages externes d'affichage, les contraintes qu'elle vérifie et les messages internes. Ensuite, le graphe d'enchaînement des fonctions atomiques est présenté, il est éventuellement accompagné d'une proposition de formalisme créée suite à la constatation d'une absence de forme.

Chaque graphe est d'abord présenté comme une boîte noire avec ses messages externes de saisie et d'affichage, ensuite la boîte noire est ouverte, nous obtenons le graphe d'enchaînement des fonctions atomiques.

Pour terminer cette sous-section 2.4, nous présenterons le graphe d'enchaînement des trois fonctions interactives. Nous mettrons ainsi en évidence les actions de l'utilisateur permettant de passer d'une fonction interactive à l'autre. Nous pourrions également connaître ainsi la destination des messages externes d'affichage et l'origine des messages externes de saisie.

Les règles de traitement des fonctions atomiques ont été détaillées lors de la présentation de la statique des traitements.

1. Traitement Saisie.

Objectif :

- Enregistrer l'identification du client.
- Rechercher la valeur identifiée par le Num_Valeur et renvoyer le ou les format(s) possible(s) de numérotation, la (ou les) coupure(s) possible(s) émise(s) pour cette valeur.
- Calculer le montant à payer au client si le Num_Valeur est absent et si l'échéance est arrivée, si le titre est belge et si tous les éléments pertinents au calcul sont présents.

Reçoit :

Nom_Cli	Num_Valeur	Type_Titre
Pré_Cli	Type_Val	Devise
Num_Cpte	Code	Date_Em
Num_Id	Num_Coupon	Date_Ech
Date_Ech_Coupon		Taux
	Val_Nom_Tot	Libellé
		Prime_Em
		Prime_Ech

Affiche :

Référence_Op
1 à N Structure_Utilisée
1 à N Coupure
Montant_Mémorisé
Sortie_1 = " Impossible de payer. La référence attribuée à l'opération
est "Référence_Op.
Erreur_1 = " Le Num_Valeur donné est incorrect : "Num_Valeur.

Vérifie les contraintes :

C1 : Le numéro de compte, s'il est non nul, doit être un compte connu
de la C.G.E.R.
C2 : Si Nom_Cli = Anonyme, alors on a un numéro de compte
obligatoirement.
C3 : Si le Num_Valeur est non nul, alors il doit exister dans le S.I..

Applique les règles :

R1 : Si le Num_Valeur est présent dans le S.I.
Alors Référence_Op
1 à N Structure_Utilisée
et 1 à N Coupure sont affichées.

R2 : Si le Num_Valeur est nul
et si le calcul du montant est possible (càd Type_Titre =
Obligation et Devise = BEF et Date_Em <> 0 et Date_Ech <> 0 et
Date_Ech <= Date_du_Jour et Taux <> 0)
Alors Référence_Op
et Montant_Mémorisé sont affichés.

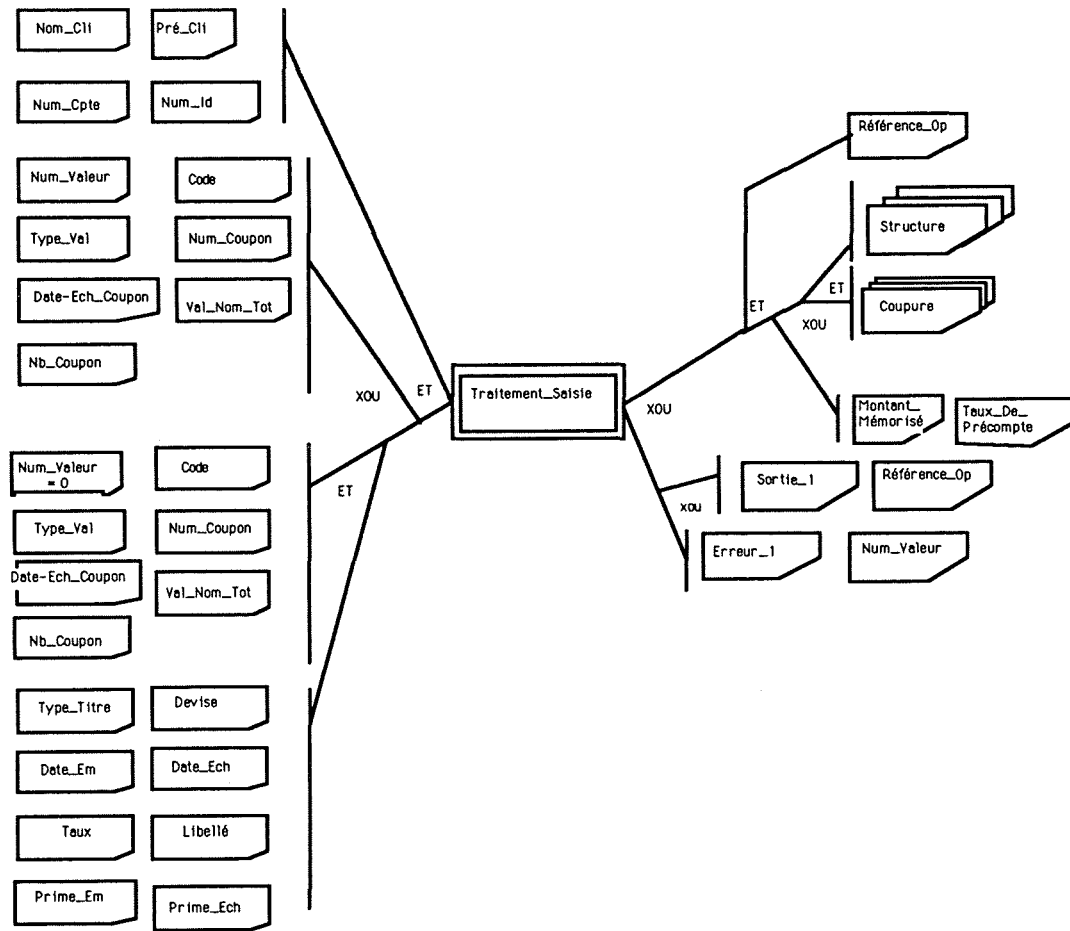
R3 : Si le Num_Valeur n'est pas dans le S.I.
Alors Erreur_1
et Num_Valeur sont affichés.

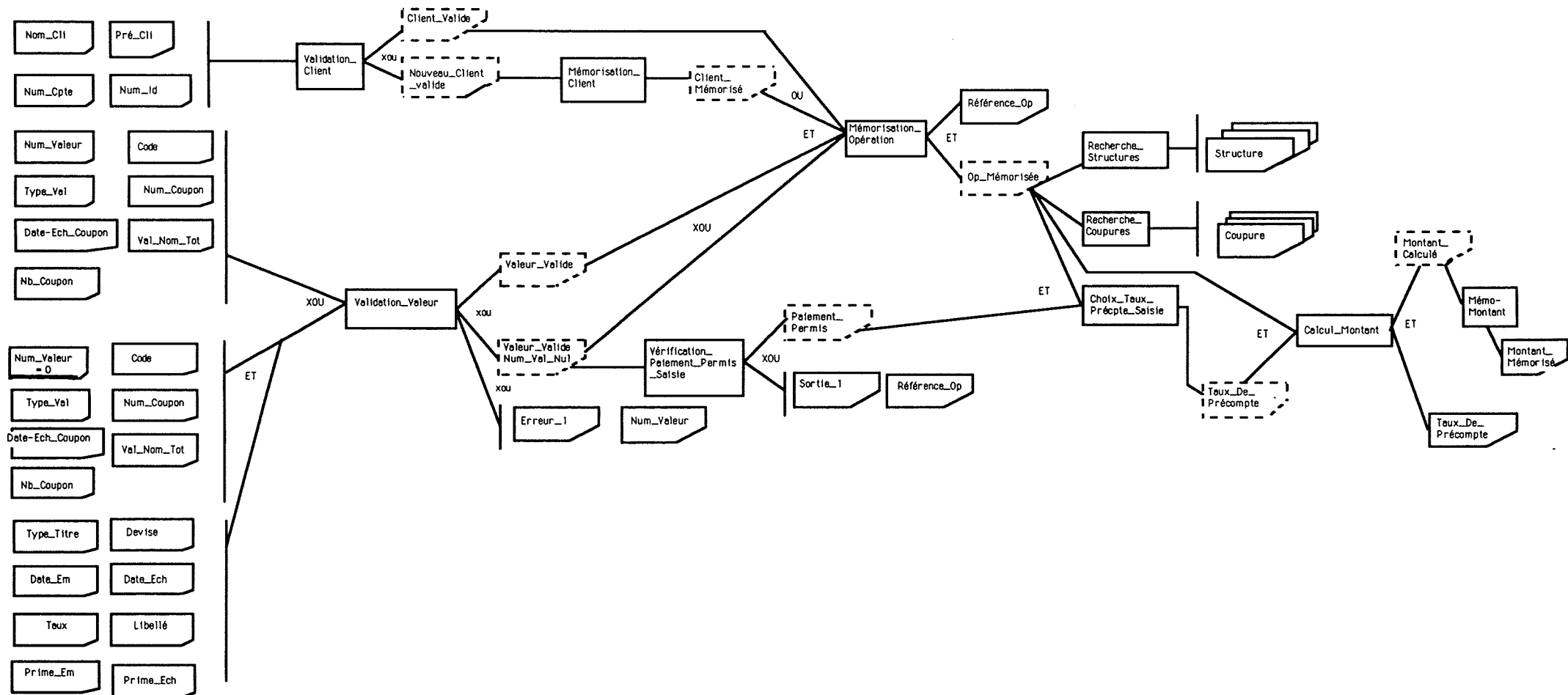
R4 : Si le Num_Valeur est nul
et si le calcul est impossible (càd Type_Titre = Action ou
Date_Em = 0 ou Date_Ech = 0 ou Date_Ech > Date_du_Jour ou
Taux = 0 ou Devise <> BEF)
Alors Sortie_1
et Référence_Op sont affichées.

Graphe d'enchaînement des fonctions atomiques :

Le graphe de la "Boîte noire" est présenté ci-après. Il est suivi du
graphe d'enchaînement des fonctions atomiques.

Traitement-Saisie reçoit toujours le groupe d'identification du client. Il
reçoit ensuite le groupe "Num-Valeur, Type-Valeur, etc ..." quand le
Num-Valeur est différent de zéro. Il reçoit ce groupe plus un autre
groupe d'informations supplémentaires si le Num-Valeur est égal à
zéro.





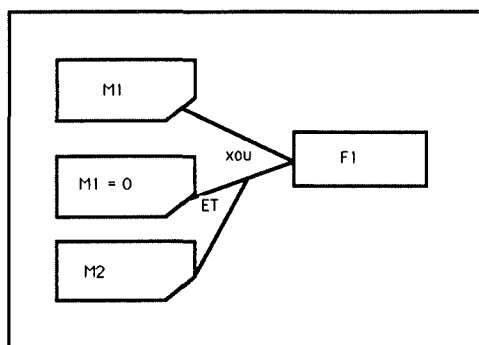
Propositions au niveau du formalisme :

A. La fonction atomique Validation_Valeur reçoit un groupe de messages externes concernant le Client accompagné d'un groupe de messages externes (Num_Valeur, Type_Valeur, Code, Num_Coupon, Date_Ech_Coupon, Val_nom_Tot, Nb_Coupon) lorsque la valeur de l'item Num_Valeur est différente de zéro.

Lorsque la valeur du message Num_Valeur est égale à zéro, la fonction atomique Validation_Valeur reçoit, en plus de ces deux groupes de messages externes, un autre groupe de messages externes (Type_Titre, Devise, Date_Em, Date_Ech, Taux, Libellé, Prime_Em, Prime_Ech).

Pour exprimer cette contrainte graphiquement, je propose de reproduire le groupe de messages contenant Num_Valeur en indiquant la valeur du Num_Valeur.

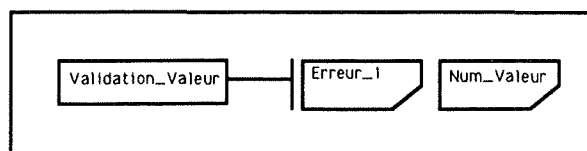
Nous aurons donc cette nouvelle forme :



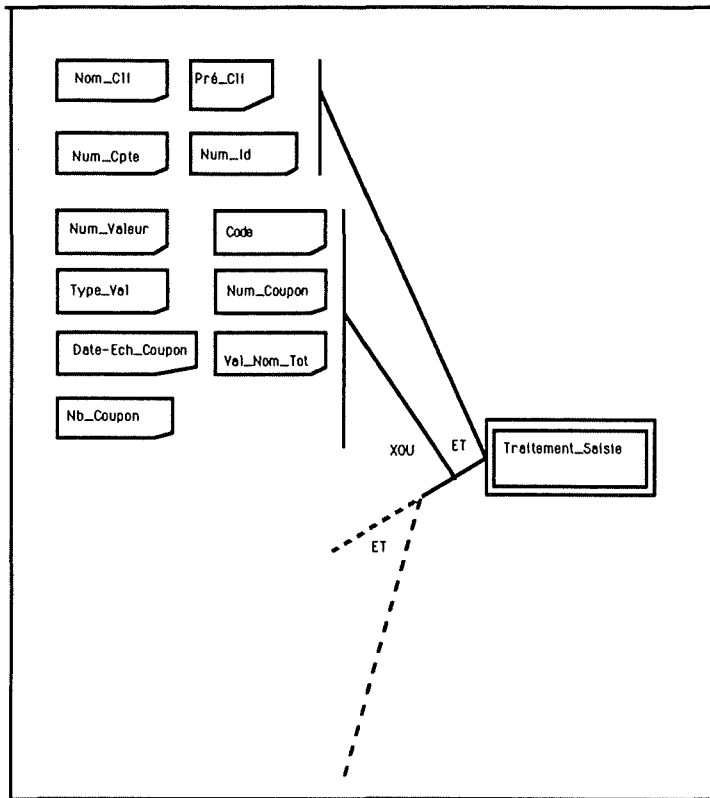
B. La fonction atomique Validation_Valeur affiche le groupe de messages externes Erreur_1 et Num_Valeur. Or, Num_Valeur est contenu dans la définition de Erreur_1 :

Erreur_1 : "Le Numéro_Valeur donné est incorrect : "Num_Valeur.

Le message Erreur_1 étant un message statique, je propose de le différencier du message Num_Valeur qui est, lui, dynamique en les dissociant. La présence d'un élément dynamique est ainsi représentée graphiquement. Ce qui donne :



C. Lorsque une fonction reçoit des messages externes qui seront groupés à chaque réception, c'est-à-dire qui ne seront jamais reçus de façon individuelle, nous emploierons la barre verticale exprimant le regroupement. Il s'agit là de la réciproque du formalisme exprimant la génération de groupe qui emploie une barre verticale pour grouper les messages externes d'affichage.



2. Traitement Numéros.

Objectif :

Vérifier les oppositions et le tirage sur les séries de numéros reçues et calculer le montant à payer au client.

Reçoit :

Référence_Op
1 à N Qté/Coupure
1 à N Coupure_Saisie
1 à N séries de ... à ... Série_Numéros
N_Structure

Affiche :

Sortie_1 = " Les numéros "Numéro_Ok" ne sont soumis ni au tirage ni à l'opposition."
Sortie_2 = " Les numéros "Num_Oppo" subissent une opposition."
Sortie_3 = " Les numéros "N-Tira" sont sujets au tirage, le montant unitaire s'élève à "Montant_Unit_Tira."
Sortie_4 = " Lemontant net à payer = "Montant_A_Payer_Mémorisé" et le taux de précompte appliqué = "Taux_Précompte_Num."
Sortie_5 = " Le paiement n'est pas permis : l'échéance n'est pas atteinte ("Date_Ech").
Sortie_6 = " La Référence de l'opération est "Référence_Op."
Erreur_1 = " La quantité de numéros encodés ne correspond pas à la Valeur Nominale Totale encodée à la saisie."
Erreur_2 = " La Référence_Opération "Référence_Op" n'existe pas dans le S.I.."

Vérifie les contraintes :

C1 : La Référence_Op doit exister dans le S.I..
C2 : La somme des produits (Coupure_Saisie * Qté/Coupure) doit être égale à la Valeur Nominale Totale saisie précédemment pour Traitement_Saisie.
C3 : Le client ne peut être payé que si l'échéance est atteinte.

Applique les règles :

R1 : Si tous les numéros sont "OK"
Alors Sortie_1
et Sortie_4 sont affichées.

R2 : Si tous les numéros subissent une opposition,
Alors Sortie_2 est affichée.

R3 : Si tous les numéros subissent une opposition ou le tirage,
Alors Sortie_2
Sortie_3 et Sortie_4 sont affichées.

R4 : Si tous les numéros sont soumis au tirage,
Alors Sortie_3
et Sortie_4 sont affichées.

R5 : Si certains numéros subissent le tirage,

Alors Sortie_1

Sortie_3 et Sortie_4 sont affichées.

R6 : Si certains numéros subissent une opposition,

Alors Sortie_1

Sortie_2 et Sortie_4 sont affichées.

R7 : Si des numéros sont soumis à l'opposition, d'autres au tirage et les autres à rien,

