

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Utilisation d'une méthode "orientée-scénario" pour la spécification d'un jeu de rôle en Smalltalk/V

Deglin, Vincent

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.

**Facultés Universitaires Notre-Dame de la Paix
Institut d'Informatique**

Rue Grandgagnage, 21
B-5000 NAMUR (Belgium)

**Utilisation d'une méthode
"orientée-scénario" pour la
spécification d'un jeu de
rôle en Smalltalk/V**

Vincent DEGLIN

Promoteur: Professeur Roland Lesuisse

Mémoire présenté en vue de l'obtention
du titre de Licencié et Maître en Informatique

Année académique 1990-1991

Résumé

Ce mémoire de fin d'études expose et utilise une démarche de conception d'animation graphique sur ordinateur d'un jeu de rôle. Cette démarche s'appuie sur la notion de scénario telle qu'elle est présente dans le monde audio-visuel. Elle couvre à la fois les aspects spécification et implantation dans un environnement orienté objet. La démarche est abondamment illustrée par l'étude d'un cas réaliste.

Abstract

This thesis presents and makes use of a scenario-based method for developing graphical animation of parlour games. The idea is that such a method can benefit from techniques used in mass media communications, such as television or animation. The method covers both specification and object-oriented implementation. It is profusely illustrated through a realistic case.

Avant-propos

Je suis particulièrement heureux de remercier ici mon promoteur, Mr Roland Lesuisse qui, tout au long de cette dernière année d'études, a suivi l'évolution de ce travail d'un oeil attentif. Ses conseils éclairés ainsi que ses remarques pertinentes ont permis à ce mémoire de devenir ce qu'il est.

J'adresse également mes plus vifs remerciements à Mr Bernard Moulin pour son accueil, le temps qu'il m'a consacré et l'aide qu'il m'a prodiguée au cours de mon stage à l'Université Laval à Sainte-Foy (Québec). Que soient remerciés avec lui les autres membres de son équipe, avec lesquels une collaboration fructueuse a été possible.

Je suis reconnaissant à Mme Nadine Jamotte-Dachouffe et à Mr Benoît Sacré pour leur aide, souvent ponctuelle, mais précieuse, dans des domaines tant théoriques que techniques.

Enfin, je voudrais exprimer ma gratitude à ma famille et mes amis pour leur soutien et la compréhension qu'ils m'ont témoignés tout au long de mes études.

Table des matières

Résumé et Abstract

Avant-propos

INTRODUCTION 1

Chapitre I : OBJET DE L'ETUDE..... 2

1.1. Introduction 2

1.2. Exposé du projet..... 2

1.3. Définition des concepts..... 4

1.4. Pourquoi le concept de scénario?..... 9

Chapitre II : VERS UNE DEMARCHE DE CONCEPTION BASEE SUR
LA NOTION DE SCENARIO..... 11

2.1. Introduction 11

2.2. L'animation graphique et la notion de scénario en
informatique..... 11

2.2.1. Méthodologies existantes en matière de
conception d'animation graphique..... 12

2.2.2. Quels types de langages pour l'animation
graphique? 12

2.3. Les domaines de la production audio-visuelle..... 13

2.3.1. Analyse de la production audio-visuelle..... 14

2.3.2. Aspects intéressants à retirer de
l'observation du monde audio-visuel..... 18

Chapitre III : CHOIX D'UN TYPE DE LANGAGE CAPABLE DE
SUPPORTER LES NOTIONS D'ACTEUR ET DE SCENARIO..... 19

3.1. Introduction 19

3.2. Types de langages permettant une programmation orientée objet et principaux concepts associés	19
3.3. Propriétés des langages de classes et principaux concepts associés	22
Chapitre IV : METHODOLOGIE DE SPECIFICATION BASEE SUR LA NOTION DE SCENARIO.....	24
4.1. Définition des concepts utilisés et d'un formalisme de représentation de modèles	24
4.1.1. Concepts	24
4.1.2. Définition d'un formalisme de représentation de modèles	25
4.2. Démarche de spécification	26
4.2.1. Canevas de présentation	26
4.2.2. Description du jeu de rôle FRIK	26
4.2.3. Démarche de spécification	27
4.3. Etude d'une rationalisation de la méthode	70
4.3.1. Rapprochement du réseau de transition entre jalons et des cycles de vie des acteurs	71
4.3.2. Rationalisation dans la découpe du scénario	73
4.3.3. Analyse des données	76
4.4. Une étape supplémentaire : la spécification par prédicats	79
4.4.1. Explication théorique	79
4.4.2. Illustration	81
4.4.3. Réflexions sur les prédicats	82

4.5. Intérêt des différents documents rédigés pour l'implantation	84
Chapitre V : APPROCHE THEORIQUE D'IMPLANTATION BASEE SUR LA NOTION DE SCENARIO	85
5.1. Introduction	85
5.2. Démarche d'implantation.....	85
5.2.1. Initialisation du système.....	86
5.2.2. Identification des classes	87
5.2.3. Description de l'état des objets.....	90
5.2.4. Etablissement de la liste des interfaces des objets	91
5.2.5. Implantation des méthodes.....	92
5.3. Etape d'implantation des méthodes : une démarche?	93
5.3.1. Les messages liés au système adopté lui-même.....	94
5.3.2. Les messages représentant un comportement.....	94
Chapitre VI : REFLEXIONS ET CRITIQUES.....	98
6.1. Introduction	98
6.2. Evaluation de la méthode	98
6.3. Perspectives et prolongements envisageables	99
CONCLUSION	102
BIBLIOGRAPHIE	104
ANNEXES	106

INTRODUCTION

Chacun d'entre nous a certainement déjà joué dans son enfance à un jeu ou l'autre. Un des jeux préférés de bon nombre d'enfants consiste à se raconter une histoire et, à grand renfort d'imagination, à le jouer à l'aide d'objets n'ayant en général aucun rapport avec celle-ci.

Ayant grandi, l'âge et la formation aidant, certains de ces enfants, devenus adolescents ou adultes entre temps, continuent de jouer à ce jeu en l'adaptant quelque peu : ils se créent en effet une représentation de leurs histoires en se construisant des animations graphiques sur ordinateur. D'autres encore poussent leur passion plus loin en construisant des animations graphiques sur ordinateur de leurs jeux favoris. C'est dans ce cadre que s'inscrit ce travail.