Alors Sortie_1

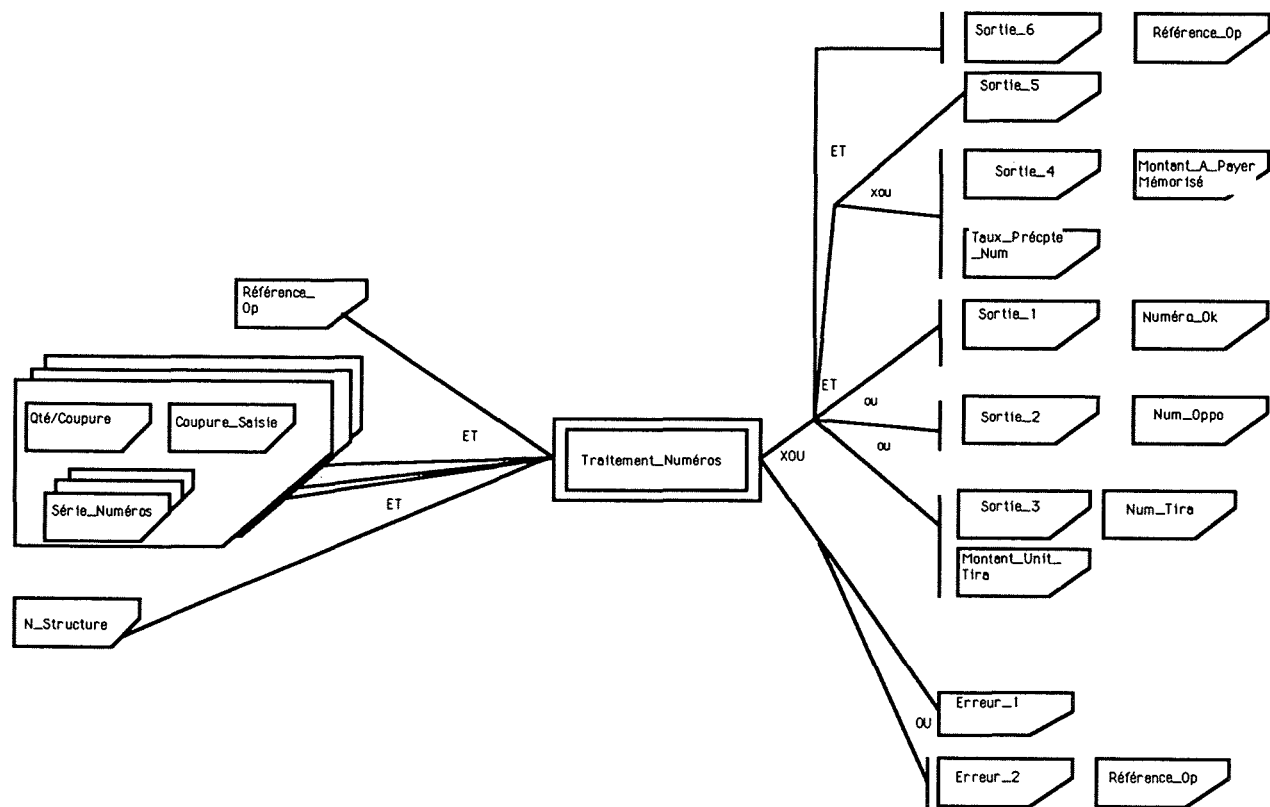
Sortie_2

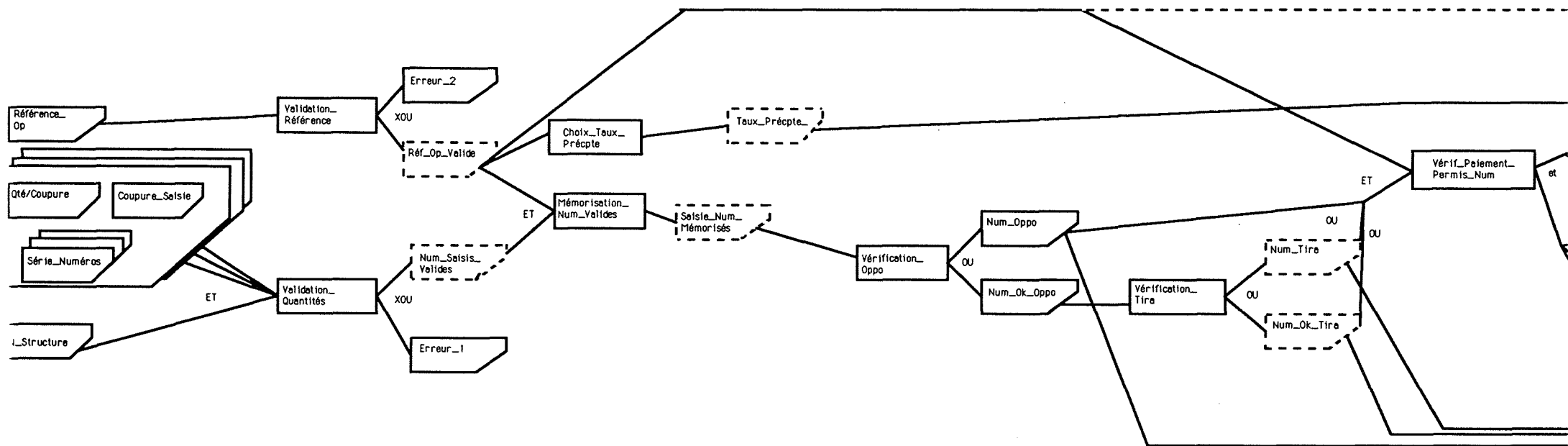
Sortie_3

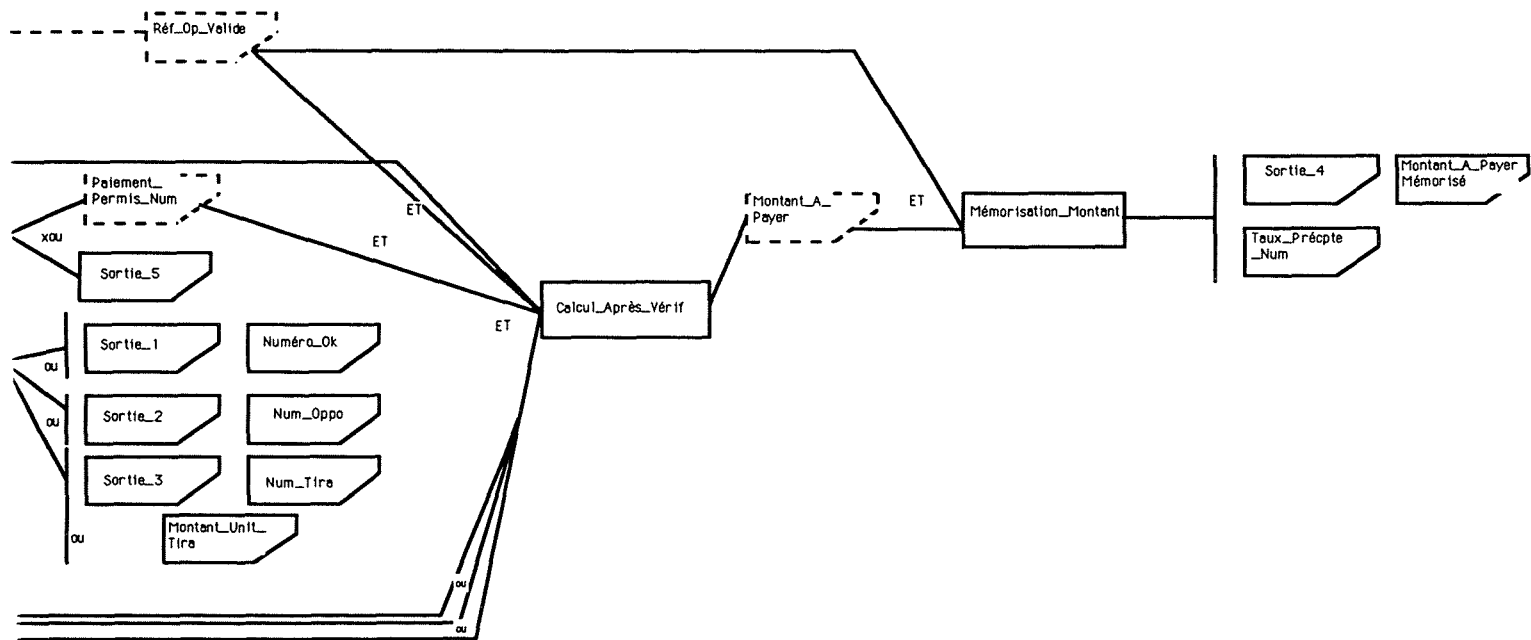
Sortie_4 sont affichées.

Graphe d'enchaînement des fonctions atomiques :

Les trois pages suivantes présentent le graphe sous le modèle de la "boîte noire" et le graphe d'enchaînement des fonctions atomiques.



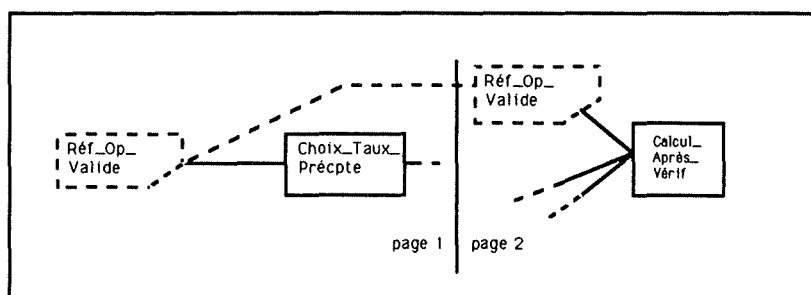




Proposition au niveau du formalisme :

Lorsqu'un message interne est reçu par une fonction atomique distante, c'est-à-dire sur une autre page, il sera dupliqué sur cette autre page. Ainsi, le lecteur ne devra pas retourner en arrière pour connaître le message en entrée de la fonction atomique considérée.

Pour différencier le lien du message généré à son double des autres liens, nous le mettrons en pointillés. Ce trait en pointillés avertira le lecteur de la réception de ce message par un fonction atomique non encore visible parce que située sur une page ultérieure. C'est le cas du message interne Réf_Op_Valide du Traitement Numéros :



3. Enregistrement Paiement.

Objectif :

Enregistrer la confirmation de paiement de l'opération identifiée par Référence_Op et le Type de paiement demandé.

Reçoit :

Référence_Op
Montant_A_Payer
Type_Paiement.

Affiche :

Sortie_1 = "Le paiement de l'opération numérotée "Référence_Op" est enregistrée."

Erreur_1 = "La référence de l'opération est inconnu du S.I.."

Erreur_2 = "Le Type de paiement - Sur Compte MAIS aucun N° de compte n'a été encodé."

Erreur_3 = "Il n'est pas possible de payer CASH en devises."

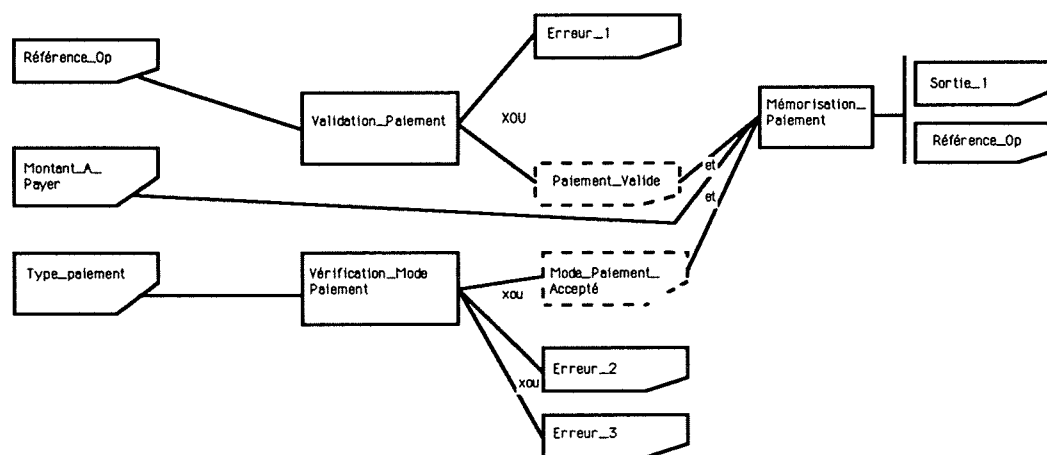
Applique la règle :

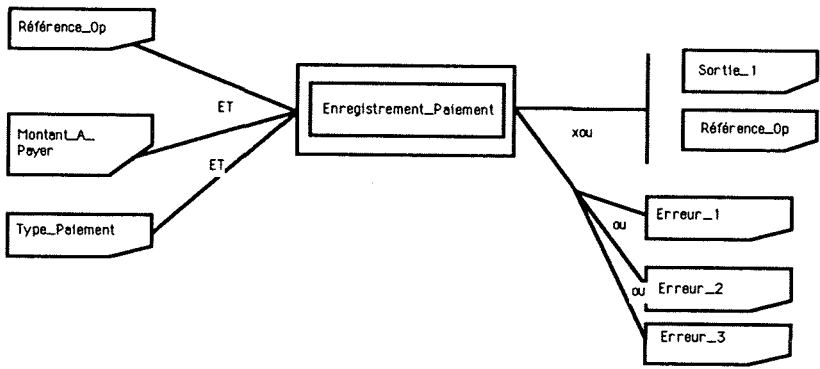
R1 : Le paiement n'est pas valide si la Référence_Op n'existe pas dans le S.I..

Remarque : Le paiement est possible car cette fonction ne peut être demandée s'il en est autrement.

Graphe d'enchaînement des fonctions atomiques :

Le graphe de la "boîte noire" et le graphe d'enchaînement des fonctions atomiques sont présentés sur les deux pages suivantes.





Graphe d'enchaînement des trois fonctions interactives.

Ce graphe reprend les trois graphes présentés selon le modèle de la "boîte noire" et présente la liaison entre eux.

Ainsi, entre le Traitement-Saisie et le Traitement-Numéro, quatre types d'actions peuvent être faites par l'utilisateur : Choix_Structure, Choix_Qté_Coupure, Encodage_Série_Numéro et Paquet_Coupure.

L'action Choix_Structure n'a lieu qu'une fois par opération.

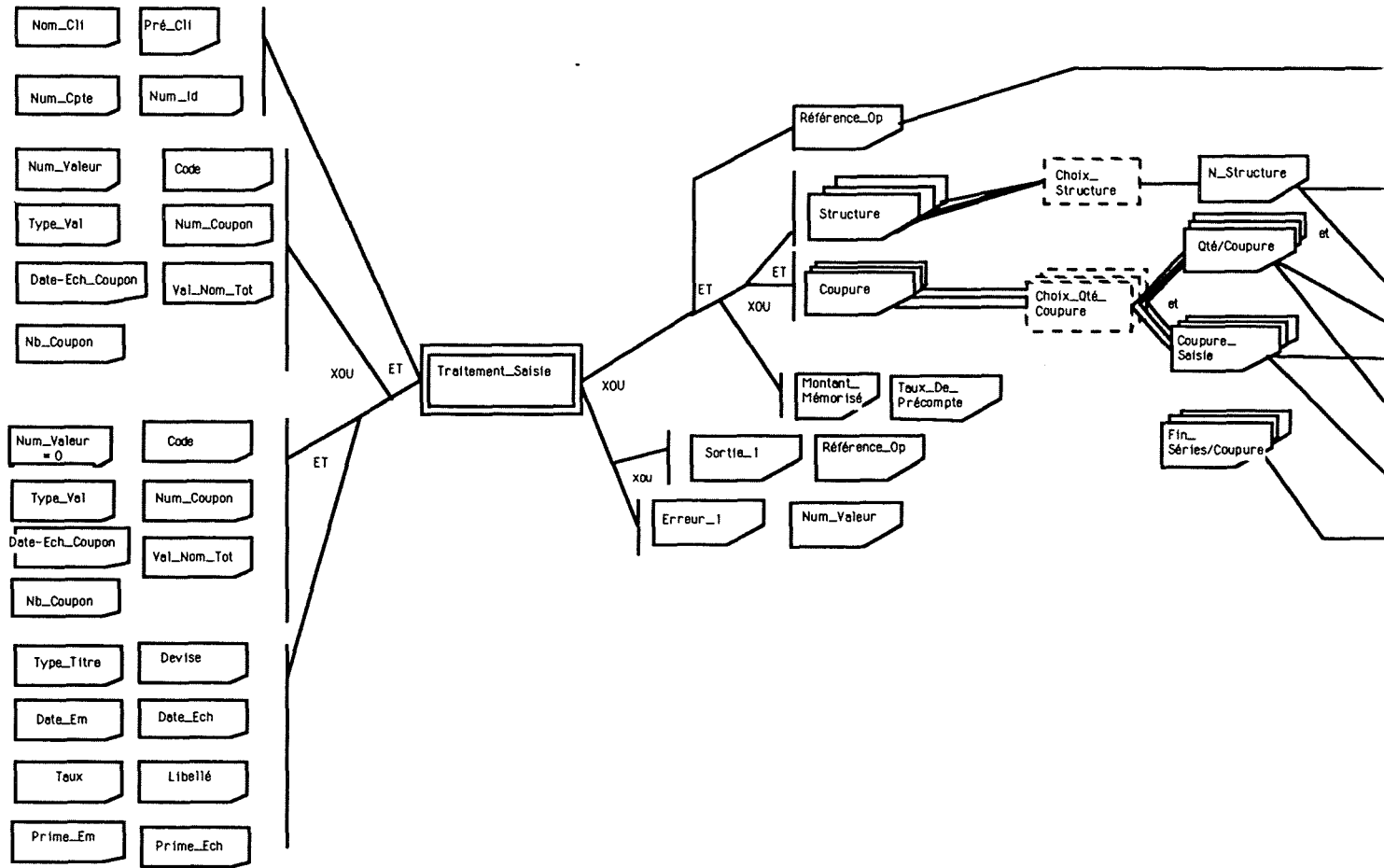
L'action Choix_Quantité_Coupure exprime le fait que l'utilisateur choisit une quantité pour la coupure reçue. Cette action aura lieu autant de fois qu'il y a de coupures. Lorsqu'il n'y aura pas de titre de la coupure reçue, la quantité (encodée implicitement) sera égale à zéro.

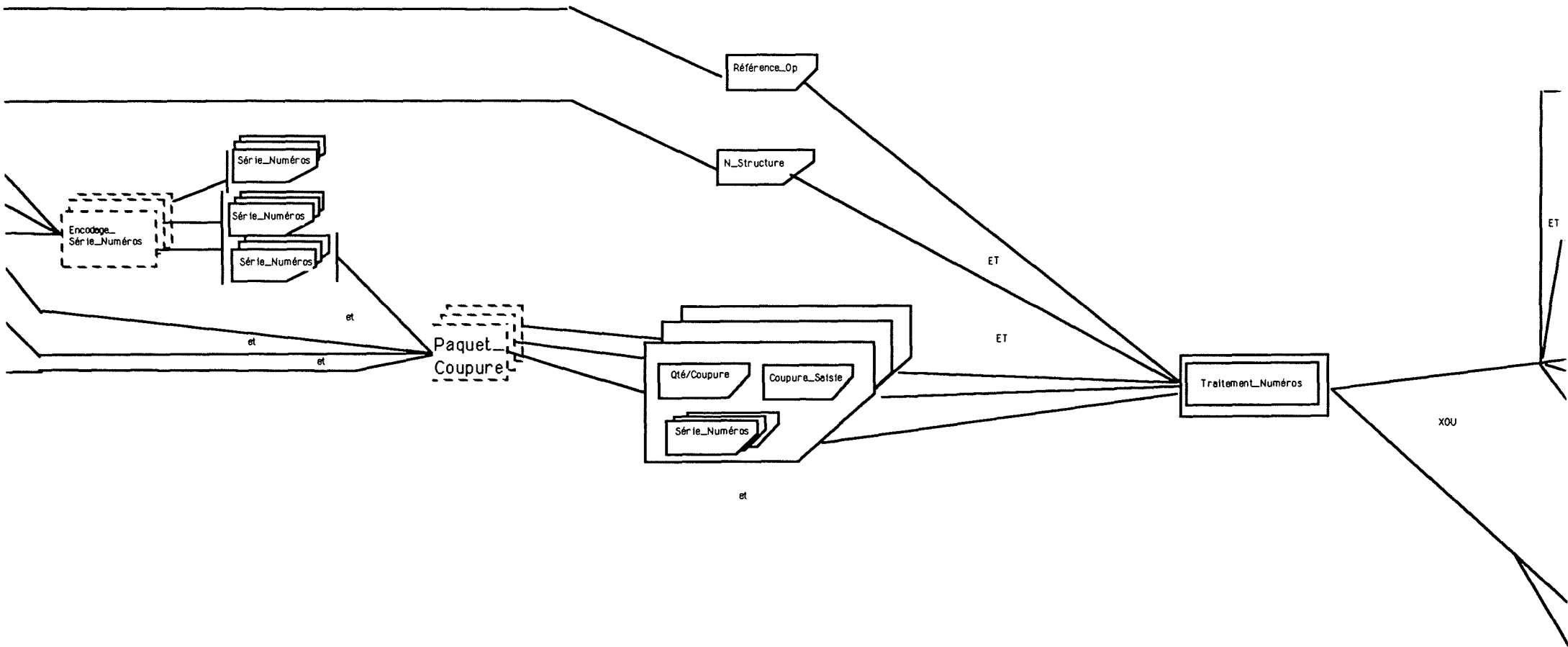
L'action Encodage_Série_Numéro représente l'encodage d'une série "De ... à ..." de numéros. Elle a lieu au minimum une fois par type de coupure choisi.

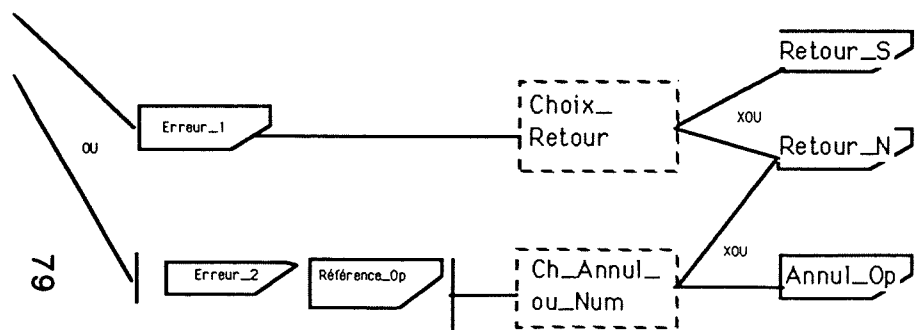
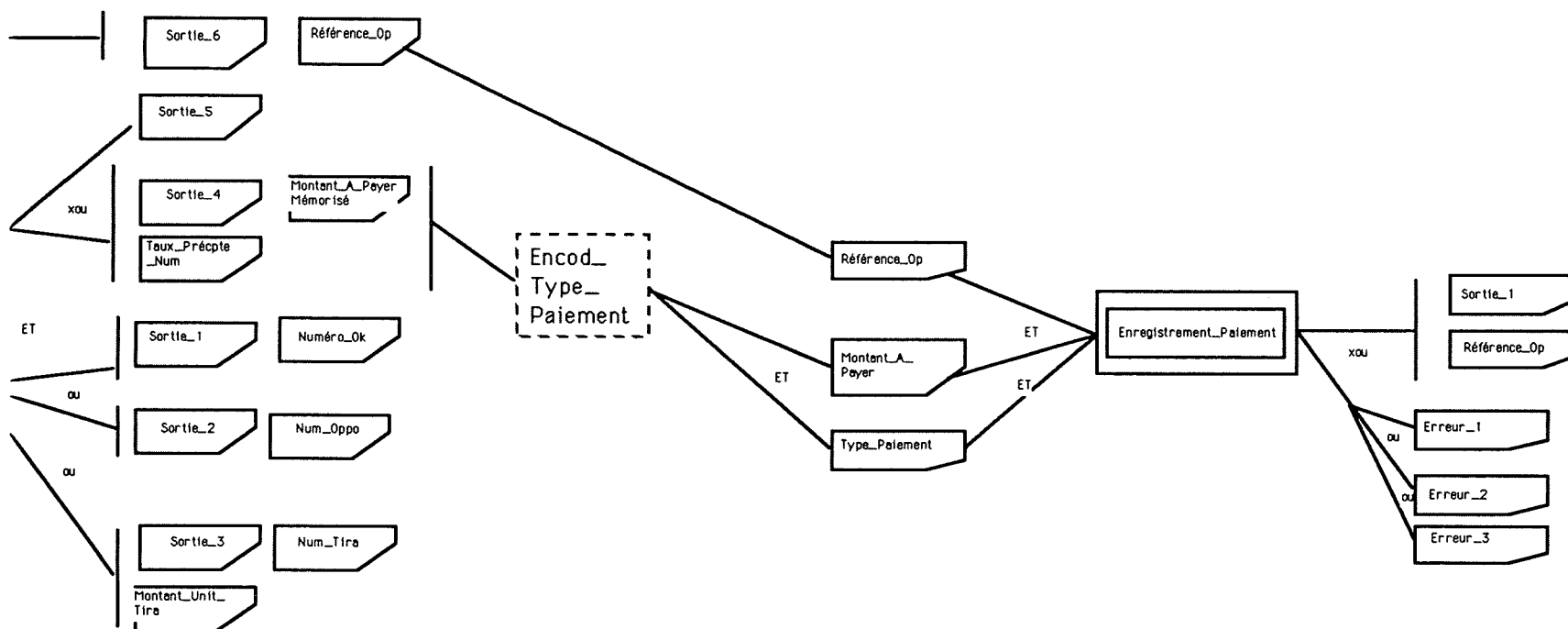
L'action Paquet_Coupure a lieu lorsque l'utilisateur encode la Fin-Séries/Coupure, c'est à dire lorsqu'il a terminé d'encoder les séries de numéros pour la coupure en question. Cette action a lieu une fois par type de coupure choisi. Elle reçoit la coupure, la quantité de cette coupure, les séries de numéros pour cette coupure et l'indication de Fin-Séries/Coupure.

Entre le Traitement_Numéro et le traitement Enregistrement_Paiement, trois actions sont possibles selon le résultat de Traitement_Numéros :

- L'action Encod_Type_Paiement est déclenchée lorsque le message externe d'affichage Sortie_4 est généré par Traitement_Numéros. Cette action a pour résultat un Type-Paiement et le Montant-A-Payer.
- L'action Choix_Retour a lieu lorsque le message externe d'affichage Erreur_1 est affiché par Traitement_Numéros. Cette action génère deux messages possibles : Retour_S et Retour_N. Le message Retour_S exprime la volonté de recommencer la saisie initiale (Valeur). Le message Retour_N signifie le retour à l'encodage des quantités/Coupure, des coupures et des numéros.
- L'action Ch_Annul_ou_Num est déclenchée si Erreur_2 est affichée par Traitement_Numéros. Elle permet de choisir entre l'annulation de l'opération et la reprise de l'encodage des coupures et numéros. Elle génère les messages externes Retour_N ou Annul_Op. Le message Annul_Op permet d'annuler purement et simplement l'opération en cours.



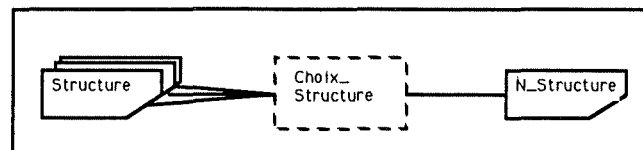




Propositions au niveau du formalisme :

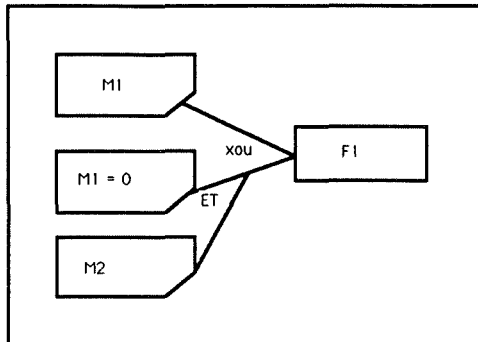
Pour garder la lisibilité, lorsqu'un groupe de messages sont générés et reçus, nous n'indiquons sur le graphe que les arcs concernant le premier message du groupe.

Pour exprimer le choix de l'utilisateur d'une valeur parmi plusieurs, nous pourrions choisir le rectangle en trait pointillé. Le rectangle parce qu'il s'agit d'une fonction au service de l'utilisateur. C'est le cas lorsque l'utilisateur doit choisir une Structure parmi plusieurs qui lui sont présentées :



E. Résumons le formalisme retenu.

1) Lorsqu'une fonction atomique reçoit un message M1 quand la valeur d'une donnée est différente de zéro, par exemple, et un message M1 plus un message M2 lorsque cette même donnée est égale à zéro, je propose ce formalisme-ci :



Exemple :

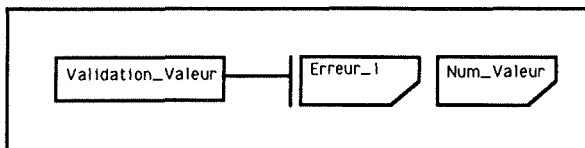
Dans le Traitement_Saisie, la fonction atomique Validation_Valeur reçoit un groupe de messages externes (Num_Valeur, Type_Valeur, Code, Num_Coupon, Date_Ech_Coupon, Val_Nom_Tot, Nb_Coupon) lorsque le message externe Num_Valeur est différent de zéro. Lorsque le message externe Num_Valeur = 0, ce même groupe de messages externes est reçu par la fonction accompagnée d'un autre groupe de messages externes (Type_Titre, Devise, Date_Em, Date_Ech, Taux, Libellé, Prime_Em, Prime_Ech).

2) Lorsque le message externe d'affichage comporte en lui-même une variable, le message et la variable associée sont distingués de sorte d'exprimer graphiquement la présence d'un élément dynamique dans le message :

Exemple :

Dans Traitement_Saisie, nous trouvons:

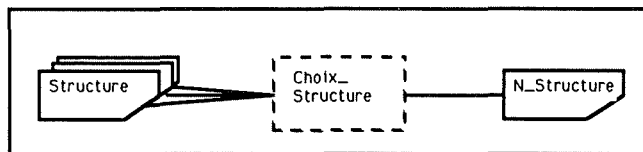
Erreur_1 : "Le Numéro_Valeur donné est incorrect : "Num_Valeur.



3) Lorsque nous voudrions exprimer graphiquement le choix d'un utilisateur parmi plusieurs valeurs présentées, nous pourrions le représenter par un rectangle en pointillés représentant l'intervention externe de l'utilisateur. Le rectangle parce qu'il s'agit d'une fonction mise au service de l'utilisateur.

Exemple :

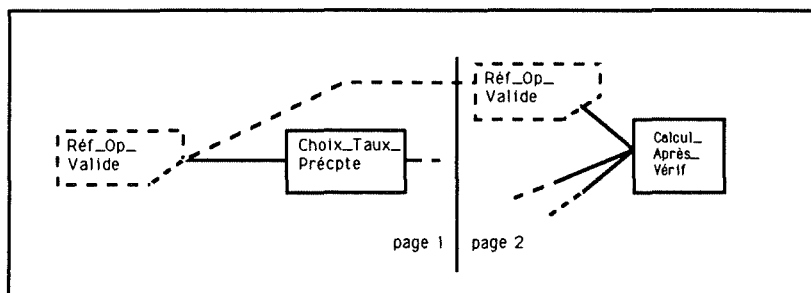
Lorsque le Traitement_Saisie affiche à l'utilisateur plusieurs structures, l'utilisateur choisit la structure adéquate de numérotation qui est dès lors renvoyée au Traitement_Num.



4) Lorsque un message interne est reçu par une fonction atomique distante (c'est-à-dire) sur une autre page, il serait bon dans un souci de lisibilité de rappeler le nom et la nature de ce message en employant le même formalisme que l'original, et d'utiliser une ligne en pointillés allant du message à son "double". Ainsi, le lecteur est prévenu de l'utilisation ultérieure du message ainsi que de sa répétition.

Exemple :

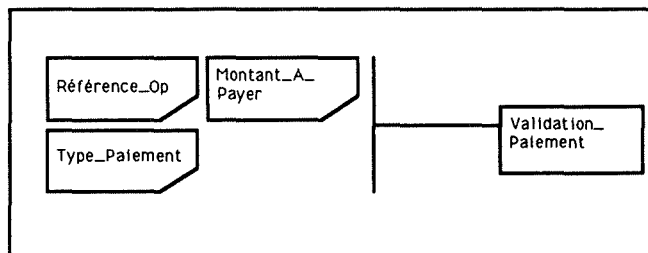
Le message Réf_Op_Valide du Traitement_Num.



5) Lorsqu'une fonction reçoit un groupe d'informations, l'emploi de la barre verticale, placée à côté des messages groupés, et reliée à la fonction réceptrice est proposé pour une raison de clarté et de lisibilité du graphique d'autant plus que ces informations ne se présenteront pas indépendamment l'une de l'autre. Ce formalisme a l'avantage de montrer la dépendance entre messages graphiquement par le regroupement.

Exemple:

La fonction Validation_Paiement de la phase Enregistrement_Paiement reçoit un groupe de trois messages externes :



Section 2.2. Conception de l'interface proprement dite

Nous avons à présent à notre disposition l'ensemble des fonctions de l'application, la définition des messages externes et les conditions de déclenchement des fonctions atomiques.

Nous pouvons dès lors entamer la conception de l'interface réelle. Par interface réelle, nous comprenons l'interface qui tient compte des contraintes contextuelles, techniques et de standardisation imposées par le problème et par la solution choisie.

Nous procéderons en trois étapes séquentielles. La première étape concerne le choix du style d'interaction, la deuxième étape définit le macro-dialogue en identifiant les micros-dialogues et la troisième étape définit la conversation au sein d'un micro-dialogue.

A. Choix au niveau de la présentation.

Le choix au niveau de la présentation signifie choisir le style d'interaction adéquat étant donné les contraintes issues du contexte de travail, du gestionnaire d'interaction cible et de la complexité de l'application.

Etant donné que le remboursement de titres est une opération de guichet, le temps d'exécution est un facteur important à considérer.

Cette opération demande l'encodage d'un nombre important d'informations. Dès lors, nous choisirons le formulaire qui semble le style d'interaction le plus adéquat accompagné d'un menu pour la sélection de l'opération sur titre.

Nous aurons donc :

- un menu général où un item est "Opérations sur Titre",
- un menu d'opérations sur titre où est présentée l'opération de Remboursement/Paiement Coupon,
- des formulaires pour les saisies Client, Valeur et Numéros.

B. Choix au niveau de la macro-conversation.

Le macro-dialogue définit l'enchaînement des micro-dialogues.

Etant donné l'application, nous savons qu'elle se divise en trois grandes parties :

- le traitement de la première saisie ;
- le traitement des numéros ;
- l'enregistrement du paiement.

a. Le traitement de la première saisie.

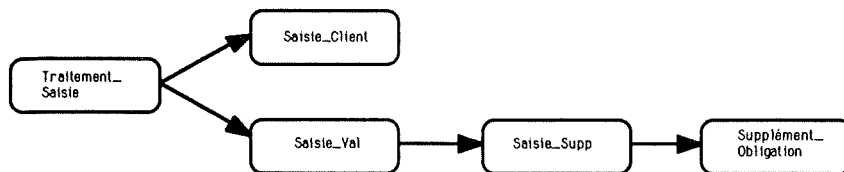
Nous pouvons le subdiviser en trois unités de saisie :

- la saisie de l'identification du client ;
- la saisie de l'identification de la valeur ;
- la saisie d'informations supplémentaires sur la valeur.

Elles correspondent à trois unités de dialogues ou micro-dialogues autonomes ayant un objectif et se composant d'un ensemble d'objets interactifs (supportant le dialogue avec l'utilisateur final).

Le déclenchement sera modal : l'utilisateur sera obligé de clôturer le micro-dialogue avant de faire autre chose. Il ne pourra donc faire qu'une chose à la fois. Cela paraît préférable étant donné l'univers du guichet dans lequel il travaille.

Le macro-dialogue du Traitement-Saisie peut être décrit à l'aide d'un graphe d'enchaînement dont les sommets représentent les micro-dialogues et les axes les enchaînements possibles. Le déclenchement modal est représenté par la simple flèche. Le déclenchement modeless serait représenté par une flèche sur laquelle se trouvent deux barres parallèles.



b. Le traitement des numéros.

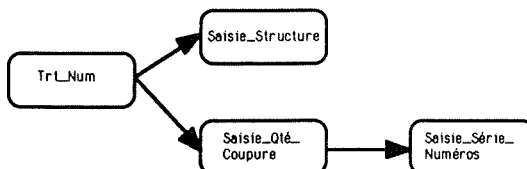
Il possède trois unités de dialogue :

- le choix du format de numérotation ;
- l'encodage des quantités par coupure et des coupures ;
- l'encodage des numéros par série.

Etant donné que le choix du format ne se fera pas chaque fois et qu'il sera toujours préalable à l'encodage des numéros, nous le différencions du reste de l'encodage.

Etant donné qu'il est possible d'encoder plusieurs séries de numéros pour un type de coupure, j'ai différencié ces deux encodages afin d'avoir un type de formulaire par type d'encodage.

Le graphe d'enchaînement de ce micro-dialogue est le suivant :

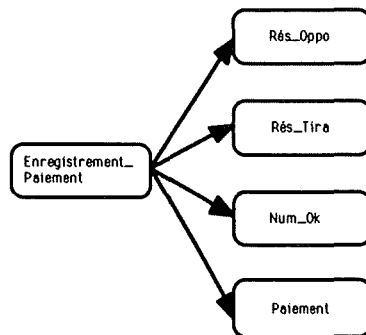


c. L'enregistrement du paiement.

Il se subdivise en quatre unités de dialogue selon le résultat des vérifications :

- affichage des numéros soumis à l'opposition ;
- affichage des numéros soumis au tirage ;
- affichage des numéros qui sont "OK" ;
- affichage du montant à payer et saisie du type de paiement désiré.

Ces quatres unités de dialogue étant indépendantes, leur graphe d'enchaînement se présente comme ceci :

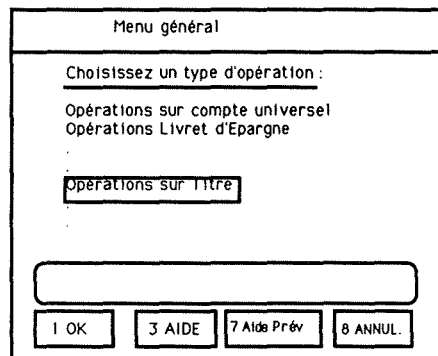


C. Choix au niveau de la conversation et conception de la maquette.

A ce niveau, nous allons décrire les micro-dialogues ainsi que la conversation au sein de ceux-ci. Nous allons tenter de justifier chaque choix à l'aide des règles ergonomiques de Scapin (SCA, 87) ou à l'aide de la norme CUA d'IBM (IBM, 89a et 89b). Les micro-dialogues retenus sont le menu sous forme de liste et le formulaire.

Pour définir la conversation, nous avons employé le langage commun. Néanmoins, étant donné la séquentialité, nous pourrions employer le formalisme du diagramme états-transitions.

Le menu permet la sélection des opérations sur titres parmi d'autres types d'opération. Un second menu apparaît une fois l'item d'Opérations sur titre sélectionné. Ce menu permet la sélection de l'opération de remboursement parmi les opérations sur titre. Etant donné le nombre élevé d'opérations possibles en agence, le menu principal se présente sous forme de liste :



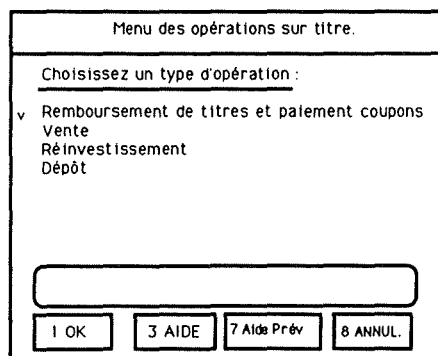
Menu général

Choisissez un type d'opération :

- Opérations sur compte universel
- Opérations Livret d'Epargne
- Opérations sur titre

1 OK 3 AIDE 7 Aide Prév 8 ANNUL.

Le menu d'opération sur titres peut quant à lui se présenter sous la forme d'une barre de menu étant donné qu'il n'y a que quatre opérations sur titres :



Menu des opérations sur titre.

Choisissez un type d'opération :

- v Remboursement de titres et paiement coupons
- Vente
- Réinvestissement
- Dépôt

1 OK 3 AIDE 7 Aide Prév 8 ANNUL.

Les formulaires seront composés d'une zone de saisie/affichage des messages externes, d'une zone de titre, d'une zone réservée aux messages d'erreur et d'une zone réservée aux labels des touches fonctions. Les fonctions de l'application recevant ces différentes saisies sont déclenchées de façon implicite lorsqu'elles reçoivent tous les messages requis pour leur exécution.

Chaque formulaire nécessite la confirmation explicite de l'utilisateur pour exprimer la fin de la saisie pour ce formulaire (déclenchement modal).

Au sein d'un formulaire, nous passerons à l'item suivant de façon implicite si l'item qui vient d'être saisi avait une longueur fixe obligatoire. Nous passerons à l'item suivant de façon explicite par un ENTER lorsque l'item que nous venons d'encoder n'a pas une longueur obligatoire mais une longueur maximale. La longueur maximale d'un item sera mise en vidéo inversé afin d'en informer l'utilisateur, conformément aux règles CUA (IBM, 89a) page 93. L'entrée d'informations se fera par remplacement direct des caractères (petits rectangles) mis en vidéo inversé conformément à Scapin (SCA, 87) page 25.

Nous disposerons à chaque étape des touches fonctions F1 étiquetée OK, F2 étiquetée CORR et F3 étiquetée AIDE. Quand d'autres fonctions seront utiles, elles étiqueteront d'autres touches fonction que ces trois-là.

La touche F1 permettra de confirmer la fin de la saisie au sein d'un formulaire (application de Scapin (SCA, 87) page 26).

La touche F2 intitulée CORR permet quant à elle de libérer l'utilisateur du carcan dans lequel l'ensère le dirigisme du curseur. Si l'utilisateur s'aperçoit d'une erreur, il peut, en appuyant sur CORR et à l'aide des touches fléchées, positionner le curseur à l'endroit de l'erreur et corriger l'erreur par superposition. Eventuellement (si la longueur de l'item n'est pas obligatoire), il doit confirmer son encodage par ENTER.

On peut ici discuter de la raison d'être de CORR. Pourquoi, en effet, ne pas laisser l'utilisateur libre d'encoder le formulaire comme il l'entend ? Pourquoi le dirige-t-on ainsi ?

Considérons le cas où l'utilisateur serait libre : il pourrait voyager d'un item à l'autre. Il pourrait commencer l'encodage d'un item, l'interrompre et commencer l'encodage d'un autre item, revenir au précédent plus tard, etc... A ce moment, toute validation ne pourrait avoir lieu qu'après la confirmation de fin de saisie du formulaire entier. Ceci représente un net désavantage par rapport à l'encodage dirigé.

De plus, quel peut être l'intérêt pour le guichetier de ne pas encoder séquentiellement et de passer d'un item à l'autre ? A la réflexion, aucun. En effet, étant donné le poste de travail qu'est le guichet, le facteur temps joue un rôle prépondérant. Le fait d'être dirigé et d'avoir la confirmation implicite quand c'est possible favorise le travail du guichetier : il est aidé à l'encodage, le temps d'exécution est réduit.

Dès lors, le dirigisme étant la meilleure solution, il doit du moins pouvoir être contourné dans certains cas pour permettre la correction. C'est là le rôle attribué à F2.

La touche F3, elle, permettra de faire appel à l'aide.

Une aide préventive est cependant recommandée pour l'utilisateur novice ou occasionnel. Lorsque le curseur passe au champs de saisie de l'item suivant, une aide à l'encodage brève et précise permet à l'utilisateur de savoir comment procéder dans l'immédiat sans devoir recourir à l'aide souvent plus longue (F3). Ce message d'aide préventive s'affiche dans la zone du message d'erreur (conformément à (IBM, 89a) page 181 et 183). Etant donné qu'une même zone va avoir deux utilisations différentes, nous proposons d'afficher les messages d'erreur en vidéo inversé afin de retenir l'attention de l'utilisateur.

Afin d'utiliser cette aide préventive, l'utilisateur pourrait avoir recours à une touche fonction intitulée AIDE PREV. Lorsque cette aide est utilisée, la touche fonction change de nom et de rôle. Elle devient FIN AIDE PREV et permet la suppression de l'affichage de l'aide préventive. Cela peut s'avérer particulièrement intéressant pour l'utilisateur expérimenté qui ne souhaiterait pas voir constamment cette aide préventive. Nous pourrions attribuer cette fonction à la touche fonction F7, par exemple.

Enfin, nous devons prévoir la possibilité d'annuler l'opération en cours. Cette fonction présente l'intérêt de pouvoir recommencer l'opération sur titre en cours lorsque, par exemple, un erreur importante d'encodage est constatée. Nous attribuerons cette fonction à la touche fonction F8.

La dernière ligne de l'écran sera réservée aux étiquettes des touches fonction accessibles du micro-dialogue courant. Cette disposition est conforme aux normes CUA (IBM, 89a) page 20 et à Scapin (SCA, 87) pages 20 et 21.

a. Le Traitement-Saisie

Le Traitement-Saisie se subdivise en trois parties :

- Saisie-Client ;
- Saisie-Valeur ;
- Saisie-Suppl.

a.1 Le formulaire Saisie-Client

The diagram shows a rectangular form titled "Identification du Client". It is divided into several sections. At the top is a title bar. Below it are four input fields: "Nom :", "Prénom :", "Num-Cpte :", and "Num-Id :", each followed by a blacked-out input area. Below these is a horizontal line. At the bottom is a row of five buttons: "1 OK", "2 Corr", "3 Aide", "7 Aid. Prev", and "8 ANNUL.". To the right of the form, four arrows point to specific areas with labels: "zone de titre" points to the title bar, "zone de saisie" points to the input fields, "zone de message d'erreur" points to the horizontal line, and "zone réservée aux labels des touches fonctions" points to the button row.

La présentation des items a été choisie en fonction de la fréquence d'encodage de ceux-ci ((SCA,87) page26). Le Nom du client sera toujours encodé même s'il est anonyme. Si le Nom est différent de Anonyme, le Prénom sera également encodé.

Le Numéro de compte (titre ou espèce) sera toujours donné sauf si le client n'a pas de compte à la CGER. A ce moment, son numéro de carte d'identité devra obligatoirement être encodé.