Ce mémoire propose et utilise une démarche de conception d'animation graphique sur ordinateur d'un jeu de rôle. Avant d'aborder l'élaboration de cette démarche, nous présenterons au chapitre I l'objet de notre étude. Au chapitre II, nous proposerons ensuite une investigation de la littérature pouvant nous aider à concevoir notre démarche. Nous analyserons alors, dans un troisième chapitre, les différents types de langages pouvant supporter l'élaboration de la démarche. Nous pourrons dès lors exposer cette démarche de spécification et d'implantation d'animation graphique d'un jeu de rôle aux chapitres IV et V et nous l'appliquerons à un exemple réaliste. Enfin, le chapitre VI apportera quelques réflexions et critiques vis-à-vis de la démarche proposée.

Chapitre I : OBJET DE L'ETUDE

1.1. INTRODUCTION

La finalité de ce chapitre est d'exposer le projet qui est à la base de ce mémoire, de décrire les différents concepts clés qui seront utilisés, et de nous interroger sur l'opportunité du choix de la notion de scénario comme base de la spécification d'une animation graphique d'un jeu de rôle sur ordinateur.

1.2. EXPOSE DU PROJET

Le travail présenté dans les chapitres qui suivent vise à fournir une animation graphique sur ordinateur d'un jeu de rôle. Intuitivement, on peut définir une animation graphique sur ordinateur comme une succession rapide d'images sur un écran d'ordinateur en vue de donner une illusion de mouvement. De la même manière, un jeu de rôle est un jeu qui rassemble plusieurs participants soumis à des règles dictées par un joueur spécial appelé "Meneur de jeu". Chaque joueur a un objectif qui lui est propre et qu'il tentera de réaliser durant la partie. Le Meneur de jeu, comme son nom l'indique, est celui qui dirige le jeu et veille au bon déroulement de la partie entamée.

Une des caractéristiques principales d'un jeu de rôle est l'existence d'un scénario ¹ décrivant toutes les règles (souvent nombreuses) à respecter. Nous allons précisément articuler ce travail autour de la notion de scénario.

En effet, en 1989-1990, une méthode permettant de spécifier des systèmes complexes sous la forme de scénarios et d'acteurs (structures de connaissances, comportements, coordination, etc) a été développée

¹ Avant d'en donner une définition plus précise, nous considérerons un scénario comme une marche à suivre.

par Moulin, de Looz et Jourquin ([DEL-JOU]). Cette méthode inspirée de la méthode EPAS [MOUL-88] intègre diverses idées nouvelles telles que la spécification de scénarios et l'introduction d'un système appelé meneur de jeu (MJ) pour la coordination des échanges entre acteurs.

Cette méthode fut utilisée afin de simuler les déplacements des divers acteurs d'une agence bancaire et le système ainsi spécifié fut implanté en Smalltalk/V.

Les résultats de ces travaux ont été publiés dans un mémoire proposant une démarche de conception d'animation graphique sur ordinateur [DEL-JOU]. Le travail proposé ci-après s'inspire largement de ces résultats. Cependant le domaine d'application de la méthode a glissé du bureau vers l'espace ludique.

Un des buts du présent projet est d'explorer l'utilisation de cette méthode, non plus pour la modélisation de l'organisation d'un bureau, mais pour la spécification d'un jeu de rôle informatisé.

La méthode est-elle assez forte? Est-elle bien indiquée? Quels problèmes avons-nous rencontrés lors de son application? Quelles modifications ou précisions apporter pour la rendre plus puissante? Une certaine rationalisation ² ne serait-elle pas possible? Telles sont les questions auxquelles nous allons tenter de répondre.

Afin d'aboutir à une animation graphique d'un jeu de rôle informatisé, et donc que l'ordinateur puisse dessiner et animer un jeu de rôle, il faut lui fournir une description dynamique détaillée des composantes du jeu. Nous reprendrons donc l'idée originale de l'approche proposée par de Looz et Jourquin [DEL-JOU], à savoir la présentation de cette description sous la forme d'un **scénario**.

Mais avant de poursuivre, il nous paraît nécessaire de définir les quelques concepts clés qui seront utilisés tout au long de la méthode.

² Au chapitre 4, nous ne manquerons pas de remarquer que la méthode de spécification proposée semble induire une certaine redondance. La rationalisation invoquée consisterait principalement à essayer d'éliminer celle-ci.

1.3. DEFINITION DES CONCEPTS

Les différents concepts que nous utiliserons ont, tous, été défini par [DEL-JOU] à l'exception du concept de jeu de rôle. Nous nous inspirerons donc de tout ou partie de leurs définitions ci-dessous.

Nous donnerons tout d'abord une définition plus rigoureuse des concepts d'animation graphique sur ordinateur et de jeu de rôle.

a) Animation graphique sur ordinateur

" **L'animation graphique sur ordinateur** se rapporte au processus de génération dynamique sur un écran d'ordinateur d'une série d'images d'un ensemble d'objets, dans lequel chaque image est une altération de l'image précédente. Cette technique donne l'illusion de mouvement grâce à la cadence de projection des images à l'écran (typiquement vingt-quatre images par seconde) ; c'est une application du phénomène bien connu de persistance rétinienne. " ([DEL-JOU], p.4)

Les principales activités qu'implique une animation graphique sur ordinateur sont au nombre de trois, à savoir la création d'objets à animer, la spécification et la synchronisation des mouvements et enfin la visualisation et le traitement de l'image.

L'ensemble des objets dont il est question ci-dessus définit ce que nous appelons le **domaine d'animation**. " Ces objets générés à l'écran sont en quelque sorte les "acteurs" de l'animation. " ([DEL-JOU], p.4)

b) Jeu de rôle

Un jeu de rôle est un jeu rassemblant autour d'une table plusieurs participants. L'un des joueurs est appelé "Meneur de jeu"-MJ-. Celui-ci invite les autres joueurs à tenir un rôle et à explorer un univers soumis à de nombreuses règles et précisément décrit dans un scénario. Chaque joueur va choisir, façonner et incarner un personnage qui sera investi d'une mission à réaliser. Chaque personnage est doté de caractéristiques physiques et psychologiques spécifiques, déterminées en début de partie. La fonction du Meneur de jeu est essentielle car c'est lui qui dirige les aventures, articule les séquences de simulation en

fonction du scénario et incarne tous les personnages secondaires de l'histoire. ([J&S])

Reprenons un à un les différents éléments intervenant dans la définition ci-dessus en les adaptant à notre contexte.

Un jeu de rôle se déroule en général autour d'une table (en fait, un jeu de rôle peut se dérouler n'importe où, à la seule condition que les joueurs puissent parler !). Dans notre cas, le jeu se déroulera sur un écran d'ordinateur.

Un jeu de rôle met en oeuvre des participants que nous appellerons "joueurs". En fait, ces joueurs sont des acteurs (cf point e). Et comme nous l'avons vu ci-dessus, les joueurs doivent pouvoir se parler. Ainsi, les échanges d'informations entre ceux-ci se feront via une interface.

L'un de ces joueurs joue un rôle spécifique propre à tous les jeux de rôle. Il s'agit du Meneur de jeu. Il va faire jouer un scénario (cf point d). Pour ce faire, il décrit aux joueurs la situation initiale et leur expose leur mission. Il veille également à l'application des règles lors de la création des personnages puis tout au long de la partie où il répondra aux joueurs et leur indiquera les conséquences des actions qu'ils entreprendront. Dès lors, outre les règles qu'il doit parfaitement maîtriser, il doit connaître précisément le scénario qu'il fait jouer. Il a en effet un rôle d'arbitrage. Mais nous reviendrons plus tard sur cet acteur particulier et crucial. Notons cependant déjà que, dans notre animation graphique, il ne sera pas incarné par un des joueurs mais par un acteur invisible, en ce sens qu'il n'est pas représenté à l'écran. Ceci aura pour conséquence directe que le nombre de joueurs sera identique au nombre de participants.

Comme nous l'avons vu, le Meneur de jeu veille à l'application des règles. En effet, comme tout jeu, un jeu de rôle possède des règles : on ne fait pas plus "n'importe quoi" dans un jeu de rôle qu'au go ou au bridge. Les règles se divisent en général en trois parties. La première d'entre elles concerne le plus souvent la création des personnages. La deuxième regroupe l'ensemble des moyens de gérer une partie, ainsi que les événements divers : ce sont les règles de simulation. La dernière enfin nous donne un aperçu du monde dans lequel se passe le jeu. Toutefois, nombre de jeux mélangent allègrement les trois parties et

produisent en un seul document un amalgame de toutes les règles (nous en aurons d'ailleurs un bel exemple avec le jeu qui nous servira d'illustration plus loin). Qu'à cela ne tienne, ceci n'est pas de la plus haute importance. On repérera pour toute règle la partie à laquelle elle appartient. Si elle appartient au premier groupe, elle concerne les caractéristiques des joueurs (par exemple le portefeuille des joueurs). Si elle appartient au deuxième groupe, elle concerne alors le "fonctionnement" du jeu et doit dans ce cas être intégrée dans le scénario (par exemple, si un joueur tombe sur une certaine case, il peut effectuer telle ou telle action). Enfin, si elle appartient à la dernière catégorie, deux possibilités s'offrent à nous : soit la règle n'est qu'une règle traduisant l'esprit dans lequel le jeu se déroule (par exemple, la police d'assurance aide à payer les frais encourus à la suite de catastrophe) et dans ce cas elle n'intervient pas réellement au niveau du scénario, soit elle traduit un fait propre au monde dans lequel on se trouve (par exemple, le fait qu'une dette ne puisse être étalée ou reportée et doive être payée immédiatement) et dans ce cas elle doit être intégrée au scénario.

La plupart des règles vont donc être introduites dans le scénario. Celui-ci est la base de l'aventure que vont vivre les joueurs. Cette notion étant le fil conducteur de ce travail, nous y reviendrons plus en détail ci-dessous (cf point d).

Un jeu de rôle met en compétition les joueurs. Chaque joueur est en effet investi d'une mission qui lui est propre et qui est contradictoire (du moins partiellement) des missions des autres joueurs. Ainsi, de façon générale, la réussite de l'objectif personnel d'un joueur implique l'échec des objectifs adverses. Toutefois, le but à atteindre peut être collectif. Dans ce cas, les joueurs sont voués à une collaboration étroite et leur réussite personnelle dépend de la réussite de l'objectif collectif ³.

Chaque joueur va devoir tenir compte pour la réalisation de sa mission des caractéristiques, tant physiques que psychologiques, qui lui

³ Notons encore qu'il existe beaucoup de variantes intermédiaires en matière de but à atteindre.

ont été attribuées pour incarner son personnage. Ces caractéristiques peuvent être des contraintes (par exemple des contraintes financières), des reconnaissances de dons et/ou de capacités particulières (par exemple le personnage est le roi des pickpockets) ou plus simplement des caractéristiques n'influençant pas (ou presque pas) le jeu lui-même mais y ajoutant un plus en crédibilité en donnant une description détaillée de ses personnages (par exemple, le personnage est petit). Ces dernières caractéristiques ne seront d'ailleurs reprises pour l'animation graphique que dans la mesure où les concepteurs trouvent intéressant de représenter chaque personnage tel qu'il est "réellement".

Toute partie d'un jeu de rôle est décomposable en une suite de petites séquences. Chacune de ces séquences s'articule de la façon suivante :

1° Le Meneur de jeu décrit aux joueurs la situation dans laquelle ils se trouvent, de leur point de vue, à un moment donné;

2° Les joueurs décident de l'action qu'entreprennent leurs personnages et l'annoncent au Meneur de jeu;

3° Le Meneur de jeu, avec l'aide du scénario et par l'utilisation des règles, décide des conséquences du comportement des personnages;

4° On revient en 1°, avec une situation modifiée. Et ainsi de suite... .

Enfin, le Meneur de jeu incarne tous les personnages non joueurs. Ainsi, il incarnera aussi bien l'ennemi irréductible que l'allié précieux. Un de ces personnages pourrait être par exemple le banquier (ou en général, la banque).

Un jeu de rôle brasse, comme nous pouvons nous en apercevoir, un nombre important de données. Il est évident que toutes ces données, qui devront être traitées, doivent être organisées. Elles le seront par un programme appelé "système".

c) Système

" Un **système**, au sens informatique du terme, est constitué, selon [LEGA-75], d'un ensemble de méthodes, de procédures, de données, de programmes, organisés pour un traitement de l'information. " ([DEL-

JOU], p. 5). Dans le contexte qui nous préoccupe, nous entendons par "système" le système d'information sous-jacent à l'animation graphique d'un jeu de rôle.

Un jeu de rôle propose à des joueurs d'explorer un univers soumis à de nombreuses règles. Cet univers est décrit dans un scénario.

d) Scénario

La définition commune d'un **scénario** peut s'énoncer ainsi : canevas d'une pièce, d'un roman, ou rédaction détaillée des diverses scènes dont un film sera composé.