Le numéro de la carte d'identité sera également encodé si le client, ayant donné son Nom, son Prénom et son Numéro de compte, n'est pas un client habituel de l'agence, c'est à dire s'il n'est pas repris dans la base de données Client de l'agence. La fin de l'encodage du Nom et du Prénom doit être confirmée par ENTER. La fin de l'encodage de Num-Cpte et de Num-Id est confirmée implicitement puisque leur longueur est fixe.

La touche fonction F1 est étiquetée OK et permet la confirmation de l'encodage de l'identification du Client. Elle apparaît en grisé quand rien n'a été encodé.

La touche fonction F2 est étiquetée CORR et permet à l'utilisateur de corriger n'importe quel item et de voyager d'un champ de saisie à l'autre à l'aide des touches fléchées. La correction se fait par surcharge ((SCA, 87) page 26).

La touche fonction F3 permet de faire appel à l'aide générale propre à ce formulaire.

Si le Nom est Anonyme, alors le Num-Cpte doit être un compte de l'agence sinon un message d'erreur est affiché.

Pour Num-Cpte et Num-Id, les tirets entre groupes de numéros sont apparents mais le guichetier entre les douze numéros l'un à la suite de l'autre sans devoir marquer le saut d'un groupe à l'autre.

a.2 Le formulaire Saisie-Valeur.

Identification de la Valeur.	
Numéro-Valeur :	
Code :	
Type de valeur :	
Num-Coupon :	OU Date-Ech-Coupon :
Valeur Nominale Totale :	
Nombre de Coupons :	
1 OK	2 CORR
3 AIDE	7 Aid.Prév.
8 ANNUL.	

Le Numéro-Valeur qui identifie la valeur et conditionne la suite de l'opération a été placé en première ligne (SCA, 87) page 26. Sur la même ligne, le Code suit. En effet, sans son Code, le Numéro-Valeur ne peut être identifié. Le champ de saisie du code présente une valeur par défaut (application de (SCA, 87) page 25).

L'item Type-de-Valeur, quant à lui, conditionne l'encodage ou non des items Num-Coupon, Date-Ech-Coupon et Nbre-Coupons. De plus, le fait que la valeur présentée soit du Type Manteau, Coupon ou Manteau + Coupon est la première caractéristique qui saute aux yeux. Il est donc logique de placer cet item directement après l'identifiant.

Les items Num-Coupon et Date-Ech-Coupon identifient chacun une sorte de coupon. Ils sont exclusifs. L'utilisateur encodera l'un des deux selon que le coupon est un coupon d'action ou un coupon d'obligation. C'est pour cette raison que je les ai placés sur la même ligne avec un "OU" entre eux.

Viennent ensuite les items Valeur-Nominale-Totale et Nbre-Coupons. Ils expriment tous les deux la notion de quantité, l'un au niveau du titre et l'autre au niveau du coupon. C'est ainsi que je les ai regroupés après l'identification éventuelle du coupon.

Les items Num-Valeur, Num-Coupon, Valeur-Nominale-Totale et Nbre de Coupons demandent la confirmation de l'utilisateur étant donné leur longueur dynamique.

Lorsque le curseur est sur Code, l'étiquette de la touche fonction F4 s'affiche sous le nom de Liste. En tapant F4, la liste des codes possibles se superpose au formulaire. L'utilisateur sélectionne l'item avec les touches fléchées + ENTER. Dès qu'un ENTER est frappé, la liste se referme et le curseur passe sur le champs de saisie de l'item Type-de-Valeur. Pour encoder le Code, l'utilisateur peut également entrer directement la lettre correspondant au Code.

Quand le curseur est sur le champs de saisie de l'item Type-de-Valeur, trois étiquettes de touches fonction s'affichent : la touche F4 s'étiquette MANTEAU, la touche F5, COUPON et la touche F6, MANTEAU + COUPON. Seul l'encodage par une de ces touches est permis. Si l'utilisateur essaie autrement, un bip sonore se fait entendre.

Les items Num-Coupon et Date-Ech-Coupon se mettent en grisé si le Type-de-Valeur = Manteau.

Les items Num-Coupon et Date-Ech-Coupon sont mutuellement exclusifs. Le numéro du coupon est à encoder lorsque la valeur est une action, tandis que Dat-Ech-Coupon est à encoder quand la valeur est une obligation. Dès que l'utilisateur a commencé l'encodage d'un de ces deux items, l'autre se met en grisé (application du guidage de(SCA, 87) page28).

L'utilisateur peut corriger à l'aide de F2 et peut faire appel à l'aide en tapant F3.

La Valeur-Nominale-Totale est la somme des produits (Qté/Coupure x Coupure).

L'item Nbre-de-Coupons est mis en grisé si le Type-de-Valeur est un manteau ((SCA,87) page 28).

La confirmation de l'encodage de ce formulaire a pour résultat :

- soit l'affichage d'un message signifiant à l'utilisateur d'attendre le résultat de la vérification si le Num-Valeur est différent de zéro ;
- soit l'affichage du formulaire d'encodage d'informations supplémentaires si le Num-Valeur est nul.

a.3 Le formulaire d'informations supplémentaires.

Informations supplémentaires.

Type-Titre :

Libellé :

Devise :

1 OK 2 CORR 3 AIDE 4 ACTION 5 OBLIG. 7 Aid. Priv. 8 ANNUL.

zone d'étiquettes des touches fonctions
quand le curseur est sur le champs
de saisie de Type-Titre

Ce formulaire présente les trois items communs à l'obligation et à l'action.

Une fenêtre s'ouvrira sur un autre formulaire appelé Supplément Obligation si le titre encodé est une obligation et dès que l'encodage du Type-Titre sera effectué(guidage (SCA, 87) page 28).

L'encodage du Type-Titre ne sera accepté que par les deux touches fonction F4 étiquetée ACTION et F5 étiquetée OBLIGATION. Ce choix a été fait dans un souci de guider au mieux l'utilisateur en ménageant sa mémoire à court terme (concision (SCA, 87) page 13).

Le Libellé représente le nom figurant sur la valeur et permettant de la reconnaître.

La Devise peut être encodée à partir du clavier, ou bien à partir de la touche fonction F4 qui fait apparaître une liste des devises possibles. L'utilisateur se positionne sur la devise adéquate à l'aide des touches fléchées en confirmant par ENTER (application à nouveau du principe de la concision (SCA, 87) page 13).

La confirmation de l'encodage du formulaire d'informations supplémentaires a pour effet de faire apparaître un message d'attente (principe du retour d'informations (SCA, 87) page 33).

a.4 Le formulaire Supplément Obligation.

Informations supplémentaires.

Type-Titre : OBLIGATION

Supplément Obligation

Date d'émission : [REDACTED]

Date d'échéance : [REDACTED]

Prime d'émission : [REDACTED]

Prime d'échéance : [REDACTED]

[Empty input box]

1 OK 2 CORR 3 AIDE 7 Aid.Prév. 8 ANNUL.

ligne de touches fonctions disponibles à partir de Supplément Obligation

Les dates d'émission et d'échéance sont obligatoires. C'est pour cette raison qu'elles sont placées en premier lieu et de façon chronologique ((SCA, 87) page 26).

Les Primes d'Emission et de Remboursement ne sont pas obligatoires (car tous les titres n'en bénéficient pas), c'est pourquoi elles sont placées en dernier lieu et de façon chronologique entre elles ((SCA, 87) page 26).

Après la confirmation de l'encodage de ce formulaire à l'aide de F1, la fenêtre de ce formulaire se referme et le curseur se place sur le champs de saisie de l'item Libellé du formulaire d'informations supplémentaires (retour d'information (SCA,87) page 33).

b. Le traitement des numéros.

b.1 Le formulaire Saisie-Structure.

Choix du format des numéros.

Format N° ... = - / - / -

Votre choix : N°

seul champs de saisie

1 OK 3 AIDE 7 Aide Prév 8 ANNUL.

Ce formulaire est affiché en réponse au Traitement Saisie si plusieurs formats de numérotation sont possibles. Il présente une liste des formats possibles avec le numéro permettant leur identification. Un seul choix est possible.

S'il n'y a qu'un format possible, le formulaire Saisie-Qté-Coupure est affiché en réponse au traitement Saisie.

b.2 Le formulaire Saisie-Qté-Coupure.

Saisie des quantités et des coupures.

Coupures de _____

Coupures de _____

Coupures de _____

1 OK 2 CORR 3 AIDE 7 Aid.Prév. 8 ANNUL.

Dans ce formulaire, les coupures possibles sont affichées.

L'utilisateur doit encoder la quantité d'une coupure dans la ligne correspondant à la coupure adéquate.

L'utilisateur choisit la coupure voulue à l'aide des touches fléchées. Il encode la quantité et confirme par ENTER. Quand il veut corriger un item, à l'aide des touches fléchées il se positionne sur le champs de saisie de cet item, tape F2 et peut ainsi encoder, par surcharge(correction (SCA, 87) page 26), la quantité voulue et confirmer par ENTER.

La confirmation de l'encodage d'une quantité avec ENTER a pour effet d'ouvrir le formulaire de saisie des numéros correspondants à la coupure choisie.

Une fois la confirmation de fin d'encodage des numéros introduite, le formulaire de saisie des numéros se referme et le curseur se déplace vers le champs de saisie de la quantité de la coupure suivante (principe du retour d'informations (SCA, 87) page 33).

La confirmation de la fin de l'encodage des quantités et coupures a pour effet de refermer la fenêtre du formulaire Saisie-Qté-Coupure et d'afficher un message d'attente (retour d'informations, idem).

b.3 Le formulaire Saisie-Numéros.

Saisie des numéros.

_____ Coupures de _____

De [] à []

De [] à []

De [] à []

[]

1 OK 3 AIDE 7 Aide Prév. 8 ANNUL

Cette ligne reprend la ligne qui vient d'être encodée.

La ligne Qté/Coupure et Coupure qui concerne les numéros qui vont être encodés est affichée.

Ensuite, viennent trois lignes "De ... à ..." où les champs de saisie ont le format choisi dans le formulaire Saisie-Structure (ou le seul format possible si aucun choix n'a été effectué) (guidage (SCA, 87) page 28) .

La touche fonction F1 permet de confirmer la fin de l'encodage et de retourner au formulaire Saisie-Qté-Coupure.

La touche fonction F8 permet d'annuler, c'est à dire de refermer la fenêtre sans avoir effectué d'encodage. Cette hypothèse est valable lorsque l'utilisateur s'est trompé dans le choix de la coupure.

Lorsqu'il n'y a pas de numéros qui se suivent, et donc pas de série, l'utilisateur encode chaque numéro dans la première partie (De ...) de la ligne.

Lorsque les trois lignes sont encodées, un message apparaît pour demander à l'utilisateur s'il veut encore des lignes d'encodage de numéros pour cette coupure. Si la réponse est oui, les champs saisis disparaissent et l'écran fait de nouveau apparaître la ligne "...Coupure de ..." ainsi que trois lignes de saisie de numéros. L'affichage du mot "Suite" permet à l'utilisateur de savoir qu'il est à la suite de l'écran précédent (feedback et guidage (SCA, 87) page 14).

Au lieu d'avoir un message de demande de lignes supplémentaires, on pourrait avoir à droite de la dernière ligne un bouton avec une flèche orientée vers la droite dont le libellé serait "Suite". Sa sélection aurait pour but d'afficher de nouvelles lignes de saisie de séries de numéros.

c. Le traitement Enregistrement-Paiement.

c.1 Le formulaire d'affichage Rés-Oppo.

Ce formulaire affiche les séries de numéros frappés d'opposition.

L'utilisateur tape F1 = OK pour dire qu'il a terminé la lecture du formulaire.

La touche fonction F4 =SUITE est en grisé lorsqu'il n'y a pas de fenêtre qui suit c'est à dire lorsque la consultation de la liste est terminée (défilement (SCA, 87) page 47).

C'est dans la zone de message d'erreur que serait renseignée l'information précisant que le paiement est permis ou non.

c.2 Le formulaire d'affichage Rés-Tira.

Ce formulaire affiche les séries de numéros tirés.

En fin d'affichage de ces séries de numéros, le montant unitaire est indiqué.

Section 3. Conception d'une interface moderne

Nous allons à présent considérer la solution que nous pourrions avoir si nous disposions d'une technique de pointe nous permettant l'usage de la souris et de tous les objets interactifs avancés existants ainsi que l'accès à la base de données locale. Nous ne supprimons pas les contraintes techniques mais nous les modifions.

Cet exercice nous permettra de tester plus en avant la démarche de conception d'interface homme-machine et l'utilisation de l'arbre de sélection d'objets interactifs abstraits. Etant donné les hypothèses techniques décrites ci-avant, ce test s'appliquera à une interface dont la complexité de conception sera accrue par rapport à l'interface réelle.

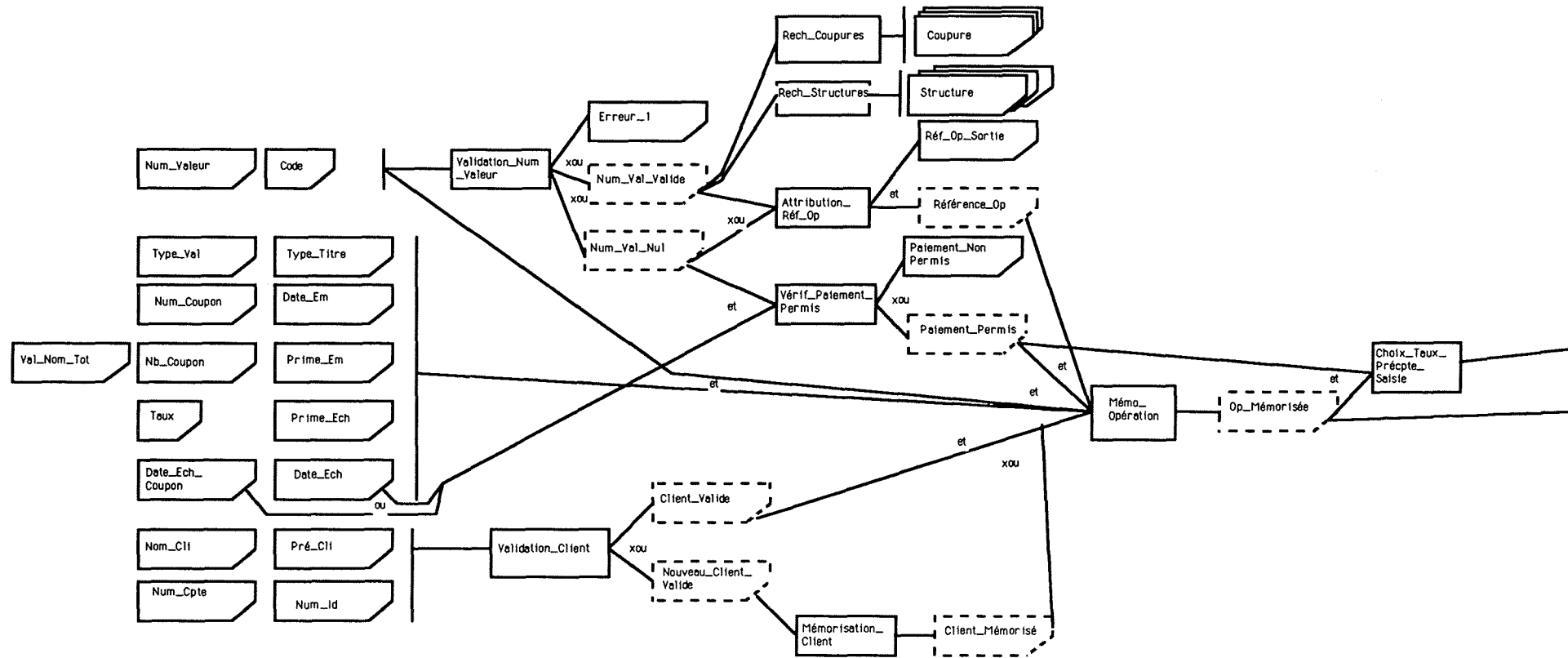
Nous n'allons pas, comme nous l'avons fait pour la solution réelle, présenter chacune des étapes de la conception. Nous nous concentrerons, au contraire, à mettre en évidence les changements (ex : les nouvelles fonctions atomiques) et les différences constatées au niveau ergonomique au travers de la comparaison des graphes d'enchaînement des fonctions atomiques des fonctions interactives ainsi qu'au travers de la comparaison des deux maquettes d'interface obtenues.

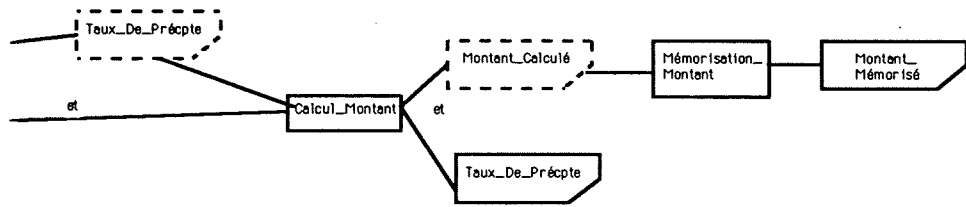
Nous pourrions faire ici une analyse abstraite de la tâche étant donné qu'à ce stade-ci, nous ne connaissons pas les choix qui auront lieu dans les étapes ultérieures. Cette analyse permettrait de présenter une décomposition de la tâche indépendante d'actions concrètes à effectuer.

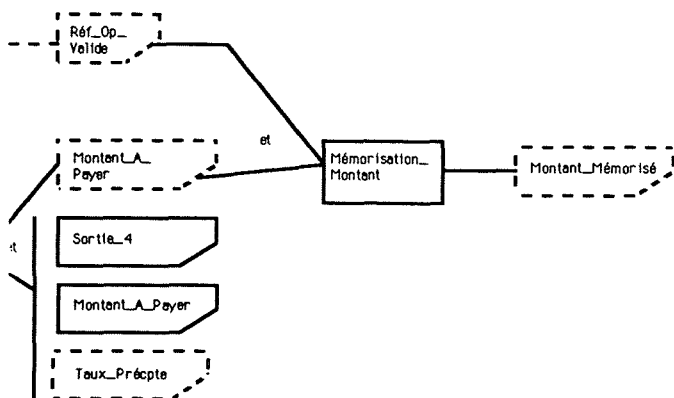
Sous-Section 3.1 Comparaison des graphes d'enchaînement des fonctions atomiques.

A. Comparaison au niveau du graphe d'enchaînement des fonctions atomiques

Les quatre pages suivantes présentent les graphes d'enchaînement obtenus pour le Traitement-Saisie et pour le Traitement-Num envisagés sous les hypothèses exposées ci-dessus.







TRAITEMENT - NUMEROS (Suite)

A.1 Voyons tout d'abord ce qui se passe au niveau de la phase Trt-Saisie.

- A.1.1** Dès que le Numéro-Valeur et son Code ont été encodés, la fonction Validation-Num-Valeur peut être exécutée. Dès lors, si l'utilisateur se trompe, il est immédiatement prévenu et peut corriger son erreur avant de poursuivre l'encodage.
- A.1.2** Dès que les Nom et Prénom du client sont encodés, ils peuvent être validés et mémorisés. Dans la solution réelle, la validation et la mémorisation des données Client se font quand tout le groupe d'items est encodé.
- A.1.3** Aussitôt que la date d'échéance est encodée et si le Numéro-Valeur est nul, la fonction Vérif-Paiement-Permis se déclenche et permet de savoir plus rapidement si le paiement est permis ou non.
- A.1.4** Nous allons reprendre les différences issues de la comparaison des deux graphes en les regroupant en cinq ensembles : les nouvelles fonctions, les fonctions modifiées au niveau du traitement ou au niveau des messages reçus et générés, les fonctions disparues, les nouveaux messages externes d'affichage et de saisie et les messages internes ou externes disparus.

a. Nouvelles fonctions :

- Validation-Num-Valeur :
A pour objectif de vérifier la présence du Numéro-Valeur dans le S.I.
Reçoit : Num-Valeur et Code.
Génère : Erreur-1, Num-Val-Valide et Num-Val-Nul.
- Attribution-Réf-Op :
A pour objectif d'attribuer une Référence-Op à la nouvelle opération sur titre.
Reçoit : Num-Val-Nul et Num-Val-Valide.
Génère : Ref-Op-Sortie et Référence-Op.

b. Fonctions modifiées :

- Rech-Coupures remplace Recherche-Coupures.
A pour objectif de rechercher les coupures émises et de les afficher.
Reçoit : Num-Val-Valide.
Génère : Coupure.
- Rech-Structures remplace Recherche-Structures.
A pour objectif de rechercher la (les) structure(s) de numérotation.
Reçoit : Num-Val-Valide.

Génère : Structure.

- MémO-Op remplace Mémorisation-Opération.
A pour objectif de mémoriser le client et la valeur.
Reçoit : Référence-Op, Type-Val, Nb-Coupon,
Paiement-Permis, Num-Valeur, Code, Type-Titre,
Num-Coupon, Prime-Em, Prime-Ech, Taux, Date-
Ech, Date-Em, Date-Ech-Coupon, Client-Valide,
Client-Mémorisé.
Génère : Op-Mémorisée.
- Vérif-Paiement-Permis remplace Vérification-
Paiement-Permis-Saisie.
A pour objectif de vérifier si le paiement est permis.
Reçoit : Num-Val-Nul, Date-Ech-Coupon et Date-
Ech.
Génère : Paiement-Non-Permis ou Paiement-
Permis.

c. Fonction disparue : Validation-Valeur.

d. Nouveaux messages internes :

- Num-Val-Valide
comprend un Num-Valeur.
- Num-Val-Nul
comprend un Num-Valeur égal à zéro.

e. Nouveaux messages externes d'affichage :

- Réf-Op-Sortie remplace Référence-Op.
- Paiement-Non-Permis remplace Sortie et Référence-
Op.

f. Messages internes disparus :

- Valeur-Valide.
- Valeur-Valide-Num-Val-Nul.

g. Message externe disparu : Référence-Op.