Nous élargirons cette notion au domaine de l'animation graphique sur ordinateur car on peut voir cette dernière comme le jeu d'un certain nombre d'acteurs dotés d'un ensemble de comportements. Ces comportements sont dictés par le système et s'accordent avec un "comportement global". C'est ce comportement global, préalablement défini pour l'ensemble des acteurs, que nous appellerons scénario.

" Cette définition n'exclut pas une part d'improvisation dans un scénario; en effet, le comportement global décrit pour chaque acteur les comportements possibles, en fonction des circonstances. Un même scénario peut donc donner lieu à une multitude d'enchaînements de comportements " ([DEL-JOU], p. 6)

Enfin, une animation graphique sur ordinateur d'un jeu de rôle vise à mettre en interaction des acteurs dotés de comportements selon un scénario donné.

e) Acteur

Sera considéré comme **acteur** tout personnage de l'action et tout objet qui joue un rôle (actif ou passif) dans le domaine d'animation. Ainsi, par exemple, les joueurs sont considérés comme des acteurs, au même titre que les cases du jeu, les cartes, les dés, voire même une négociation entre joueurs. Par contre, un dessin sur une case ne sera

vraisemblablement pas considéré comme acteur, à moins qu'il ne joue un quelconque rôle autre que décoratif. " Tout acteur peut avoir un ensemble de comportements (cf point f), être caractérisé par un état, être vecteur d'information, être visible ou invisible par rapport au domaine d'animation, etc. " ([DEL-JOU], p. 4)

f) Comportement

" Un **comportement** est une action, de tout type, exécutée par un acteur et définie dans un système. Cette action peut être visualisable dans le domaine d'animation ou ne pas l'être. Le comportement d'un acteur peut être caractérisé par un mouvement de cet acteur, un changement d'aspect à l'écran de cet acteur, (...), un échange d'information de cet acteur avec un autre acteur, un changement d'état de cet acteur, etc. " ([DEL-JOU], p. 5)

Maintenant que nous disposons de ces concepts, il nous est loisible de nous interroger sur l'opportunité du concept de scénario comme base à notre travail.

1.4. POURQUOI LE CONCEPT DE SCENARIO?

L'idée de baser la démarche de spécification de systèmes complexes sur la notion de scénario, comme l'indique [DEL-JOU], nous conduit à considérer une animation graphique comme un ensemble d'éléments (en fait, les **acteurs** de l'animation) jouant un rôle composé d'une succession de comportements. L'animation graphique sera donc considérée comme un ensemble de processus ⁴ interdépendants; chacun de ces processus se rapporte à un acteur du scénario.

⁴ Un processus représente un comportement, une activité du système. Il représente ce que fait le système. [MOU-88]

Cette idée de scénario a l'avantage d'avoir été utilisée avec succès dans les domaines audio-visuels. Pourquoi ne pourrait-elle pas être adaptée à l'animation graphique sur ordinateur, et plus précisément dans le cas qui nous concerne à la spécification d'un jeu de rôle? Nous pouvons en effet considérer un jeu comme un film (à l'issue incertaine, certes, mais un film quand même !) dans lequel interagissent des acteurs entre-eux (les joueurs) et se comportant de manière à favoriser leurs propres intérêts. Nous retrouvons donc tous les ingrédients nécessaires à l'utilisation d'une méthode qui serait orientée-scénario.

Chapitre II : VERS UNE DEMARCHE DE CONCEPTION BASEE SUR LA NOTION DE SCENARIO

2.1. INTRODUCTION

Nous allons, dans ce chapitre, relever dans la littérature spécifique au domaine informatique aussi bien que dans celle qui lui est étrangère des références sur lesquelles nous puissions nous fonder pour élaborer une démarche de conception d'animation graphique basée sur la rédaction d'un scénario.

Dans un premier temps, nous nous intéresserons aux éléments de démarches méthodologiques de conception d'animation graphique existants dans la littérature informatique et compatibles avec la notion de scénario. Nous tenterons ensuite de déterminer quels types de langages se prêtent le mieux à la conception d'animation graphique.

Dans un deuxième temps, nous essaierons de tirer des bénéfices de l'observation de domaines extérieurs au domaine informatique et susceptibles de nous inspirer dans notre démarche de conception d'animation graphique basée sur la notion de scénario.

2.2. L'ANIMATION GRAPHIQUE ET LA NOTION DE SCENARIO EN INFORMATIQUE

Pour évoquer ce problème d'élaboration de méthodologie, de Looz et Jourquin, dans [DEL-JOU], ont recherché à travers la littérature informatique existante une démarche de conception. Ils se sont ensuite penchés sur les outils permettant la réalisation d'une animation graphique sur ordinateur. Les résultats de leurs recherches sont décrits ci-dessous.

2.2.1. Méthodologies existantes en matière de conception d'animation graphique.

L'animation graphique sur ordinateur est un domaine qui reste à explorer et pour lequel on ne trouve pas de méthodologie dans la littérature.

" Toutefois, la notion de scénario ou d'acteur se retrouve dans des systèmes de plus en plus nombreux, sous une forme plus ou moins élaborée. A l'ère de l'intelligence artificielle, en effet, il devient fréquent de vouloir concevoir des systèmes d'information mettant en oeuvre des acteurs dotés d'un ensemble de comportements. Ceci est l'esprit de la programmation orientée objet. Ces comportements s'accordent avec un "comportement global", c'est-à-dire en fait avec un scénario d'ensemble. " ([DEL-JOU], p.10)

L'idée sous-jacente à tout raisonnement orienté objet est donc de modéliser un système complexe comme un ensemble d'acteurs autonomes interagissant entre-eux selon un scénario donné.

Mais bien que présente en filigrane dans la littérature récente, la notion de scénario comme base réelle d'une démarche de conception d'animation graphique n'a pas été étudiée en tant que telle.

2.2.2. Quels types de langages pour l'animation graphique?

Afin de répondre à cette question, commençons par rappeler les principales activités qu'implique une animation graphique sur ordinateur. Elles sont au nombre de trois : la création d'objets à animer, la spécification et la synchronisation des mouvements et enfin la visualisation et le traitement de l'image.

Le langage choisi doit ainsi comporter des outils permettant la définition structurée des objets et opérations associées, le parallélisme au niveau des comportements des objets et la communication entre objets.

Il semble donc que la programmation "orientée objet" convienne tout particulièrement. " Nous appelons objet un ensemble de données

associé, en un même concept, aux procédures chargées de les exploiter.
" ⁵ ([DEL-JOU], p.12)

Tous les langages conventionnels permettent d'adopter un style de programmation objet. Cependant, seuls certains d'entre eux nous offre des mécanismes nous permettant d'utiliser directement les objets. Nous les examinerons de manière plus approfondie au chapitre suivant (point 3.2).

D'autres outils d'animation sur ordinateur fournissent le moyen de manipuler avec précision des images. Ces outils sont des outils de graphisme pur, c'est-à-dire manipulant des pixels (points minuscules constituant l'image). Ils ne se préoccupent cependant pas de ce que ces images mettent en oeuvre différents acteurs "autonomes". De ce fait, ils ne nous intéresseront pas.

Suite au constat de l'insuffisance de la réponse qu'apporte la littérature informatique à une méthodologie basée sur la notion de scénario, nous avons donc décidé d'élargir notre champ de recherche à un domaine qui a éprouvé (avec succès) cette notion de scénario, à savoir le monde audio-visuel.

2.3. LES DOMAINES DE LA PRODUCTION AUDIO-VISUELLE

Les domaines de la production audio-visuelle, à savoir la production de films cinématographiques, d'émissions télévisées et de dessins animés, ont paru particulièrement intéressant au yeux de de Looz et Jourquin. En effet, le scénario n'est-il pas à la base de tout film, de toute émission télévisée et de tout dessin animé? Bien que le

⁵ Rappelons qu'un langage autorise une "programmation orientée objet" s'il permet, contrairement à la démarche de la programmation classique, de donner la priorité aux objets manipulés par le système, en associant les programmes aux données, et s'il permet de définir et d'utiliser des objets décrits par leurs caractéristiques externes (opérations applicables sur l'objet) et non pas leur représentation.

produit final soit différent, on peut penser que la démarche méthodologique puisse, elle, être sensiblement identique.

Dans un premier temps, nous analyserons ces trois domaines en décrivant les étapes techniques du travail de production, ainsi que les rôles respectifs des différents membres de l'équipe de production.

Dans un second temps, nous nous attacherons à retirer les aspects intéressants pour la démarche de conception de cette observation du monde audio-visuel.

2.3.1. Analyse de la production audio-visuelle

Dans ce point, nous nous bornerons uniquement à donner un résumé succinct des recherches menées par de Looz et Jourquin dans [DEL-JOU]. Le lecteur intéressé par l'analyse complète se référera à [DEL-JOU], p. 14 à 22.

Nous commencerons par définir les différentes étapes du travail de production dans les trois domaines de la production audio-visuelle.

a) Les étapes du travail de production

L'**idée originale** est l'idée se situant au départ de toute production artistique. C'est elle qui sera développée et adaptée en fonction du moyen d'expression utilisé :

- Au **cinéma**, de cette idée, un texte court par lequel le sujet est présenté au producteur est rédigé. Ce texte, donnant une vision globale de l'histoire, est appelé **synopsis**. Un scénariste adapte alors ce synopsis pour le cinéma en définissant les **parties** essentielles de l'histoire. Ensuite, il réalise une ébauche élaborée du scénario définitif (la continuité dialoguée) en divisant le scénario en **séquences** et en **scènes**. La séquence est une suite d'images significatives et la scène est une partie de séquence se déroulant dans un même décor et entre les mêmes personnages. Finalement, le réalisateur procède à un découpage technique intégrant les contraintes techniques et les indications indispensables à l'équipe de réalisation. Il résulte de cette phase une découpe en **plans** du film, un plan se définissant comme une suite

d'images cinématographiques tournées sans discontinuité de prise de vue.

- En matière de production d'**émissions télévisées** ⁶, on procède également à un raffinement successif de l'idée originale. Toutefois, celui-ci est conduit moins loin puisque le jeu final des acteurs n'est pas déterminé à l'avance. Le scénario imaginé à partir de l'idée originale aboutit à " l'élaboration d'une **marche à suivre** indiquant pour chaque **séquence** de l'émission son titre, les acteurs qui y interviennent, sa durée estimée, l'emplacement et le rôle spécifique des différents éléments du décor, la position des caméra, etc. Cette marche à suivre servira de base à la rédaction plus formelle du **script**. Le script est le document de travail de référence qui servira de support aux prises de vue finales. Il présente de manière beaucoup plus approfondie le détail des scènes qui se succéderont dans l'émission, le texte de support, le rôle des personnages et différentes indications techniques. " ([DEL-JOU], p. 17)

- En matière de production de **dessins animés**, l'idée originale induit une première esquisse des personnages et des principaux traits de décor. Suivra une définition plus précise des personnages, tant au niveau des caractéristiques physiques, de la démarche, des comportements, qu'au niveau du style, des habitudes ou du vocabulaire. On dépeint ensuite l'histoire au travers d'un **story-board**. Un story-board est constitué d'une série de dessins décrivant les moments clés du récit ainsi que les indications techniques s'y rapportant. Ce document conduira le travail de l'équipe de production. Les dessins intermédiaires (c'est-à-dire ceux se situant entre les planches du story-board) sont réalisés par une équipe d'animation. Une première photographie et une projection sont alors effectuées afin de relever les anomalies éventuelles survenant entre dessins. Dès que le résultat se révèle satisfaisant, les dessins sont transposés sur feuilles

⁶ Nous nous intéressons plus particulièrement aux émissions en direct car les émissions enregistrées ne sont finalement que du cinéma ([DEL-JOU], p. 14)

transparentes (appelées **cels**), coloriées et enfin photographiées. Après réalisation de la bande sonore, l'**éditeur** donnera son aspect définitif au produit.

Après avoir défini les différentes étapes du travail de production, passons maintenant en revue les différentes personnes responsables de ces tâches.

b) Les différents membres d'une équipe de production et leur rôle

• " Au **cinéma**, on distingue les rôles suivants :

- l'auteur, au départ d'une **idée originale**, rédige un **synopsis** ;
- le producteur a pour rôle de rassembler les fonds nécessaires à la réalisation du film et de gérer l'utilisation de ces ressources financières ;
- le directeur de production est chargé de coordonner et de contrôler l'exécution du film ;
- le régisseur général est responsable de l'infrastructure et de l'administration liées au travail de l'équipe ;
- le scénariste a pour rôle d'adapter le synopsis pour le cinéma ;
- le réalisateur a pour rôle d'écrire le **découpage technique** et de mettre en place les ressources physiques nécessaires à sa réalisation. Du fait que ce découpage technique constitue l'instrument de travail de référence, le réalisateur est le véritable maître d'oeuvre ;
- d'autres personnes (comme le décorateur, l'accessoiriste, etc) sont également fortement impliquées dans le travail de production mais leur rôle s'éloigne du cadre qui nous préoccupe.