A.2 Passons à présent à la phase Trt-Num.

Il apparaît clairement, à l'examen des deux graphes, que le deuxième est plus concis que le premier grâce à davantage de parallélisme.

A.2.1 La Référence-Op peut être immédiatement validée et l'utilisateur de suite prévenu d'une erreur d'encodage éventuelle et du fait que le paiement est permis ou non.

A.2.2 Dès qu'une ligne Qté/Coupure et Coupure-Saisie est

encodée, elle est immédiatement mémorisée dans le S.I. par Mémo-Qté-Coupure. Dans la solution précédente, il fallait attendre l'encodage total et la Validation-Qté avant la mémorisation.

- A.2.3 Dès qu'une série de Numéros (De...à...) est encodée, celle-ci peut être vérifiée. S'il y a opposition sur cette série, l'utilisateur est immédiatement prévenu par un message. Ceci constitue une grande différence par rapport à la solution réelle : les vérifications se font ligne par ligne, prévenant l'utilisateur au fur et à mesure du résultat de ces contrôles.
Les traitements Vérification-Oppo et Vérification-Tira sont déclenchés pour chaque ligne Série-Numéros et non plus par la Saisie-Num-Mémorisés qui reprenait toutes les lignes Séries-Numéros.
- A.2.4 Un phénomène d'attente apparaît : une nouvelle fonction atomique Attente-Mémo-Résultat-Vérif mémorise les résultats des vérifications lorsque l'encodage est terminé, c'est-à-dire lorsque le message externe Fin-Encodage-Num lui parvient. Elle globalise les différents résultats en regroupant les messages Num-Oppo-Ligne, Num-Tira-Ligne et Num-OK-Ligne en trois messages Num-Oppo, Num-Tira et Num-OK.
- A.2.5 La fonction Vérif-Paiement-Permis-Num n'a pas les mêmes messages en entrée et en sortie dans les deux solutions.
Dans la solution réelle, cette fonction recevait Num-Oppo, Num-Tira et Num-OK-Tira. Elle affichait directement le résultat des vérifications avec, éventuellement, le message Paiement-Non-Permis ou le Montant-A-Payer ainsi que le Taux-Précpte.
Dans la seconde solution, étant donné la vérification ligne par ligne et la production immédiate du message résultant de cette vérification, la fonction Vérif-Paiement-Permis-Num a pour seul rôle de contrôler si la valeur peut être payée ou non.
- A.2.6 La fonction atomique Choix-Taux-Précpte a en entrée le message Réf-Op-Valide dans la première solution. Dans la seconde, cette fonction nécessite également le message Paiement-Permis.
Il s'agit ici d'un pur choix de conception. Dans la solution réelle, si le choix du taux de précompte avait dû se faire une fois que l'on savait si le paiement était permis ou non, l'exécution de la phase aurait duré plus longtemps. Si j'ai donc décidé de rechercher le taux de précompte sans savoir si on avait à s'en servir (c'est à dire sans savoir si le paiement était permis), c'est par volonté de promouvoir le parallélisme quand cela était possible.

- A.2.7 Une nouvelle fonction est créée : Mémo-Structure-Choisie.
En effet, N-Structure est encodée une seule fois au début de l'opération. Elle peut donc être immédiatement mémorisée. Dans la solution réelle, elle était mémorisée par la fonction Mémorisation-Num-Valides qui mémorisait toute la saisie.
- A.2.8 Disparition de la fonction Validation-Qtés : celle-ci est devenue inutile puisque toute l'opération se fait sur place (étant donné la disponibilité de la base de données).
- A.2.9 Nous allons à présent reprendre une liste des nouvelles fonctions, des nouveaux messages,... (en renommant éventuellement certaines fonctions du fait de la différence de messages en entrée/sortie ou du fait de la modification au niveau du traitement même).

a. Nouvelles fonctions :

- Mémo-Résultat-Vérif :
A pour objectif de mémoriser les Qté/Coupure, les Coupures-Saisies et les résultats successifs des vérifications quand l'encodage est terminé.
Reçoit : Num-Oppo-Ligne, Num-Tira-Ligne, Num-OK-Ligne, Structure-Mémorisée, Qté-Coupure-Mémorisée et Fin-Encodage-Num.
Génère : Num-Oppo, Num-OK et Num-Tira.
- Mémo-Qté-Coupure :
A pour objectif de mémoriser une ligne Qté/Coupure et Coupure-Saisie.
Reçoit : Qté/Coupure et Coupure-Saisie.
Génère : Qté-Coupure-Mémorisée.
- Mémo-Structure-Choisie
A pour objectif de mémoriser le format de numérotation.
Reçoit : N-Structure.
Génère : Structure-Mémorisée.

b. Fonctions modifiées :

- Vérif-Ligne-Oppo remplace Vérification-Oppo.
A pour objectif de vérifier si les numéros de Série-Numéros subissent l'opposition.
Reçoit : Série-Numéro.
Génère : Num-Oppo-Ligne, Sortie-2, Num-OK-Oppo-Ligne.

- Vérif-Ligne-Tira remplace Vérification Tira.
A pour objectif de vérifier si les numéros de Num-OK-Oppo-Ligne sont soumis au tirage ou non.
Reçoit : Num-OK-Oppo-Ligne.
Génère : Num-Tira-Ligne, Sortie-3, Num-OK-Ligne, Sortie-1.
- Vérification-Paiement remplace Vérif-Num-Permis-Num.
A pour objectif de vérifier si le paiement est permis.
Reçoit : Réf-Op-Valide.
Génère : Paiement-Permis et Sortie-5.
- Choisir-Taux-Précpte remplace Choix-Taux-Précpte-Num.
A pour objectif de choisir le taux de précompte.
Reçoit : Réf-Op-Valide et Paiement-Permis.
Génère : Taux-Précpte-Num.

c. Fonctions disparues :

- Validation-Quantités.
- Mémorisation-Num-Valides.

d. Nouveaux messages internes :

- Structure-Mémorisée
comprend une Structure.
- Qté-Coupure-Mémorisée
comprend une Qté/Coupure et une Coupure-Saisie.
- Num-Oppo-Ligne
comprend des numéros (De...à...) de Série-Numéros qui subissent l'opposition.
- Num-OK-Oppo-Ligne
comprend des numéros (De...à...) de Série-Numéros qui ne subissent pas l'opposition.
- Num-Tira-Ligne
comprend les numéros de Série-Numéros qui sont tirés.
- Num-OK-Ligne
comprend les numéros de Série-Numéros qui ne sont ni tirés ni soumis à l'opposition.

e. Nouveau message externe de saisie :

- Fin-Encodage-Num
comprend un booléen Fin-N° = Vrai.

f. Messages internes disparus :

- Num-Saisis-Valides.
- Saisie-Num-Mémorisés.

g. Message externe d'affichage disparu : Erreur-1.

A.3 La phase Enregistrement-Paiement n'est pas modifiée quelles que soient les hypothèses techniques envisagées.

B. Synthèse.

La comparaison met en évidence certaines différences entre les deux solutions importantes à souligner.

La solution moderne encourage le parallélisme au niveau des traitements. Ainsi, pendant que l'utilisateur continue à encoder, ce qui peut être validé indépendamment du reste est validé, éventuellement mémorisé.

Le terme Validation signifie aussi Vérification. L'utilisateur est prévenu d'une erreur avant que l'encodage total ne soit terminé. Il peut dès lors corriger immédiatement.

La deuxième solution est sensiblement meilleure que la solution réelle, notamment par un retour d'informations beaucoup plus fréquent. Cela est flagrant au niveau du traitement des séries de numéros. L'utilisateur encode une série de numéros, elle est immédiatement vérifiée et l'utilisateur est prévenu du résultat de la vérification série par série. Ainsi, l'interaction entre l'utilisateur et l'application est beaucoup plus fréquente.

Nous voyons donc que la deuxième solution présente des avantages certains notamment en permettant la correction immédiate d'erreurs de distraction et la validation au fur et à mesure de l'encodage. L'utilisateur se sent ainsi davantage en interaction avec l'application.

Sous-section 3.2. Choix au niveau de la présentation

Nous allons ici définir le style d'interaction et choisir les objets interactifs les plus appropriés étant donné les contraintes définies par l'utilisateur et par la tâche.

L'accès à la base de données par le guichetier introduit la possibilité de nouvelles fonctions. Ces fonctions permettront de consulter la base de données. Nous appellerons ces fonctions les fonctions de service. Elles se différencient des fonctions de l'application. Elles ne sont pas nécessaires à l'application mais elles offrent des services à l'utilisateur lors de l'exécution des fonctions de l'application.

Ces fonctions doivent également pouvoir être disponibles en dehors de toute exécution d'une fonction d'application. Ainsi, l'utilisateur peut consulter la base de données, informer le client sans être obligé d'entamer l'exécution d'une opération sur titre. Le travail du guichetier est ainsi enrichi. Son travail est plus orienté vers le contact avec le client, il ne se limite plus à l'encodage pur et simple imposé par la solution réelle.

Cependant, lors du choix du style d'interaction et des objets interactifs abstraits, il ne faut pas oublier qu'il y a toujours ce travail d'encodage à minimiser et à faciliter au maximum. Nous aurons donc toujours besoin de formulaires.

Le choix des objets interactifs est fait en fonction des règles ergonomiques et en fonction de l'arbre de sélection d'objets interactifs (Vanderdonckt, 91).

Nous aurons un menu initial sous forme de liste reprenant les divers types d'opérations possibles en agence. Un item de ce menu sera intitulé "Opérations sur titre". La sélection de cet item aura pour effet l'affichage d'une liste de sélection présentant les opérations sur titre possibles et un item de consultation de la base de données. Cet item permettrait la consultation de la base de données d'après différents critères. Ainsi, l'utilisateur aura la possibilité de consulter la base de données en dehors de l'exécution de toute opération.

En général, notre choix en ce qui concerne les objets interactifs sera guidé par ce qui suit :

- Lorsque les items à encoder ont un domaine de valeur limité à deux ou trois éléments différents, nous emploierons les boutons-radio.
- Nous utiliserons la boîte à option lorsque le nombre d'items est supérieur à trois et inférieur à huit (sept étant le chiffre magique) et lorsque cette liste de valeurs ne peut être étendue.
- Nous emploierons une liste de combinaison déroulante lorsque l'item peut être encodé à partir du clavier ou par sélection dans la liste à afficher en cliquant sur la flèche positionnée à droite du champ

d'édition.

- Nous utiliserons les boutons pour confirmer la sélection, l'encodage d'un formulaire, la lecture d'un message d'erreur ...,et pour annuler et revenir au micro-dialogue précédent. Le bouton étiqueté "OK" sera la valeur par défaut : la touche RETURN a le même effet que le clic-souris sur le bouton "OK".
- Nous aurons une barre de menu présentant les fonctions disponibles à partir du micro-dialogue déclenché telles que Défaire, Consultation de la base de données générale et une ou plusieurs autres fonctions de service. Au dessus de cette barre de menu, nous aurons une zone de titre présentant le micro-dialogue courant.
- Nous utiliserons des listes de sélection pour l'affichage d'éléments à sélectionner lorsque le nombre de ces éléments sera supérieur à sept.

Sous-section 3.3. Choix au niveau de la macro-conversation

Nous aurons d'une part les fonctions de l'application et les fonctions de service d'autre part. Identifions-les :

Fonctions de l'application :

1. Encoder le Client
2. Encoder le Numéro-Valeur et son Code
3. Encoder l'identification de la valeur quand le Numéro-Valeur $\neq 0$
4. Encoder l'identification de la valeur quand le Numéro-Valeur = 0
5. Encoder les quantités et les coupures
6. Encoder les numéros
7. Encoder le paiement

Fonctions de services :

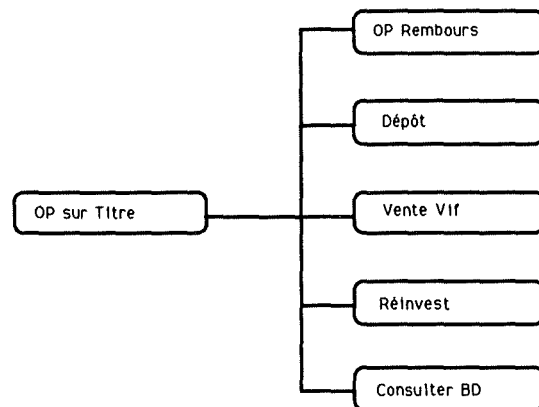
- Consulter Client
- Consulter l'Emetteur
- Consulter la valeur mobilière (Vamo)
- Consulter les formats
- Consulter Coupures Emises
- Consulter le taux de précompte
- Consulter le montant pour un coupon
- Consulter le montant de remboursement pour le titre
- Consulter le taux de remboursement
- Consulter les primes éventuelles
- Consulter Devise
- Consulter le Cours
- Consulter les Oppositions
- Consulter les Tirages

Chacune de ces fonctions représente une source d'information. Elles ont été choisies en fonction de l'intérêt qu'elles apportent pour promouvoir le service au client.

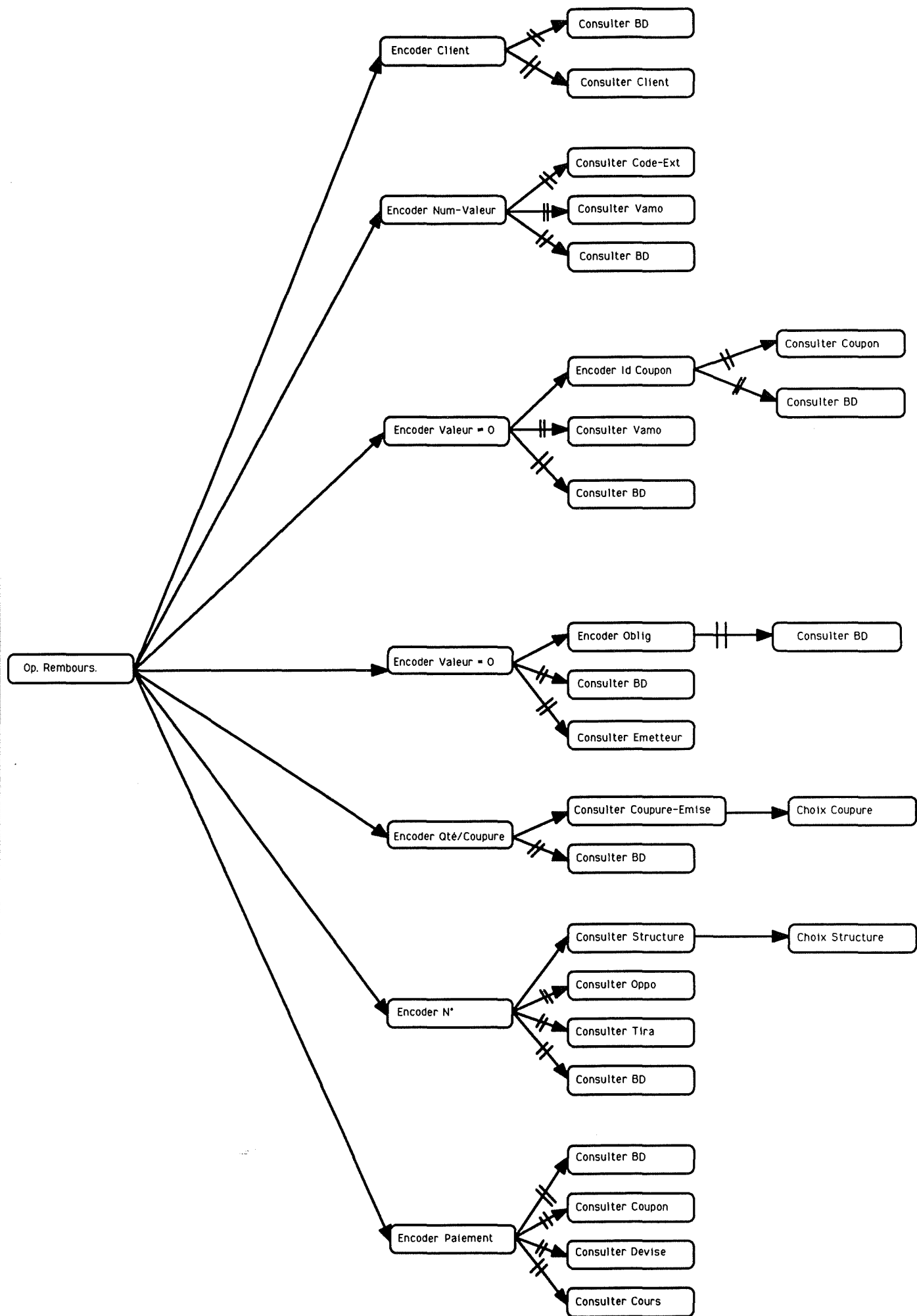
Toutes ces fonctions de services pourront être accessibles à partir de tout micro-dialogue par un item "Consultation BD" présent dans la barre de menu.

Cependant, les fonctions pertinentes à l'exécution du micro-dialogue en cours seront plus accessibles que les autres car directement présentées par un item dans la barre de menu. Leur sélection, dont la fréquence est a priori plus élevée que pour les autres critères de sélection, est ainsi facilitée.

Voici le graphe d'enchaînement des micro-dialogues présents dans la barre de menu initiale :



Nous vous présentons à la page suivante l'enchaînement des micro-dialogues de l'opération de remboursement de titres. La définition du formalisme peut être consultée en Section 1 de ce chapitre.



Nous voyons apparaître dans ce schéma le déclenchement modeless. Ce type d'enchaînement a été choisi pour tout micro-dialogue non nécessaire à l'application. Il s'agit des micro-dialogues de consultation de la base de données. En effet, il peut être utile au guichetier de consulter la base de données au cours de l'exécution d'une fonction de l'application et de garder le résultat de cette consultation sous les yeux en poursuivant l'encodage.

Le déclenchement modal a été conservé pour les fonctions de l'application et les fonctions de service nécessaires à la bonne exécution de fonctions de l'application (Ex : Consulter Structure).

Sous-Section 3.4 Choix au niveau de la conversation - Usage de l'arbre de sélection

1) A l'aide des règles ergonomiques.

Toute boîte de dialogue aura une zone de titre l'identifiant. Chaque boîte de dialogue comportera une barre de menu présentant les items Défaire et Consult. BD. La sélection de l'item Défaire permettra d'annuler l'action faite par l'utilisateur qui précède directement la sélection de l'item Défaire. L'item Consult. BD permettra de faire apparaître une liste de sélection présentant toutes les consultations possibles de la base de données concernant l'opération de remboursement.

Dans cette barre de menu, nous aurons également un item d'Aide représentant l'aide générale au micro-dialogue. Et, comme dans la solution réelle, nous aurons la possibilité de faire apparaître l'aide préventive à l'aide d'un item de la barre de menu Voir Aide Prév. qui, une fois déclenché, change de nom et de rôle pour devenir Fin Aide Prév.

Un ou plusieurs items présents dans la barre de menu permettront de consulter des éléments de la base de données plus pertinents au micro-dialogue en cours. Ces items spécifiques apparaîtront en même temps que la boîte de dialogue à laquelle ils se réfèrent et disparaîtront avec elle.

Chaque item présent dans la barre de menu déclenchera une action lui correspondant. La sélection de ces items pourra se faire à l'aide du clic-souris ou par raccourcis (Ctrl + la lettre soulignée) (choix d'accélérateurs (IBM, 89b) page 50).

Les messages d'aide préventive apparaissent en bas de l'écran à une place prévue à cet effet. Les messages d'erreur sont affichés en superposition et demandent confirmation par clic sur "OK" pour être effacés ((IBM, 89b) page 104). Le message d'aide préventive s'effacera dès que l'action à laquelle il se réfère est terminée.

La longueur maximale ou obligatoire sera représentée par la longueur du cadre dans lequel se fera la saisie, l'encodage aura pour effet de surcharger ce cadre au fur et à mesure de l'encodage ((IBM, 89b) page 93). La confirmation de l'encodage de la valeur d'un item sera implicite si tout le champ doit être complété, sinon elle devra se faire explicitement à l'aide d'ENTER.

A présent, passons en revue une à une les fonctions de l'application et voyons, d'après les types des items et d'après leur domaine de valeur les objets interactifs les plus appropriés pour la saisie des valeurs. Tout choix sera justifié à l'aide de la norme CUA mise au point par IBM (IBM, 89a et 89b) ou à l'aide des règles ergonomiques de Scapin (SCA, 87). Après la présentation de l'interface ainsi obtenue, nous appliquerons l'arbre de sélection d'objets interactifs abstraits aux messages externes de saisie. Nous comparerons ainsi les résultats de ces deux procédures de sélection et mettrons ainsi en évidence les faiblesses éventuelles de l'arbre de sélection.