• A la **télévision**, on distingue habituellement les rôles suivants :

- l'auteur a pour rôle de décrire son **idée originale**. Cette description servira de support à l'émission télévisée que l'on veut produire ;
- le scénariste a pour rôle de concrétiser les idées de l'auteur en un **scénario** ;
- le producteur est le responsable de l'équipe et du travail réalisé ;
- l'assistant de production joue un rôle déterminant aux côtés du producteur. Sa fonction comprend des aspects organisationnels, planificateurs, de repérage et de travail en studio. Il doit veiller à la

bonne organisation générale des scènes. C'est lui qui connaît la **marche à suivre** et qui veille à ce qu'elle soit respectée. L'assistant de production déclenche les actions de chacun des acteurs qui connaissent le détail de leur rôle. Il est également responsable du **script** et doit pouvoir organiser le travail global au travers des différentes prises de vues. C'est cet aspect ordonnateur de l'assistant de production que nous retenons surtout;

- le directeur est le véritable chef sur le plateau. Il met en scène les acteurs et est responsable de l'interprétation du script ;

- d'autres personnes (comme la secrétaire de production, les responsables des décors, etc) gravitent autour du producteur ou du directeur dans le but de leur faciliter le travail.

• En **cinéma d'animation**, on distingue les rôles suivants :

- l'auteur soumet ses idées au directeur de production ;
- le directeur de production a pour fonction de rechercher et d'imaginer de nouveaux projets et de les proposer à son équipe. Il supervise l'élaboration du scénario et des dialogues ;
- l'assistant de production doit rassembler toutes les données utiles à l'équipe et élaborer un plan de travail à fournir ;
- le scénariste a pour rôle de créer une intrigue, un scénario ;
- le travail de l'esquisseur consiste à réaliser une première ébauche des dessins, d'après le travail du scénariste ;
- le personnageiste a pour rôle de préciser la définition des différents personnages ;
- le dessinateur doit dessiner le décor et les personnages ; il doit être observateur et faire preuve d'imagination ;
- l'animateur est plus technicien. Il doit pouvoir indiquer, par exemple, que quand tel personnage se déplace dans telle scène, seules les jambes doivent être redessinées, le reste du corps restant fixe ;
- l'assistant animateur a pour rôle d'ajouter les dessins intermédiaires entre deux planches du **story-board** ;
- la révision du travail de l'animateur et de son équipe est assurée par l'ordonnateur ;
- le travail du **caméraman** consiste à réaliser les photographies
- l'éditeur a pour fonction de donner à l'animation son aspect définitif (greffe de la bande sonore, ...). " ([DEL-JOU], p. 18 à 20)

2.3.2. Aspects intéressants à retirer de l'observation du monde audio-visuel

De l'analyse évoquée ci-dessus, de Looz et Jourquin ont retiré trois aspects de ces méthodologies qui nous intéressent au plus haut point.

Premièrement, l'idée maîtresse est celle d'un **scénario affiné** progressivement, au départ d'une idée originale.

Deuxièmement, le rôle de l'équipe de production semble prépondérant. Il s'avère en effet indispensable de prendre les bonnes décisions au bon moment. Ceci implique donc la nécessité d'un "meneur de jeu" qui déclenche les entrées en scène et les actions prévues des différents acteurs à l'instant requis.

Enfin, les acteurs et leurs attributs (caractéristiques physiques et comportements attendus) doivent être définis avec précision dès le départ, mais en leur maintenant une certaine marge de liberté.

Ces idées seront reprises dans la démarche de conception d'animation graphique.

Afin de disposer d'une base de travail ferme pour l'élaboration d'une méthode de spécification d'animation graphique, il reste à poser un choix entre les types de langages de programmation. Ce choix fera l'objet du chapitre suivant.

Chapitre III : CHOIX D'UN TYPE DE LANGAGE CAPABLE DE SUPPORTER LES NOTIONS D'ACTEUR ET DE SCENARIO

3.1. INTRODUCTION

On peut se demander pourquoi nous nous intéressons déjà au type de langage dans lequel sera implanté l'animation. En fait, il est nécessaire de connaître les principaux concepts propres au type de langage de programmation que l'on va utiliser si l'on veut représenter la spécification d'un scénario en une animation graphique de la manière la plus directe possible.

Au chapitre précédent, nous avons estimé que l'utilisation d'un langage permettant une programmation orientée objet était le plus adéquat. Un tel langage vise d'une part à associer, en un même concept, les données et les procédures chargées de les exploiter et, d'autre part à séparer les concepts manipulés dans les programmes et leur représentation interne en machine.

Plusieurs types de langages permettent une telle approche. Nous allons les passer en revue et en choisir un.

3.2. TYPES DE LANGAGES PERMETTANT UNE PROGRAMMATION ORIENTEE OBJET ET PRINCIPAUX CONCEPTS ASSOCIES

" Tout langage permettant une programmation orientée objet se fonde sur les concepts d'**encapsulation** et d'**abstraction des données**.

L'encapsulation consiste à considérer que les données et les procédures qui les manipulent sont groupées dans une même entité : l'objet.

L'abstraction des données est le principe selon lequel il n'y a pas besoin de connaître l'implantation et la représentation internes d'un objet pour lui adresser un message, c'est-à-dire pour utiliser une procédure (ou "méthode") appartenant à cet objet.

[ROCH-87] distingue schématiquement deux familles de langages permettant une programmation orientée objet.

La première famille est constituée de langages "classiques", au sens de la séparation des données et des programmes, mais qui possèdent des constructions syntaxiques permettant l'encapsulation et l'abstraction des données (ex. : ADA) " ([DEL-JOU], p.24)

La seconde famille est constituée des langages orientés objet proprement dits. Ces langages possèdent les notions d'encapsulation et d'abstraction des données auxquelles viennent s'ajouter d'autres propriétés en fonction du point de vue adopté. Nous pouvons en effet compter trois points de vue :

a) Le point de vue structurel des langages de classes.