Les fonctions de services, à déclenchement modeless, seront présentées à la suite des fonctions de l'application qui sont, elles, à déclenchement modal.

Tout choix d'objet interactif sera justifié à l'aide des règles ergonomiques édictées par Scapin (SCA, 87) ou à l'aide de la norme CUA d'IBM (IBM, 89a et 89b).

Chaque boîte de dialogue respecte les règles (IBM, 89b) page 97.

A. Présentation des micro-dialogues à déclenchement modal.

A.1. Identification du client et de la valeur.

Cette boîte de dialogue permet la saisie de l'identification du client et de la valeur nécessaire à toute opération de remboursement.

Les items sont regroupés par type d'identification (identification du client, de la valeur et du coupon) en fonction des règles de (SCA, 87) page 82 et d'(IBM, 89b) page 92.

Le nom peut être encodé à partir du champ d'édition ou par sélection de "Anonyme" à l'aide de la boîte à cocher ((IBM, 89b) pages 78 et 79). Cette sélection a pour effet d'écrire "Anonyme" dans le champ de saisie de Nom.

Le Numéro de Compte, le Numéro d'identité et la Date d'échéance du coupon sont encodés à l'aide de champs d'édition avec un profil correspondant à leur type (guidage (SCA, 87) page 14).

Le Code a une valeur par défaut ("0"). Une autre valeur peut être sélectionnée dans la liste de sélection déroulante associée. Cet objet interactif a été choisi car la liste est non extensible et la place limitée ((IBM, 89b) page 85). De plus, la fréquence de sélection est faible, il n'est donc pas nécessaire d'afficher la liste en permanence.

Le type de la valeur est encodé à l'aide de trois boutons-radio ((IBM, 89b) page 78) ces trois valeurs étant exclusives.

Le nombre de coupons est encodé à partir d'un champ d'édition uni-linéaire avec libellé ((IBM, 89b) page 92).

Les autres items ont une valeur inconnue à priori, c'est pourquoi on leur attribue un champ d'édition uni-linéaire ((IBM, 89b) page 81).

Les champs de saisie des trois items d'identification du coupon sont affichés en grisé lorsque la valeur ne possède pas de coupon ((IBM, 89b) page 89).

A.2. Informations supplémentaires sur la valeur.

Les items sont regroupés en fonction de leur degré d'application à une valeur (général ou spécifique à l'obligation) à l'aide de boîtes de regroupement ((SCA, 87) page 82) et ((IBM, 89b) page 92).

Le type du titre est encodé à l'aide de boutons-radio ((IBM, 89b) page 78).

Le champ de saisie de la devise présente la devise belge par défaut ((SCA, 87) page 25). D'autres devises peuvent être sélectionnées à partir de la liste de combinaison déroulante ((IBM, 89b) pages 84 et 85), cette liste étant extensible.

Les autres items ont une valeur inconnue à priori, il leur est attribué un champ d'édition uni-linéaire ((IBM, 89b) page 81).

A.3. Choix du format des numéros.

Le format est encodé par sélection dans une liste non déroulante car la fréquence de sélection est non négligeable (pour toute opération de remboursement dont la codification du numéro-valeur est reconnue) et la place non limitée ((IBM, 89b) page 79).

La phrase "Choisissez un seul format" correspond à une aide à l'encodage permanente, la place étant non limitée ((SCA, 87) page 81).

A.4. Choix des coupures.

Le choix se fait dans une liste de sélection non déroulante car la fréquence de sélection est non négligeable et la place non limitée ((IBM, 89b) page 79).

La phrase "Choisissez une ou plusieurs coupures" a la même fonction que la phrase dans la boîte de dialogue précédente.

A.5. Encodage des quantités par coupure.

La quantité est encodée à l'aide d'un champ d'édition avec libellé indiquant l'intervalle de valeurs admises ((IBM, 89b) page 92).

A.5.1. Introduction des numéros.

Cette boîte s'affiche dès qu'une quantité encodée est confirmée.

Un bouton permet d'aller à la page de numéros précédente s'il y en a une, sinon elle est affichée en grisé. Un autre bouton permet l'affichage de la page suivante de numéros (guidage (SCA, 87) page 81).

L'encodage des numéros se fait à l'aide de champs d'édition ayant le format choisi antérieurement.

Quand l'utilisateur clique sur le bouton "OK" (la saisie des numéros est terminée pour cette coupure), la boîte se referme. Le curseur se met sur la quantité suivante à saisir, s'il y en a encore une à saisir.

En montant et en descendant le curseur au travers des champs d'édition des quantités par coupure, la première page de numéros saisis pour la coupure sur laquelle se trouve le curseur s'affiche en superposition.

Lorsqu'une ligne de numéros subit l'opposition ou le tirage, l'utilisateur en est immédiatement informé par un message apparaissant dans la zone de message.

A.6. Paiement.

La saisie du type de paiement se fait à l'aide de boutons-radio étant donné le nombre restreint de choix possibles (deux) ((IBM, 89b) page 78).

La maquette de cette interface est présentée dans les pages suivantes.

Identification du client et de la valeur	
Défaire Consult. BD Voir Aide Prév. Aide Consult. Client Consult. Vamo Consult. Code externe	
Identification du client. Nom : ☐ Anonyme Prénom : Numéro-Compte : - - Numéro-Identité : - -	Identification de la valeur. Numéro-Valeur : Code : 0 ▼ Type de la valeur : ☐ manteau ☒ coupon ☐ manteau + coupon
Identification du coupon. Nombre de Coupons : 1 à 999 N° du Coupon : Date d'échéance : - -	
O K Annuler	

Identification du client et de la valeur.

Informations supplémentaires sur la valeur	
Défaire Consult. BD Voir Aide Prév. Aide Consult. Emetteur	
Informations générales. Type du titre : ☒ Action ☐ Obligation Devise : BEF ▼ Libellé : 	Informations sur l'obligation Date d'émission : Date d'échéance : Taux de remboursement : Prime d'émission : Prime d'échéance : Valeur nominale totale :
O K Annuler	

Saisie d'informations supplémentaires pour une action.

Informations supplémentaires sur la valeur	
Défaire Consult. BD Voir Aide Prév. Aide Consult. Emetteur	
Informations générales. Type du titre : ☐ Action ☒ Obligation Devise : BEF ▼ Libellé : 	Informations sur l'obligation Date d'émission : Date d'échéance : Taux de remboursement : Prime d'émission : Prime d'échéance : Valeur nominale totale :
O K	Annuler

Saisie d'informations supplémentaires pour une obligation.

Choix du format des n°			
<u>D</u> éfaire	<u>C</u> onsult. BD	<u>V</u> oir Aide	<u>P</u> rév
<u>A</u> ide			
Choisissez un seul format :			
v 2-5-1 3-6-2 1-4-3 3-6		↑ ↓	
OK		Annuler	

Choix du format des numéros.

Choix des coupures	
<u>D</u> éfaire	<u>C</u> onsult. BD
<u>V</u> oir Aide Prév	<u>A</u> ide
Choisissez une ou plusieurs coupures :	
5 000	
10 000	
☒ 20 000	
50 000	
100 000	
150 000	
OK	Annuler

Choix des coupures.

Quantités par coupure	
<u>D</u> éfaire	<u>C</u> onsult. BD
<u>V</u> oir Aide	<u>P</u> rév
<u>A</u> ide	
	Coupures de
	Coupures de
	Coupures de
O K	
Annuler	

Encodage des quantités par coupure.

Quantités par coupure																					
<u>D</u> éfaire	<u>C</u> onsult. BD	<u>V</u> oir Aide Prév	<u>A</u> ide																		
 	Coupures de _____ Consult. <u>O</u> pposition Consult. <u>T</u> irage																				
Introduisez les numéros : <table style="width: 100%; border: none;"> <tr> <td style="width: 5%;">De</td> <td style="width: 15%;"> </td> <td style="width: 5%;"> </td> <td style="width: 5%;">à</td> <td style="width: 15%;"> </td> <td style="width: 5%;"> </td> </tr> <tr> <td>De</td> <td> </td> <td> </td> <td>à</td> <td> </td> <td> </td> </tr> <tr> <td>De</td> <td> </td> <td> </td> <td>à</td> <td> </td> <td> </td> </tr> </table> <--Page préc. Page suiv.--> O K Annuler				De			à			De			à			De			à		
De			à																		
De			à																		
De			à																		

Introduction des numéros.

Paie ^{ment}		
<u>D</u> éfaire	<u>C</u> onsult. BD	<u>V</u> oir Aide Prév. <u>A</u> ide
Consult. <u>D</u> e <u>v</u> ise	Consult. <u>C</u> ours	Consult. <u>C</u> ou <u>p</u> on
Montant à payer = _____ Taux de précompte = _____ Type de paiement : ☒ Cash ☐ Sur Compte _____ O K Annuler		

Saisie du type de paiement.

B. Présentation des micro-dialogues à déclenchement modeless.

Chaque micro-dialogue consiste en une fonction de service. Ces fonctions de service sont diverses consultations possibles de la base de données. On les appelle fonctions de service car elles ne sont pas nécessaires à l'application mais lui apportent éventuellement une aide. L'objectif de ces fonctions est d'informer le guichetier qui pourra à son tour informer le client. Ces fonctions lui permettent ainsi d'améliorer le service au client, de mettre l'accent sur le dialogue avec le client. Plus et mieux informé, le guichetier aura le sentiment de faire un travail plus intéressant que l'encodage seul.

Reprenons les consultations intéressantes à fournir à l'utilisateur. Celles-ci sont fournies à titre indicatif, l'idéal étant que l'utilisateur les définisse lui-même :

Consulter Client
 Emetteur
 Vamo
 Formats
 Coupures émises
 Taux de précompte
 Montant Coupon
 Montant de remboursement-Titre
 Taux de remboursement
 Primes
 Devise
 Cours
 Opposition
 Tirage

Pour chaque consultation, il sera demandé à l'utilisateur d'introduire la clé d'accès au fichier correspondant. C'est ainsi que le Client pourra être consulté à partir de ses nom et prénom s'il est client habituel sinon à partir de son numéro de compte. Pour consulter l'émetteur, la valeur mobilière, les coupures émises, les primes, les oppositions et le tirage, il faudra avoir pu ou pouvoir reconnaître la codification du N°-Valeur. Les taux de précompte, montant du coupon, montant du titre et taux de remboursement pourront être consultés à partir du numéro de valeur ou du code de l'opération. La devise peut être consultée à partir du numéro de valeur. La liste complète des devises est également accessible. Le cours sera consulté à partir de la devise et du marché désiré par exemple.

Les diverses informations obtenues resteront affichées à l'écran tant que l'utilisateur n'aura pas demandé leur effacement. Il pourra passer du micro-dialogue application au micro-dialogue service et vice-versa. Il pourra ainsi être éventuellement guidé au cours de l'encodage.

2) Application de l'arbre de sélection des O.I.A. aux messages externes.

Après avoir conçu cette interface moderne et justifié nos choix à l'aide des normes C.U.A. ou à l'aide des règles ergonomiques de Scapin, nous allons reprendre les messages externes de saisie et leur fournir un objet interactif abstrait à l'aide de l'arbre de sélection des O.I.A. mis au point par Jean Vanderdonckt (Vanderdonckt, 91).

Nous considérons l'expérience de l'utilisateur au niveau intermédiaire pour ne pas prendre un cas extrême (inexpérience ou maîtrise).

Nom-Cli (Chaîne de 1 à 20 car.)

- a) - Type de la valeur = alphabétique.
 - Valeurs inconnues.
 - ==> O.I.A. recommandé : champs d'édition uni-linéaire.
- b) - Valeur connue (anonyme).
 - Liste non extensible.
 - nombre de valeurs à choisir = 1.
 - liste ordonnée (valeur unique dans la liste).
 - espace d'affichage non limité.
 - ==> O.I.A. recommandé : liste de sélection.

Pré-Cli (Chaîne de 1 à 12 car.)

Idem Nom-Cli a).

Num-Cpte (Chaîne numérique de 12 car.)

- Type de la valeur = numérique.
- Nombre de valeurs à choisir = 1.
- Valeurs non continues.
- Nombre de valeurs possibles ≥ 50 .
- Valeurs inconnues.
- ==> O.I.A. recommandé : champ d'édition uni-linéaire.

Num-Id (Chaîne numérique de 12 car.)

Idem Num-Cpte.

Type-Val (Chaîne de 2 car. , 3 valeurs possibles)

- Type de la valeur = alphabétique ou alphanumérique.
- Valeurs connues.
- Liste non extensible.
- Nombre de valeurs à choisir = 1.
- Liste non ordonnée.
- Nombre de valeurs possibles = 3.
- ==> O.I.A. recommandé : boutons-radios.

Num-Valeur (Chaîne de 14 car.)

Idem Num-Cpte.

Code (Chaîne de 1 caractère, 7 valeurs possibles)

- Type de la valeur = alphanumérique.
 - Valeurs connues.
 - Liste extensible.
 - Place d'affichage limitée.
- ==> O.I.A. recommandé : boîte de combinaison déroulante.

Num-Coupon (Entier à 2 car.)

- Type de la valeur = entier.
 - Nombre de valeurs à choisir = 1.
 - Valeurs non continues.
 - Nombre de valeurs possibles ≤ 50 .
 - Longueur du champs d'édition < 40 .
- ==> O.I.A. recommandé : liste de sélection.

Date-Ech-Coupon (Date)

- Type de la valeur = date.
 - Place limitée et degré d'expérience ≥ 5 .
- ==> O.I.A. recommandé : champ d'édition profilé.

Val-Nom-Tot (réel 12,3)

- Nombre de valeurs à choisir = 1.
 - Non continu.
 - Nombre de valeurs possibles ≥ 50 .
 - Valeurs inconnues.
- ==> O.I.A. recommandé : Champ d'édition uni-linéaire.

Nb-Coupon (entier 3)

- Nombre de valeurs à choisir = 1.
 - Intervalle continu.
 - Nombre de valeurs possibles compris entre 1 et 999.
 - Degré d'expérience < 5 .
- ==> O.I.A. recommandé : bouton de variation.

Type-Titre (Chaîne de 1 car. , 2 valeurs possibles)

- Type de la valeur = alphabétique.
 - Valeurs connues.
 - Liste non extensible.
 - Nombre de valeurs à choisir = 1.
 - Liste non ordonnée.
 - Nombre de valeurs possibles = 2.
- ==> O.I.A. recommandé : bouton-radio.

Devise (Chaîne de 3 caractères)

- Type de la valeur = alphabétique.
 - Valeurs connues.
 - Liste extensible.
 - Place d'affichage limitée.
- ==> O.I.A. recommandé : boîte de combinaison déroulante.

Date-Em (Date)

- Type de la valeur = date.
 - Place limitée et degré d'expérience ≥ 5 .
- ==> O.I.A. recommandé : champ d'édition uni-linéaire profilé.

Date-Ech (Date)

Idem Date-Em.

Taux (4 car.)

- Type de la valeur = numérique.
 - Nombre de valeurs à choisir = 1.
 - Intervalle non continu.
 - Nombre de valeurs possibles ≥ 50 .
 - Valeurs inconnues.
- ==> O.I.A. recommandé : champ d'édition uni-linéaire.

Libellé (Chaîne de 30 car.)

Idem Pré-Cli.

Prime-Em (4 car.)

Idem Taux.

Prime-Ech (4 car.)

Idem Taux.

Structure (Tableau d'entiers : 10,2)

- Type de la valeur = entier.
 - Nombre de valeurs à choisir = 1.
 - Valeurs connues.
 - Nombre de valeurs possibles compris entre 8 et 99.
 - Degré d'expérience > 5.
- ==> O.I.A. recommandé : champ d'édition + libellé.

Coupure (Réal)

- Type de la valeur = numérique.
 - Nombre de valeurs à choisir > 1.
 - Nombre de valeurs secondaires = 0.
 - Liste extensible.
 - Nombre de valeurs possibles compris entre 4 et 7.
- ==> O.I.A. recommandé : boîte à cocher + boîte de regroupement.

Qté/Coupure (3 car. entiers)

- Type de la valeur = entier.
 - Nombre de valeurs à choisir = 1.
 - Intervalle connu et continu.
 - Nombre de valeurs possibles compris entre 1 et 999.
 - Degré d'expérience > 5.
- ==> O.I.A. recommandé : champ d'édition uni-linéaire + libellé.

Série-Numéros (Chaîne de X car.)

- Type de la valeur = numérique.
 - Nombre de valeurs à choisir = 1.
 - Intervalle non continu.
 - Nombre de valeurs possibles ≥ 50 .
 - valeurs inconnues.
- ==> O.I.A. recommandé : champ d'édition uni-linéaire.

Type-Paiement (4 car. , 2 choix possibles)

- Type de la valeur = alphabétique.
 - Valeurs connues.
 - Liste non extensible.
 - Nombre de valeurs à choisir = 1.
 - Liste non ordonnée.
 - Nombre de valeurs possibles = 2.
- ==> O.I.A. recommandé : bouton-radio.

Critique et propositions.

Nous voyons que l'usage de l'arbre de sélection nous a amenés, en général, à attribuer les mêmes objets interactifs que le choix opéré à l'aide des règles ergonomiques.

Nous pouvons cependant noter deux faiblesses dans l'usage de cet organigramme.

La première concerne les champs d'édition attribués aux numéros de compte, au numéro d'identité et aux numéros de titres. En effet, il pourrait être ajouté un test sur la nécessité ou non d'un profil de numérotation en demandant, par exemple, si les éléments de la valeur peuvent être groupés ou non. Nous obtiendrions alors pour ces messages externes un champs d'édition uni-linéaire profilé comme il est possible d'en obtenir pour les dates.

La seconde faiblesse est le manque de test permettant l'indication ou non d'une valeur par défaut (ex. : pour le code, nous obtenons une boîte de combinaison déroulante sans valeur par défaut). Un test complémentaire possible porterait sur l'existence d'une valeur plus fréquente que les autres. Si une telle valeur existait, elle deviendrait la valeur par défaut, sinon aucune valeur par défaut ne serait générée.

On peut également relever la difficulté rencontrée pour attribuer un objet interactif abstrait adéquat pour la sélection de "Anonyme". Nous obtenons comme objet la liste de sélection alors que la valeur est unique. Pour pallier à cette difficulté, nous pourrions placer "Anonyme" en tant que valeur par défaut mais cela n'est pas recommandé car le nombre d'opérations avec un client anonyme doit être limité.

Au-delà ces petites faiblesses, l'arbre de sélection est aisé à utiliser dès que certaines abréviations sont mémorisées. Une légende en bas du graphe faciliterait et encouragerait les premières lectures.

Il serait aussi très intéressant dans le cadre du test de cet arbre de sélection de comparer les objets interactifs abstraits obtenus en prenant les deux extrêmes de niveau d'expérience de l'utilisateur (maître et débutant).

Section 4. Comparaison des deux interfaces

Nous allons comparer les deux interfaces obtenues. La comparaison est basée sur les grandes différences constatées entre elles. Nous avons d'abord constatés que certains usages des touches fonction de l'interface réelle tentent d'imiter l'usage d'objets interactifs évolués. Nous nous sommes ensuite rendus compte que l'image même de l'objet interactif informe sur la façon de procéder pour la sélection, l'encodage, etc ... Nous avons alors remarqué que la deuxième interface, grâce à l'accès à une base de données locale, permet d'améliorer le travail du guichetier. Cet accès présente également l'avantage de permettre la validation immédiate et une interaction plus fréquente entre l'utilisateur et son ordinateur. Nous comparerons enfin les deux interfaces en envisageant la limitation des erreurs et la prise en compte de l'expérience de l'utilisateur. Nous terminons par une synthèse de cette comparaison.

1. Des touches fonctions imitant le comportement d'objets interactifs.

Dans la première interface, la ligne des étiquettes des touches fonction changeait selon l'item sur lequel se trouvait le curseur. Ce moyen permettait de pallier au manque de diversité des objets interactifs. Par exemple, le Type de Paiement est saisi à l'aide de deux touches fonction étiquetées respectivement CASH et COMPTE. L'utilisateur doit utiliser une de ces deux touches pour encoder le type de paiement. Dans la deuxième interface, cette valeur est saisie à l'aide de deux boutons-radio. Dans la première interface, l'usage des touches fonction "imite" l'utilisation de boutons-radio.

L'utilisation des touches fonction pose cependant un problème. Etant donné que ces touches sont spécifiques à chaque item, l'utilisateur (malgré des touches réservées aux fonctions essentielles) mettra du temps pour connaître les fonctions disponibles pour chaque item et la manière de procéder pour encoder.

Dans le cas des touches fonction "imitant" les boutons-radio, il n'apparaît pas clairement à la lecture des étiquettes de ces touches que l'encodage de l'item ne peut se faire qu'à l'aide de celles-ci. L'utilisateur ne s'en apercevra qu'à posteriori après avoir essayé d'encoder l'item à l'aide du clavier et lorsque son erreur lui sera signifiée via un avertissement sonore ou un message. Cet exemple montre la nécessité des messages d'aide préventive pour ce type d'interface. L'objet interactif "Bouton-radio" est beaucoup plus explicite quant à la manière d'opérer une sélection. La valeur par défaut éventuelle ou la dernière sélection opérée laisse un rond noir dans le bouton-radio. Si l'utilisateur veut changer cette valeur, il comprendra immédiatement qu'il doit déplacer ce rond noir.

2. L'objet interactif prévient l'utilisateur.

Observons le cas de l'item Code.

Dans la première interface, avant que le curseur n'arrive au champ de saisie de Code, l'utilisateur voit un champ d'édition uni-linéaire muni d'une valeur par défaut.

A priori, il peut se demander comment il procèdera pour encoder une valeur autre que la valeur par défaut. Une fois que le curseur se trouve sur le champ de saisie, une touche fonction s'étiquette "Liste". L'utilisateur voit alors qu'une liste est associée à cet item mais il ne peut savoir ce que contient cette liste. Contient-elle d'autres codes abrégés, d'autres codes avec leur dénomination ou avec un exemple ? C'est ainsi qu'il apprend que d'autres codes peuvent être consultés mais pourra-t-il sélectionner dans la liste ou s'agit-il simplement d'une consultation ? Autant de questions auxquelles l'utilisateur ne peut répondre en voyant le champ d'édition et la touche fonction étiquetée "Liste".

La deuxième interface affiche quant à elle un champ d'édition uni-linéaire avec une valeur par défaut et, à sa droite, une flèche pointant vers le bas. Ici, l'image indique clairement que d'autres valeurs sont possibles et que des valeurs peuvent être ajoutées (puisque'il s'agit de l'image d'une boîte de combinaison déroulante). L'objet interactif indique donc lui-même plusieurs caractéristiques propres à la valeur à saisir (ex. : existence d'autres valeurs, possibilité de sélectionner une autre valeur). Il suffit d'un coup d'oeil sur la boîte de dialogue pour que l'utilisateur comprenne comment procéder pour opérer la sélection de chaque item par la connaissance qu'il a des objets interactifs.

3. Enrichissement et considération du travail du guichetier.

L'accès à la base de données locale présente de nombreux avantages.

Le premier consiste en la disponibilité au guichet d'informations pertinentes pour le bon déroulement de l'opération en cours. Cet accès permet à l'utilisateur de s'informer sur le client (ex. : s'informer sur le client s'il est inconnu), sur la possibilité d'opposition, de tirage, ...

Par ailleurs, la base de données permet d'informer le client. En effet, le guichetier a à sa disposition une source d'informations importante lui permettant de conseiller ou de prévenir le client pour toute valeur.

Le guichetier n'est plus simplement opérateur. Il devient, par l'interface qu'il manipule, source de conseils et d'informations pour le client. Dès lors, le client ne verra plus le guichetier comme un encodeur mais comme un interlocuteur responsable et efficace ayant les possibilités de le renseigner. De son côté, le guichetier ne se verra plus incapable d'apporter des renseignements. Il ne se sentira plus étranger (comme il l'est dans la première interface) à l'opération. Au contraire, il maîtrise celle-ci par la possibilité d'accéder aux informations permettant de la lui expliquer. Ainsi, lorsque le montant calculé lui paraît étrange, il peut consulter les divers éléments de ce calcul et le vérifier lui-même.

Nous voyons donc que la disponibilité au guichet d'une base de données constitue un grand avantage pour le service à la clientèle et pour le travail du guichetier qui s'en trouve considérablement enrichi.

4. Validation immédiate.

Un deuxième avantage apporté par l'accès à la base de données locale est la possibilité de validation immédiate des valeurs encodées.

Pour toutes les petites erreurs détectables, l'utilisateur sera prévenu immédiatement après la confirmation (implicite ou explicite) de leur encodage. Il pourra ainsi apporter immédiatement la correction nécessaire. Ceci constitue un avantage par rapport à la première solution. Dans celle-ci, tout devait être saisi avant la vérification et tous les messages d'erreur étaient renvoyés à la suite les uns des autres. Cela engendrait une surcharge de la mémoire à court terme de l'utilisateur qui recevait toutes ces informations en une fois.

Cette validation immédiate est également appréciable au niveau de la vérification des oppositions et du tirage. Dans la première interface, tous les numéros devaient être encodés avant d'être vérifiés. Le résultat de cette vérification était l'affichage des numéros regroupés par titres frappés d'opposition, par titres soumis au tirage et par titres "indemnes". Les numéros étaient donc réaffichés, ils permettaient au guichetier de trier les titres selon le résultat de leur vérification.

La deuxième solution a l'avantage de donner le résultat de la vérification série par série. Le guichetier trie les titres au fur et à mesure de l'encodage des numéros. La charge de la mémoire à court terme est nettement réduite et la manipulation des titres unique. Le guichetier gagne en temps d'exécution d'une opération et en fréquence du dialogue homme-machine plus élevée. L'ordinateur travaille au même rythme que lui, série de numéros par série de numéros.

La base de données locale permet ainsi des retours d'informations beaucoup plus fréquents. L'utilisateur est beaucoup plus rapidement informé des erreurs ou invalidités. Il sait, avant de commencer l'encodage, que ses erreurs seront détectées une à une et qu'il pourra les corriger avant de continuer la saisie. La première interface ne permet pas cette détection immédiate et l'utilisateur risque d'éprouver un sentiment de crainte en encodant les différents items car toute erreur détectée redemandera, par après, l'encodage entier.

5. Limitation des erreurs.

Nous pouvons apprécier dans la première interface la présentation à l'utilisateur des coupures émises pour la valeur. Ainsi, l'utilisateur n'a plus qu'à saisir celles qui lui conviennent. Cette solution limite les erreurs possibles au niveau de l'encodage des coupures. Chaque coupure choisie est une coupure existante pour la valeur. Seule peut subsister l'erreur lors du choix.

De même, seul(s) le(s) format(s) de numérotation possible(s) pour la valeur est (sont) présenté(s) à l'utilisateur. Une erreur peut avoir lieu lors du choix du format lorsque plusieurs sont affichés. Le risque d'erreur est néanmoins limité étant donné que le choix ne doit pas se faire parmi tous les formats possibles existants.

6. Prise en compte de l'expérience de l'utilisateur - Guidage.

La première interface propose la possibilité d'affichage de messages d'aide préventive pour l'utilisateur novice. L'utilisateur plus expérimenté pourra supprimer l'affichage de cette aide. Cette fonction étant attribuée à la touche fonction F8.

Dans la deuxième interface, trois niveaux d'expérience peuvent être distingués. L'utilisateur novice pourra demander l'affichage de l'aide préventive grâce au bouton "Voir Aide Prév" de la barre de menu. L'utilisateur intermédiaire pourra demander que l'aide préventive ne lui soit plus affichée. L'utilisateur maître pourra user des accélérateurs, c'est à dire des lettres soulignées accompagnées de la touche "Control" (CTRL) et des valeurs par défaut des boutons (Presser RETURN au lieu de cliquer sur le bouton "OK" par exemple).

Il est à noter que des raccourcis pourraient être prévus pour accélérer le passage au travers des menus.

7. Nombre de boîtes de dialogue.

En usant dans l'interface moderne d'objets interactifs adaptés aux messages externes, à leur type et à leur domaine de valeurs, la place nécessaire pour la saisie est moindre. En effet, l'usage de boutons-radio, par exemple, prend moins de place que l'usage de champ d'édition.

C'est ainsi que les saisies Client, Valeur et Coupon ont pu être groupées en une seule boîte de dialogue. Cette boîte de dialogue saisit les éléments nécessaires à la suite de l'opération. Ce regroupement permet à l'utilisateur de mieux modéliser la tâche. Toutes les informations nécessaires sont rassemblées. Cela nous semble plus clair que plusieurs boîtes de dialogue se succédant, le nombre de boîtes ne devant de toute façon pas excéder le chiffre magique (7 ± 2).

De même, il vaut mieux limiter les boîtes se superposant aux autres et qui s'affichent selon la valeur attribuée à un item. Dans l'interface réelle, nous avons une boîte d'identification du coupon qui se superpose à l'identification de la valeur lorsque le type de celle-ci contient un coupon. Nous pourrions éventuellement la placer avec l'identification de la valeur et elle serait affichée en grisé tant qu'il n'y aurait pas de coupon présent dans le type de la valeur. Rien ne sert d'avoir une boîte de dialogue par type de saisie. De plus, la succession d'affichage et effacement de boîtes de dialogue peut s'avérer intempestive et fatigante.

8. Synthèse de la comparaison des interfaces.

Les divers points de comparaison abordés démontrent une nette préférence pour l'interface moderne.

Nous avons pu mettre en évidence l'importance du choix des objets interactifs étant donné que leur rôle n'est pas uniquement de faciliter la saisie mais également d'informer l'utilisateur quant au type et au domaine de la valeur à saisir.

Nous avons pu remarquer le comportement mimétique des touches fonction dont l'utilisation tente de se rapprocher d'objets interactifs plus évolués.

Un point essentiel est l'amélioration du travail du guichetier. Son travail s'améliore dans deux directions : vers le client et vers la machine. Avec le client, il passe du simple rôle d'encodeur à celui de conseiller. Avec la machine, l'interaction est beaucoup plus fréquente. Elle est aussi plus libre : l'utilisateur a l'initiative et il peut consulter (à volonté) la base de données sans être dirigé par la machine.

Le niveau d'expérience étant pris en compte, le guichetier peut améliorer sa dextérité en se débarrassant au fur et à mesure de son apprentissage du guidage fourni par l'aide préventive.

En ce qui concerne la limitation des erreurs, nous pouvons apprécier l'usage des objets interactifs appropriés (affichage en grisé, O.I.A. spécifiques aux domaines de valeurs limités, etc.), l'usage de la base de données disponible (consultation des formats, des coupures, etc.) et une aide disponible à tout moment.

Chapitre 4. Critique de la démarche

Section 1. Problèmes rencontrés - Causes

Sous-section 1. Etapes préalables à la conception de l'interface

La décomposition de la tâche guide la structuration des traitements. L'avantage, lorsque nous procédons à une analyse abstraite de la tâche, serait, en principe, de ne pas devoir tenir compte d'éléments technologiques. Or, en pratique, nous sommes contraints à procéder à l'analyse de la tâche en fonction de l'environnement technologique. Cette remarque est autant valable pour l'analyse abstraite que pour l'analyse concrète. Nous sommes en effet guidés par les possibilités techniques du matériel et par les contraintes déduites de l'application.

Sous-section 2. Conception de l'interface proprement dite

Les étapes de la conception d'interface (choix au niveau de la présentation, au niveau du micro-dialogue et au niveau de la conversation) sont quant à elles séquentielles.

On peut rencontrer quelques problèmes lors de la définition du macro-dialogue. A cette étape, les unités de dialogue doivent être identifiées et enchaînées. Or, ces enchaînements dépendent fortement du contenu des micro-dialogues (contraintes dues à certaines valeurs). En concevant le macro-dialogue, les informations sont déjà triées, classées et regroupées par micro-dialogue. Or, nous ne savons pas à cette étape quels seront les O.I.A. les plus appropriés pour représenter ces informations. Les choix au niveau du micro-dialogue concernent le choix des O.I.A. (en fonction des caractéristiques des valeurs en utilisant l'arbre de sélection), leur disposition spatiale, leur ordre de présentation, leur regroupement, etc.

Ainsi, l'étape concernant le choix au niveau du macro-dialogue va engendrer des contraintes pour le choix au niveau du micro-dialogue. Le nombre d'informations par micro-dialogue, par exemple, influencera la sélection des O.I.A. Ainsi, le choix pour un message externe d'une liste de sélection déroulante ou non dépendra de la place disponible. Or, cette place disponible dépend du nombre d'informations à représenter dans le micro-dialogue. Ceci montre bien que les choix opérés pour le macro-dialogue orientent les choix lors de l'élaboration de la conversation. Dès lors, de mauvais choix au niveau du macro-dialogue peuvent avoir de sérieuses conséquences pour la conversation. Nous en avons l'illustration au sein de la première interface. Nous disposons d'une boîte de dialogue par type de saisie, chacune contenant peu d'informations, avec entre elles des contraintes qui sont fonction de la valeur de certains items. Ceci est la conséquence d'un trop grand triage au niveau du macro-dialogue.

En ce qui concerne la conversation, le choix est facilité par l'arbre de sélection. Cependant, le résultat obtenu en utilisant cet instrument ne doit pas toujours être accepté tel quel. Il faut parfois faire appel au bon sens (ex. : l'objet interactif "Liste de sélection" proposé pour représenter la valeur "Anonyme" du Nom).

Une fois que les objets interactifs sont définis, le concepteur doit leur appliquer les règles de dimensionnement, de localisation et d'arrangement pour les positionner à l'intérieur de la boîte de dialogue. Or, ces positions sont bien souvent choisies de manière intuitive. Le concepteur a toujours une idée à priori de la manière dont il va disposer ces objets. A posteriori, il considèrera les règles ergonomiques applicables et apportera les corrections éventuelles.

Exemple : Intuitivement, les boutons-radio ont été placés à gauche de leur libellé. Or, la règle de positionnement des libellés ((Vanderdonck, 91) Page 14) préconise de les placer à droite. L'interface doit donc être corrigée pour être conforme à cette règle.

Section 2. Avantages de la démarche

Le grand avantage de cette démarche est l'identification et la succession d'étapes partant du modèle conceptuel et de la décomposition de la tâche pour terminer par la définition complète de l'interface en suivant un raisonnement déductif. Nous disposons ainsi d'une démarche complète où application et interface sont conçues comme un ensemble formant un tout.

Basée sur les informations à saisir et à afficher, et donc sur le dialogue avec l'utilisateur, cette démarche permet de différencier messages externes de saisie et messages externes d'affichage. Ces messages sont la base de départ du raisonnement déductif permettant de dégager les fonctions atomiques de la fonction interactive. Découvertes lors de cette étape, ces fonctions sont définies à l'étape de la statique des traitements. Vient enfin une illustration de leur enchaînement à l'aide du formalisme du graphe d'enchaînement.

Le graphe d'enchaînement peut présenter à lui seul beaucoup d'informations relatives à la fonction interactive dont il représente la dynamique. Ainsi, il permet de représenter les conditions de déclenchement des fonctions atomiques à l'aide des connecteurs logiques (OU, ET, XOU). Ces mêmes connecteurs logiques permettent de montrer les messages possibles générés par toute fonction atomique.

Ces avantages résultent du formalisme employé. Ce dernier permet, à l'aide de matériaux de base, la représentation d'informations à structure complexe. Il faudra qu'il soit encore testé sur d'autres applications pour voir si les différentes formes existantes suffisent ou bien s'il faut l'étendre. En effet, ce formalisme a l'avantage de pouvoir être étendu aisément. C'est ainsi que différentes propositions ont pu être faites lorsqu'une contrainte était difficilement exprimable (Voir Chap. 3, Sous-sections 2.1, les points D et E).

Un autre avantage du graphe d'enchaînement des fonctions atomiques est la facilité avec laquelle la possibilité de parallélisme est détectable. C'est notamment le cas de la fonction "Choisir-Taux-Précpte" présentée lors de la comparaison des graphes d'enchaînement (Voir Chap. 3, Sous-section 4.1).

Un avantage de la démarche réside également dans la présentation d'étapes séquentielles pour la conception d'interface. Ces étapes vont du général au particulier. Le premier choix concerne l'ensemble de l'interface puisqu'il s'agit du choix du style d'interaction. Le deuxième porte sur l'identification des micro-dialogues et leur enchaînement. Le troisième (le plus précis) concerne la représentation de chaque information.

L'enchaînement du macro-dialogue est représenté à l'aide d'un formalisme qui lui est propre. Il a l'avantage de permettre la représentation des deux types de déclenchement possibles (modal et modeless). Il peut de même représenter la dépendance ou l'indépendance des micro-dialogues au travers des enchaînements des boîtes de dialogue.

Chapitre 5. Conclusion

Cette démarche se différencie des méthodes traditionnelles de conception d'interface en considérant comme point de départ la décomposition de la tâche associée au modèle conceptuel.

Les méthodes précédentes de conception d'interface portaient, elles, de la définition des fonctions de l'application.

Partant de la définition de la tâche de l'utilisateur, cette démarche présente une toute autre conception de la méthode suivie lors du développement d'application. Elle a l'avantage de dégager les fonctions interactives à partir d'une analyse de la tâche de l'utilisateur et de procéder ensuite de façon déductive afin d'identifier les fonctions atomiques composant ces fonctions interactives.

Deux points intéressants présents dans la démarche sont le graphe d'enchaînement des fonctions atomiques et l'arbre de sélection des objets interactifs abstraits.

Le graphe d'enchaînement des fonctions atomiques a pour autre nom le schéma de la dynamique (BODART-PIGNEUR, 89). Il représente un cas particulier de ce schéma en précisant certaines contraintes. La construction du graphe d'enchaînement des fonctions atomiques présente de nombreux avantages. Il illustre graphiquement le cheminement des messages externes de saisie, reçus par la fonction interactive, au travers des fonctions atomiques jusqu'à la génération des messages externes d'affichage. Ce graphe permet enfin de représenter les contraintes de déclenchement des fonctions à l'aide de connecteurs logiques. Nous apprécions en outre le formalisme de base d'usage aisé et pratique permettant de présenter graphiquement des contraintes de déclenchement parfois très complexes.

L'arbre de sélection des objets interactifs abstraits, conçu à l'aide des règles ergonomiques de sélection des objets interactifs abstraits, a l'avantage d'attribuer les objets interactifs adéquats aux messages externes de façon algorithmique. Il constitue une bonne aide à l'analyse.

Ces deux éléments présents dans la démarche mise au point par Benoît Sacré et Isabelle Provot devront encore être testés à d'autres applications. Ces tests permettront de démontrer le degré de généralité de la démarche ou bien permettront éventuellement la mise en évidence d'éléments qui pourront la compléter en vue de la rendre plus générale encore.

Références bibliographiques.

(BODART-PIGNEUR, 89) F. BODART et Y. PIGNEUR, "Conception assistée des systèmes d'informations, Méthode - Modèles - Outils", MIPS, MASSON, 1989.

(DUM, 88) J. DUMAS, "Designing User Interfaces for Software", Prentice-Hall, Englewood Cliffs (New Jersey), 1988.

(IBM, 89a) IBM System Application Architecture, Common User Access : Advanced Interface Design Guide, Document SC26-4582-0, Boca Raton (Etats-Unis), juin 1989.

(IBM, 89b) IBM System Application Architecture : Basic Interface Design Guide, Document SC-26-4583-0, Cary (Etats-Unis), décembre 1989.

(NORMAN) D.A. NORMAN, "Cognitive Engineering", in D.A. Norman, S.W. Draper.

(PRO, 91) I. PROVOT, "Exemple de conception d'une interface homme-machine : cas d'un hôpital", Rapport interne, Institut d'informatique, FUNDP, Namur, 5 avril 1991.

(SACRE, 91a) B. SACRE et I. PROVOT, "Démarche de conception de l'interface homme-machine", Rapport interne, Institut d'informatique, FUNDP, Namur, 5 avril 1991.

(SACRE, 91b) B. SACRE et I. PROVOT, "Méthodologie de conception d'une application interactive", Rapport interne, Institut d'informatique, FUNDP, Namur, 17 mai 1991.

(SCA, 87) D.L. SCAPIN, "Guide ergonomique de conception des interfaces homme-ordinateur", Institut National de Recherche en Informatique et en Automatique, Rocquencourt, 1987.

(SHN, 87) B. SCHNEIDERMAN, "Designing the user interface : strategies for effective human-computer interaction", Addison-Wesley, Reading (Massachusetts), 1987.

(VANDERDONCKT, 91) J. VANDERDONCKT, "Les règles ergonomiques de sélection d'objets interactifs abstraits", Rapport interne IHM/Règles/1, Institut d'informatique, FUNDP, Namur, 27 juin 1991.

ANNEXE.

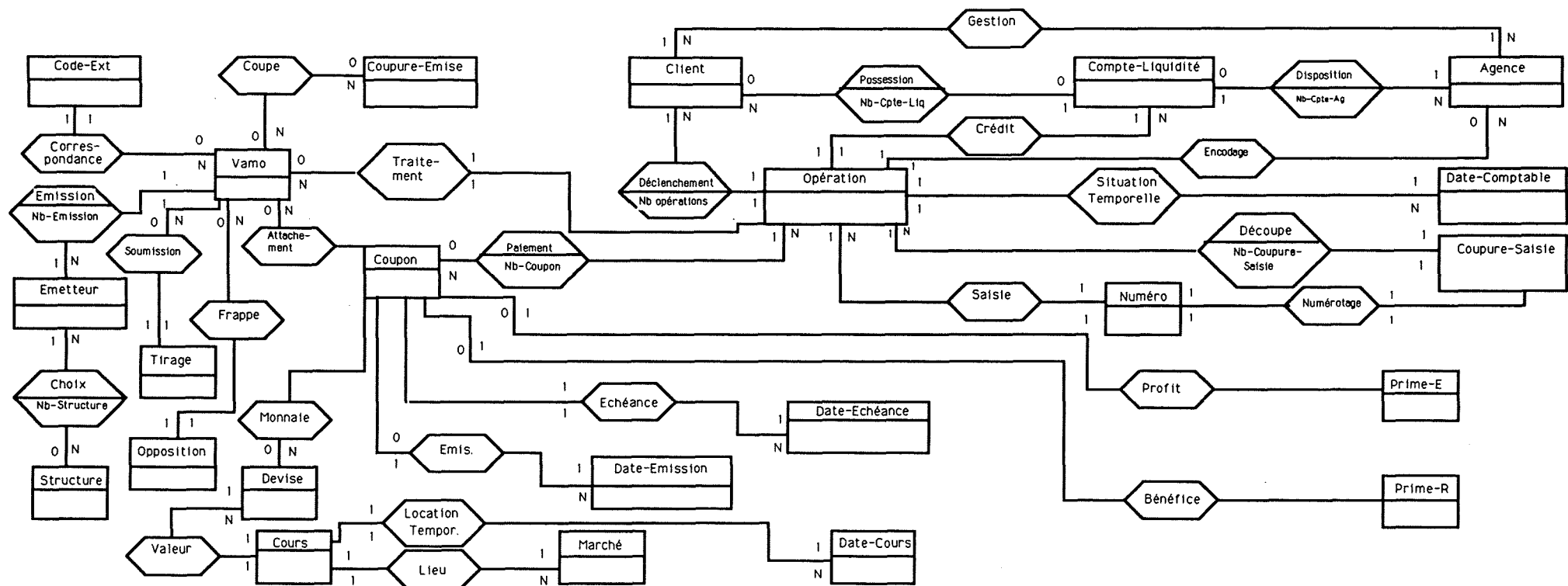


Schéma conceptuel relatif à l'application Remboursement/Paiement Coupons.