" L'objet est un type de données, modèle pour la structure de ses représentants physiques. Le concept de classe est un concept de haut niveau s'apparentant à un type abstrait de données ([LISK-74] et [GUTT-77]). Une classe décrit la structure et le comportement partagés par un ensemble d'objets. Les représentants physiques de la classe, appelés instances, sont obtenus par "moulage". "([DEL-JOU], p 25) . Les classes peuvent pour leur part être organisées de façon hiérarchique. Cette organisation permet la mise en oeuvre d'un mécanisme d'héritage. (ex. : Simula et Smalltalk-80)

b) Le point de vue conceptuel des langages de frames.

" L'objet (frame) est une unité de connaissance, représentant le prototype d'un concept. En effet, un frame est un prototype décrivant une situation ou un objet standard. L'ensemble des frames est organisé en une hiérarchie d'héritage où tout objet est à la fois un représentant des frames dont il est issu et un générateur de frames plus spécialisés.

La notion de frame est précisée dans [MINS-75] " ([DEL-JOU], p.25) (ex. : KRL et KL-One)

c) Le point de vue "acteur" des langages d'acteurs

" Un acteur est défini comme un objet qui peut envoyer ou recevoir des messages. Tous les éléments d'un système sont des acteurs et la seule activité possible dans le système est la transmission de messages entre ces acteurs. Programmer revient simplement à dire aux différentes classes d'acteurs comment répondre aux messages qu'elles reçoivent. Les acteurs sont des entités autonomes et leur comportement est défini par une sorte de scénario, qui décrit la manière dont ils doivent réagir aux événements provoqués par les autres acteurs. " ([DEL-JOU], p. 26) (ex. : Plasma, Act1 et Act2)

Notons enfin que de nombreux langages empruntent des caractéristiques à l'une et l'autre de ces familles. Nous les classerons dans la famille des langages hybrides.

Deux types de langages semblent particulièrement convenir : les langages de classes et les langages d'acteurs. Notre choix se portera cependant sur les langages de classes. Les raisons de ce choix sont la possibilité d'organiser hiérarchiquement les classes (ce que ne permettent pas les langages d'acteurs) et la possibilité de décrire au niveau d'un même objet la structure et le comportement associés à chaque acteur de l'animation graphique.

La méthode qui sera exposée aux chapitres IV et V sera donc adaptée aux langages orientés objets, et plus particulièrement aux langages de classes. Passons donc rapidement en revue les caractéristiques et propriétés des langages de classes.

3.3. PROPRIETES DES LANGAGES DE CLASSES ET PRINCIPAUX CONCEPTS ASSOCIES

Intuitivement, la notion de classe permet de regrouper les objets de description (structures de données ou champs) et de comportements (méthodes applicables) identiques.

" D'une manière plus formelle, une **classe** est définie par un modèle d'un ensemble de données et par un ensemble d'opérateurs applicables sur cet ensemble de données. " ([DEL-JOU], p. 27). On peut créer ainsi une hiérarchie de classes dans laquelle chaque sous-classe hérite des données et des méthodes (plus générales) de sa (ou ses) classe(s) mère(s). " Une sous-classe est donc une spécialisation de la description d'une classe, appelée sa **superclasse**, dont elle partage les variables et les méthodes. " ([DEL-JOU], p. 28)

La transmission de message est le moyen unique de communiquer avec les objets. Un message est en fait une requête dont la satisfaction est à la charge de l'objet auquel elle est adressée ⁷. Pour être complet, un message doit toujours mentionner l'objet concerné (le **receveur** du message), le **sélecteur** (c'est-à-dire le nom) de la méthode à activer et éventuellement les **arguments** sur lesquels celle-ci s'applique.

Une opération peut accepter des arguments de différents types. Cette caractéristique est connue sous le nom de **polymorphisme**.

Le mécanisme d'**instanciation** permet, pour sa part, de définir un objet comme un individu appartenant à une classe donnée. La liste des champs est détenue par la classe, mais les instances, c'est-à-dire les représentants physiques de la classe, en possèdent les valeurs.

⁷ " Un objet étant une entité indépendante dont la structure est connue de lui seul, il ne peut pas agir directement sur un autre objet. Il doit pour ce faire utiliser une des méthodes appartenant à l'interface de cet autre objet et lui envoyer un message demandant l'exécution de la méthode en question. " ([DEL-JOU], p. 28)

Les trois premiers chapitres ont permis de délimiter le cadre de notre étude, d'évoquer dans différents domaines les fondements à la base de la démarche proposée par de Looz et Jourquin dans [DEL-JOU] et d'exposer leurs choix quant aux types d'outils qui supporteront cette démarche.

Nous pouvons dès lors passer aux chapitres IV et V présentant et adaptant leur démarche de spécification et d'implantation d'animation graphique.

Chapitre IV : METHODOLOGIE DE SPECIFICATION BASEE SUR LA NOTION DE SCENARIO

Suite aux choix posés dans les chapitres précédents, nous exposerons dans ce chapitre l'approche théorique de spécification basée sur la notion de scénario telle que proposée par de Looz et Jourquin dans [DEL-JOU]. Nous verrons alors les problèmes que nous avons rencontrés lors de l'application de cette démarche sur le jeu de rôle FRIK et proposerons une solution. Nous analyserons ensuite la possibilité d'une rationalisation de la méthode. Nous proposerons également une étape supplémentaire, à savoir la spécification par prédicats. Enfin, nous verrons l'intérêt à retirer de chacun des documents rédigés au fur et à mesure des étapes de la méthode de spécification basée sur la notion de scénario.

Mais avant cela, il est nécessaire d'exposer de nouveaux concepts dont nous aurons besoin, ainsi que le formalisme de modélisation que nous emploierons tout au long de la méthodologie.

4.1. DEFINITION DES CONCEPTS UTILISES ET D'UN FORMALISME DE REPRESENTATION DE MODELES

4.1.1. Concepts

La démarche proposée s'inspirant de la production audio-visuelle, le vocabulaire que nous utiliserons proviendra de ce domaine. Ainsi, nous parlerons d'idée originale, de synopsis, de parties, de séquences et de plans. Ces concepts ont, tous, été définis au chapitre II (cf point 2.3.1).

Cependant, afin d'éviter toute confusion avec la notion de "plan de système" qui sera définie ci-dessous, nous préférons le terme **tableau** (emprunté au domaine théâtral) pour désigner le terme "plan".

Ajoutons encore deux concepts spécifiques à la démarche : ceux de jalons et de réseau de transitions entre jalon.