1. Définition des types d'Entités.

TYPE D'ENTITE : Opération

DEFINITION : Une opération est toute opération ayant trait à une saisie matière en agence.

DUREE DE VIE : Tant que la "matière" n'est pas transmise à la Banque Nationale et tant que le paiement de l'émetteur n'a pas été effectué.

ATTRIBUTS : Code-Opération
Num-Opération
Date-Opération
Montant_Op
Taux_Précpte_Op
Structure_Op
Tira_Op
Oppo_Op
Paieement_Op

IDENTIFIANT : Code-Opération et Num-Opération

TYPE D'ENTITE : Vamo

DEFINITION : Une valeur mobilière est tout effet repris au Secrétariat des Valeurs Mobilières, action ou obligation, belge ou étranger.

DUREE DE VIE : Tant que la valeur est nécessaire aux diverses applications boursières.

ATTRIBUTS : Type-Codif
Num-ABB
Kndsec (type valeur définie par le SVM)
Euro-Non Euro
Titrmals-Non-mais
Libellé-fr
Libellé-nl
Sigle-fr
Sigle-nl
Date-Ech-Finale
Taux-Int-Ech-Fin
Taux-Int-Progressif
Date-Chang-Taux
Devise-Titre
Nationalité-Titre
Périodicité-Coupon

IDENTIFIANT : Num-ABB

N.B. : Cette Entité a été développée pour des applications antérieures.

TYPE D'ENTITE : Client

DEFINITION : Un client est une personne qui a effectué au moins une fois une opération de Remboursement/ Paiement Coupons.

DUREE DE VIE : Tant que l'opération est présente dans le système d'information.

ATTRIBUTS : Num-Client
Nom-Client
Prénom-Client
Adresse-Client
Num-Carte-Id

IDENTIFIANT : Num-Client

TYPE D'ENTITE : Coupure-Emise

DEFINITION : Une coupure est émise pour une valeur mobilière.

DUREE DE VIE : Tant que la valeur mobilière qui a cette coupure existe dans le système d'information.

ATTRIBUT : Type-Coupure

IDENTIFIANT : Num-ABB et Type-Coupure

TYPE D'ENTITE : Emetteur

DEFINITION : Un émetteur est l'organisme qui a émis une valeur mobilière.

DUREE DE VIE : Tant que les valeurs qu'il a émises sont présentes dans le système d'information.

ATTRIBUTS : Num-Emetteur
Nom-Emetteur
Adresse-Emetteur

IDENTIFIANT : Num-Emetteur

TYPE D'ENTITE : Structure

DEFINITION : Une structure est le format donné par un émetteur à la numérotation des valeurs émises.

DUREE DE VIE : Tant que cette structure est utilisée.

ATTRIBUTS : Num-Structure
Format-Structure

IDENTIFIANT : Num-Structure

TYPE D'ENTITE : Coupon

DEFINITION : Un coupon représente ce qui doit être payé pour une valeur mobilière donnée.

DUREE DE VIE : Tant que la valeur mobilière est présente dans le S.I.

ATTRIBUTS : Num-Coupon.

IDENTIFIANT : Num-Coupon et Num-ABB

TYPE D'ENTITE : Devise

DEFINITION : Une devise est la monnaie employée pour payer le client.

DUREE DE VIE : Tant que la devise est employée pour payer les clients.

ATTRIBUT : Nom-Devise

IDENTIFIANT : Nom-Devise

TYPE D'ENTITE : Cours

DEFINITION : Un cours est la valeur d'une devise à une date particulière sur un marché particulier.

DUREE DE VIE : Tant que l'opération y afférente se trouve dans le S.I.

ATTRIBUTS : Valeur-Cours

IDENTIFIANT : Valeur-Cours et Nom-Devise et Nom-Marché et Date-cours.

TYPE D'ENTITE : Date-Cours

DEFINITION : Une Date-Cours est la date à laquelle une devise particulière prend une certaine valeur sur un certain marché.

DUREE DE VIE : Aussi longtemps que l'opération y afférente doit être gardée dans le S.I..

ATTRIBUTS : Année-Cours
Mois-Cours
Jour-Cours

IDENTIFIANT : Année-Cours et Mois-Cours et Jour-Cours

TYPE D'ENTITE : Marché

DEFINITION : Un marché est un endroit où se fixe le cours quotidien des devises.

DUREE DE VIE : Tant que l'opération y afférente est présente dans le S.I..

ATTRIBUTS : Nom-Marché
Adresse-Marché

IDENTIFIANT : Nom-Marché

TYPE D'ENTITE : Date-Emission

DEFINITION : Une Date-Emission est la date à laquelle une valeur mobilière a été émise.

DUREE DE VIE : Tant qu'il y a un coupon dont la valeur mobilière a été émise à cette date.

ATTRIBUTS : Année-Emission
Mois-Emission
Jour-Emission

IDENTIFIANT : Année-, Mois- et Jour-Emission et Num-Coupon.

TYPE D'ENTITE : Date-Echéance

DEFINITION : Une Date-Echéance est une date à laquelle un coupon ou une obligation vient à échéance.

DUREE DE VIE : Tant qu'il y a un coupon qui vient à échéance ce jour-là.

ATTRIBUTS : Année-Echéance
Mois-Echéance
Jour-Echéance

IDENTIFIANT : Num-Coupon et Année-, Mois- et Jour-Echéance.

TYPE D'ENTITE : Prime-R

DEFINITION : Une Prime-R est le taux de la prime de remboursement d'une obligation.
DUREE DE VIE : Aussi longtemps que le coupon sur lequel porte la prime est présent dans le S.I.
ATTRIBUT : Taux-Prime-R
IDENTIFIANT : Taux-Prime-R et Num-Coupon

TYPE D'ENTITE : Prime-E

DEFINITION : Une Prime-E est le taux de la prime d'émission portant sur une obligation.
DUREE DE VIE : Aussi longtemps que le coupon est présent dans le S.I.
ATTRIBUT : Taux-Prime-E
IDENTIFIANT : Taux-Prime-E et Num-Coupon

TYPE D'ENTITE : Numéro

DEFINITION : Un numéro est l'identification d'une valeur saisie en agence lors d'une opération donnée.
DUREE DE VIE : Tant que l'opération y afférente existe dans le S.I.
ATTRIBUT : Num
IDENTIFIANT : Num et Code-Opération et Num-Opération

TYPE D'ENTITE : Coupure-Saisie

DEFINITION : Une Coupure-Saisie est une coupure saisie en agence lors d'une opération donnée.
DUREE DE VIE : Tant que l'opération y afférente est présente dans le S.I.
ATTRIBUTS : Num-Coupure-Saisie
Type-Coupure-Saisie
IDENTIFIANT : Num-Coupure-Saisie et Code-Opération et Num-Opération.

TYPE D'ENTITE : Agence

DEFINITION : Une agence est l'endroit où a été effectuée l'opération.
DUREE DE VIE : Tant que l'opération est présente dans le S.I.
ATTRIBUTS : Num-Bureau-Comptable
Nom-Gérant-Agence
Adresse-Agence
IDENTIFIANT : Num-Bureau-Comptable

TYPE D'ENTITE : Date-Comptable

DEFINITION : La Date-Comptable est la date à laquelle l'opération est comptabilisée.
DUREE DE VIE : Tant que l'opération est présente dans le S.I.
ATTRIBUTS : Année-Date-Comptable
Mois-Date-Comptable
Jour-Date-Comptable
IDENTIFIANT : Année-, Mois-, Jour-Date-Comptable et Code-Opération et Num-Opération.

TYPE D'ENTITE : Compte-Liquidité

DEFINITION : Le Compte-Liquidité est le compte du client.
DUREE DE VIE : Tant que l'opération est présente dans le S.I.
ATTRIBUT : Num-Compte
IDENTIFIANT : Num-Compte.

TYPE D'ENTITE : Code-Ext

DEFINITION : Le Code-Externe identifie une valeur mobilière particulière selon une codification différente du Secrétariat des Valeurs Mobilières.
DUREE DE VIE : Tant que la valeur mobilière est présente dans le S.I.
ATTRIBUTS : Type-Code-Ext
Code-Externe
IDENTIFIANT : Num-ABB ET Code-Externe.

TYPE D'ENTITE : Tirage

DEFINITION : Cette entité donne les séries de numéros tirés par valeur mobilière.
DUREE DE VIE : Tant que la valeur mobilière est présente dans le S.I.
ATTRIBUTS : Séries-Numéros-Tira
Montant-Unit-Tira
IDENTIFIANT : Num-ABB et Séries-Numéros-Tira

TYPE D'ENTITE : Opposition

DEFINITION : Une opposition frappe une valeur mobilière et empêche tout paiement de cette valeur.
DUREE DE VIE : Tant que cette valeur est présente dans le S.I.
ATTRIBUTS : Séries-Numéros-Oppo
IDENTIFIANT : Num-ABB et Séries-Numéros-Oppo

2.Définition des types d'Associations.

----- TYPE D'ASSOCIATION : Traitement

DEFINITION : Un traitement représente le fait qu'une opération porte sur une valeur mobilière.
DUREE DE VIE : Un Traitement existe tant que l'opération est présente dans le S.I.
ASSOCIE : Opération et Vamo
ROLES : Traite : Opération
Traité-Par : Vamo
CONNECTIVITE : 1-1 pour Traité : Opération
0-N pour Traité-Par : Vamo

TYPE D'ASSOCIATION : Coupe

DEFINITION : Une Coupe représente le fait qu'une Coupure-Emise constitue une découpe de la valeur mobilière.
DUREE DE VIE : Tant que la valeur mobilière est présente dans le S.I.
ASSOCIE : Coupure-Emise et Vamo
ROLES : Coupée-En : Vamo
Découpe : Coupure-Emise
CONNECTIVITE : 1-1 pour Découpe : Coupure-Emise
1-N pour Coupée-En : Vamo

TYPE D'ASSOCIATION : Emission

DEFINITION : Une Emission représente le nombre de valeurs émises par un émetteur.
DUREE DE VIE : Tant que la (les) valeur(s) mobilière(s) (est) sont présente(s) dans le S.I.
ASSOCIE : Emetteur et Vamo
ROLES : Emet : Emetteur
Est-Emise : Vamo
CONNECTIVITE : 1-1 pour Est-Emise : Vamo
1-N pour Emet : Emetteur
ATTRIBUT : Nbre-Emission

TYPE D'ASSOCIATION : Choix

DEFINITION : Un Choix représente le nombre de structures employées par un émetteur pour numéroté ses valeurs mobilières.
DUREE DE VIE : Tant que les valeurs ainsi numérotées sont présentes dans le S.I.
ASSOCIE : Emetteur et Structure
ROLES : Choisit : Emetteur
Choisie-Par : Structure
CONNECTIVITE : 1-N pour Choisit : Emetteur
0-N pour Choisie-Par : Structure
ATTRIBUT : Nbre-Structure

TYPE D'ASSOCIATION : Correspondance

DEFINITION : Une Correspondance représente le fait qu'un Code-Ext correspond au Num-ABB d'une valeur particulière.

DUREE DE VIE : Tant que la valeur est présente dans le S.I.

ASSOCIE : Code-Ext et Vamo

ROLES : Correspond-A : Code-Ext
Représenté-Par : Vamo

CONNECTIVITE : 1-1 pour Correspond-A : Code-Ext
0-N pour Représenté-Par : Vamo

TYPE D'ASSOCIATION : Déclenchement

DEFINITION : Un Déclenchement représente le nombre d'opérations effectuées par un client donné.

DUREE DE VIE : Tant que l'opération est présente dans le S.I.

ASSOCIE : Client et Opération

ROLES : Déclenche : Client
Déclenchée-Par : Opération

CONNECTIVITE : 1-N pour Déclenche : Client
1-1 pour Déclenchée-Par : Opération

ATTRIBUT : Nbre-Opération

TYPE D'ASSOCIATION : Possession

DEFINITION : Une Possession représente le nombre de comptes possédés par un client donné.

DUREE DE VIE : Tant que le Client ne ferme pas son compte et tant que le client est présent dans le S.I.

ASSOCIE : Client et Compte-Liquidité

ROLES : Possède : Client
Possédé-Par : Compte-Liquidité

CONNECTIVITE : 0-N pour Possède : Client
0-1 pour Possédé-Par : Compte-Liquidité

ATTRIBUT : Nbre-Compte-Liq

TYPE D'ASSOCIATION : Crédit

DEFINITION : Un Crédit représente le fait qu'une opération est portée au crédit de tel compte.

DUREE DE VIE : Tant que l'opération est dans le S.I.

ASSOCIE : Compte-Liquidité et Opération

ROLES : Portée-Au-Crédit : Opération
Crédité : Compte-Liquidité

CONNECTIVITE : 1-1 pour Portée-Au-Crédit : Opération
1-N pour Crédité : Compte-Liquidité

TYPE D'ASSOCIATION : Situation-Temporelle

DEFINITION : Représente le fait qu'une opération possède une date-comptable.

DUREE DE VIE : Tant que l'opération est dans le S.I.

ASSOCIE : Date-Comptable et Opération

ROLES : Datée : Opération
Date-Op : Date-Comptable

CONNECTIVITE : 1-1 pour Datée : Opération
1-N pour Date-Op : Date-Comptable

TYPE D'ASSOCIATION : Disposition

DEFINITION : Une Disposition représente le nombre de Comptes-Liquidité mis à la disposition d'une Agence donnée.

DUREE DE VIE : Tant que le Compte-Liquidité est présent dans le S.I.

ASSOCIE : Compte-Liquidité et Agence

ROLES : Disposé-A : Compte-Liquidité
Dispose : Agence

CONNECTIVITE : 0-1 pour Disposé-A : Compte-Liquidité
1-N pour Dispose : Agence

ATTRIBUT : Nbre-Compte-Agence

TYPE D'ASSOCIATION : Gestion

DEFINITION : Une agence assure la gestion de ses clients.

DUREE DE VIE : Tant que le Client et l'Agence sont présents dans le S.I.

ASSOCIE : Client et Agence

ROLES : Géré-Par : Client
Gère : Agence

CONNECTIVITE : 1-N pour Géré-Par : Client
1-N pour Gère : Agence

TYPE D'ASSOCIATION : Encodage

DEFINITION : Un Encodage représente le fait qu'une opération a été encodée dans une agence.

DUREE DE VIE : Tant que l'opération est présente dans le S.I.

ASSOCIE : Opération et Agence

ROLES : Encode : Agence
Encodée-Par : Opération

CONNECTIVITE : 0-N pour Encode : Agence
1-1 pour Encodée-Par : Opération

TYPE D'ASSOCIATION : Découpe

DEFINITION : Une découpe représente le fait qu'une opération porte sur une valeur dont une ou plusieurs coupures ont été saisies.

DUREE DE VIE : Tant que l'Opération est présente dans le S.I.

ASSOCIE : Opération et Coupure-Saisie

ROLES : Découpée : Opération
Saisie-Par : Coupure-Saisie

CONNECTIVITE : 1-N pour Découpée : Opération
1-1 pour Saisie-Par : Coupure-Saisie

ATTRIBUT : Nbre-Coupure-Saisie

TYPE D'ASSOCIATION : Numérotage
 DEFINITION : Un numérotage représente le fait qu'une Coupure-Saisie porte un Numéro.
 DUREE DE VIE : Aussi longtemps que l'Opération est présente dans le S.I.
 ASSOCIE : Coupure-Saisie et Numéro
 ROLES : Numérote : Numéro
 Numérotée : Coupure-Saisie
 CONNECTIVITE : 1-1 pour Numérote : Numéro
 1-1 pour Numérotée : Coupure-Saisie

TYPE D'ASSOCIATION : Saisie
 DEFINITION : Représente le fait qu'un Numéro est saisi par une Opération.
 DUREE DE VIE : Tant que l'Opération est dans le S.I.
 ASSOCIE : Numéro et Opération
 ROLES : Saisi : Numéro
 Saisit : Opération
 CONNECTIVITE : 1-1 pour Saisi : Numéro
 1-N pour Saisit : Opération

TYPE D'ASSOCIATION : Paiement
 DEFINITION : Un paiement exprime qu'une opération porte sur un ou plusieurs coupons.
 DUREE DE VIE : Tant que l'Opération est dans le S.I.
 ASSOCIE : Opération et Coupon
 ROLES : Payé-Par : Coupon
 Paie : Opération
 CONNECTIVITE : 0-N pour Payé-Par : Coupon
 1-N pour Paie : Opération
 ATTRIBUT : Nbre-Coupon

TYPE D'ASSOCIATION : Attachement
 DEFINITION : Un Attachement représente le fait qu'un coupon est attaché à une valeur mobilière.
 DUREE DE VIE : Tant que le coupon et la Vamo sont présents dans le S.I.
 ASSOCIE : Vamo et Coupon
 ROLES : Comprend : Vamo
 Attaché-A : Coupon
 CONNECTIVITE : 0-N pour Comprend : Vamo
 1-1 pour Attaché-A : Coupon

TYPE D'ASSOCIATION : Profit
 DEFINITION : Un remboursement peut profiter d'une prime d'émission.
 DUREE DE VIE : Tant que le coupon sur lequel porte la Prime-E est présent dans le S.I.
 ASSOCIE : Coupon et Prime-E
 ROLES : Profite-De : Coupon
 Profite-A : Prime-E
 CONNECTIVITE : 0-1 pour Profite-De : Coupon
 0-N pour Profite-A : Prime-E

TYPE D'ASSOCIATION : Bénéfice

DEFINITION : Un Coupon bénéficie d'une Prime de Remboursement.
DUREE DE VIE : Tant que le Coupon est dans le S.I.
ASSOCIE : Coupon et Prime-R
ROLES : Bénéficiaire-De : Coupon
Bénéficiaire-A : Prime-R
CONNECTIVITE : 0-1 pour Bénéficiaire-De : Coupon
0-N pour Bénéficiaire-A : Coupon

TYPE D'ASSOCIATION : Echéance

DEFINITION : Indique la date d'échéance du coupon.
DUREE DE VIE : Tant que le Coupon est dans le S.I.
ASSOCIE : Date-Ech et Coupon
ROLES : Echu-Le : Coupon
Porte-Sur : Date-Ech
CONNECTIVITE : 1-1 pour Echu-Le : Coupon
1-N pour Porte-Sur : Date-Ech

TYPE D'ASSOCIATION : Emis.

DEFINITION : Indique que le coupon a une date d'émission.
DUREE DE VIE : Tant que le Coupon est dans le S.I.
ASSOCIE : Coupon et Date-Emission
ROLES : Emis-Le : Coupon
Située-Emission-De : Date-Emission
CONNECTIVITE : 0-1 pour Emis-Le : Coupon
1-N pour Située-Emission-De : Date-Emission

TYPE D'ASSOCIATION : Monnaie

DEFINITION : Indique la monnaie employée pour payer le Coupon.
DUREE DE VIE : Tant que le Coupon est dans le S.I.
ASSOCIE : Devise et Coupon
ROLES : Utilisée-Par : Devise
Payé-En : Coupon
CONNECTIVITE : 0-N pour Utilisée-Par : Devise
1-1 pour Payé-En : Coupon

TYPE D'ASSOCIATION : Valeur

DEFINITION : Fait le lien entre la Devise et son Cours à une date donnée sur un marché donné.
DUREE DE VIE : Tant que le Coupon est dans le S.I.
ASSOCIE : Devise et Cours
ROLES : Valorisée : Devise
Valorise : Cours
CONNECTIVITE : 1-N pour Valorisée : Devise
1-1 pour Valorise : Cours

TYPE D'ASSOCIATION : Lieu

DEFINITION : Un Cours est fixé sur un Marché.
DUREE DE VIE : Tant que le Coupon est dans le S.I.
ASSOCIE : Cours et Marché
ROLES : Localisé : Cours
Localise : Marché
CONNECTIVITE : 1-1 pour Localisé : Cours
1-N pour Localise : Marché

TYPE D'ASSOCIATION : Location-Temporelle

DEFINITION : Indique la date à laquelle la Devise a pris tel Cours.
DUREE DE VIE : Tant que le Coupon est dans le S.I.
ASSOCIE : Date-Cours et Cours
ROLES : Situe-Temporellement : Date-Cours
Situé-Temporellement : Cours
CONNECTIVITE : 1-N pour Situe-Temporellement : Date- Cours
1-1 pour Situé-Temporellement : Cours