Un **jalon** constitue un état particulier d'une partie du scénario à un instant donné. Il représente un point de passage obligé du scénario. Notons cependant qu'il se pourrait que certains jalons ne soient pas vérifiés à chaque exécution. Ceci permet en fait une certaine flexibilité, un certain degré de liberté dans le scénario. Ainsi, tout scénario peut être caractérisé par un certain nombre de jalons, s'enchaînant selon certaines conventions connues d'un "meneur de jeu".

Si l'on dessine une représentation graphique des possibilités d'enchaînement entre jalons, c'est-à-dire des transitions permises à partir de chaque jalon, on obtiendra le **réseau de transitions** entre les jalons du scénario. Il définit en fait la dynamique du scénario.

4.1.2. Définition d'un formalisme de représentation de modèles

Nous voudrions pouvoir visualiser l'évolution de notre système et l'enchaînement des comportements des différents acteurs au sein du scénario. A cette fin, nous souhaiterions disposer d'une méthode de conception de modèles mettant en oeuvre tous les éléments du scénario (acteurs, jalons, etc) et définissant le scénario, en décrivant l'agencement de ces éléments d'une manière dynamique. La méthode choisie par de Looz et Jourquin est la méthode E.P.A.S. (Entrée-Processus-Accumulation-Sortie) développée à l'Université Laval de Québec (Canada). La méthode E.P.A.S est une méthodologie de modélisation et de conception de système d'information. On trouvera les idées conductrices à l'annexe 1 de ce travail.

Nous allons reprendre ci-dessous certaines techniques de modélisation mises en oeuvre par cette méthode et nous intéressant au plus haut point. Ces techniques sont le cycle de vie des objets, les règles de couplages, les plans de système et les diagrammes de transition.

Le **cycle de vie des objets** est une technique permettant de visualiser la dynamique d'évolution des objets en modélisant les différents états par lesquels passent les objets. Les **règles de couplages** permettent, elles, de suivre l'évolution des objets les uns par rapport aux autres.

Les **plans de système** permettent de modéliser la stratégie de comportement d'un système.

Quant aux **diagrammes de transition**, ils permettent de rendre compte de la dynamique d'évolution du système. La dimension temporelle qu'ils véhiculent permet de modéliser le fonctionnement du système à un niveau de détail supérieur par rapport aux plans de systèmes.

Possédant tous les concepts que nous venons de voir, nous pouvons maintenant nous intéresser à la démarche de spécification basée sur la notion de scénario.

4.2. DEMARCHE DE SPECIFICATION

Nous commencerons par exposer le canevas de présentation de la démarche de spécification. Nous décrirons ensuite le jeu de rôle dans lequel nous puiserons les exemples servant à l'illustration des différentes étapes. Enfin, nous présenterons la démarche de spécification, selon le canevas de présentation convenu.

4.2.1. Canevas de présentation

Chaque étape de la spécification sera présentée selon le canevas suivant. Nous commencerons dans tous les cas par l'exposé de la démarche de spécification originale, c'est-à-dire telle qu'elle est décrite par de Looz et Jourquin dans [DEL-JOU]. Suivront alors les problèmes que nous avons rencontrés lors de l'application de l'étape considérée ainsi que les solutions que nous proposons afin de les résoudre. Enfin, nous illustrerons la démarche à l'aide d'exemples tirés de la spécification du jeu de rôle FRIK, jeu décrit au point 4.2.2 ci-dessous.

4.2.2. Description du jeu de rôle FRIK

FRIK est un jeu de rôle qui permet aux joueurs de réaliser toutes sortes d'activités se rapportant au domaine des affaires. Le but du jeu est très simple : il faut être le dernier à rester en lice tout en ayant amassé le plus possible d'argent, de propriétés et fait de bons investissements. Pour ce faire, le joueur reçoit en début de partie un

portefeuille qu'il est tenu de faire fructifier (si toutefois son objectif est de gagner !). Chaque joueur aura dû se choisir une stratégie financière afin de ne pas tomber (trop vite) en faillite. Il se verra proposer en début de partie une police d'assurance (PA) qui le prémunira contre certaines catastrophes dont il pourrait être victime. Durant la partie, il sera amené à investir dans trois grands types de placements : les placements sûrs (constitués des différents titres de propriétés et de banques), les placements à risque modérés (actions, tableaux et prêts) et la spéculation (sur le prix de l'or). Il pourra également négocier avec les autres joueurs si tel est son désir.

Nous n'allons pas entrer ici dans les détails. Le lecteur trouvera à l'annexe 2 le règlement complet de ce jeu de rôle.

4.2.3. Démarche de spécification

La démarche générale de spécification d'une animation graphique telle que décrite par de Looz et Jourquin comporte sept grandes étapes :

- 1° Description de l'idée originale ou d'un synopsis pour le scénario;
- 2° Identification des acteurs et description de leurs principales caractéristiques;
- 3° Spécification du scénario en définissant les jalons et la découpe en partie du scénario;
- 4° Révision de la description des acteurs et propositions de nouvelle découpes du scénario, et établissement du réseau de transitions entre les jalons du scénario;
- 5° Spécification des acteurs : formalisation en terme d'attributs, d'une part, et cycle de vie et règles de couplage, d'autre part;
- 6° Spécification formelle des jalons;
- 7° Pour chaque acteur, description du plan de système et du diagramme de transition.

Lors de chacune de ces étapes, des documents constituant les résultats de la spécification seront rédigés (un par étape). La rédaction de ces documents est particulièrement importante et devra être soignée car nous nous servirons de ces derniers lors de la démarche d'implantation.

Passons maintenant chacune de ces étapes en revue selon le canevas défini ci-dessus.

1° Description de l'idée originale ou d'un synopsis pour le scénario

a) Démarche de spécification originale

Lors de cette étape, on décrira l'idée originale sur laquelle s'étayera le scénario. Cependant, si l'on a déjà une vue plus précise, on veillera plutôt à exposer un bref résumé du scénario.

Cette étape ne posant aucun problème particulier, nous passerons directement à l'illustration.

b) Illustration

Le jeu automatisé se déroulera de la façon suivante :

- Le jeu est ouvert
- Les différents joueurs recevront leur argent en début de partie
- Les différents joueurs se verront proposer une PA
- Tant qu'il restera plus d'un joueur, la séquence suivante sera exécutée :
 - Un joueur lance les dés et avance du nombre de cases voulu en déclenchant éventuellement certains comportements de cases en passant par-dessus (case de prêt et case départ).
 - Une série de comportements due au type de la case sur laquelle on s'est arrêté est déclenchée.
 - Les joueurs tombés en faillite sont éliminés
 - Les joueurs peuvent négocier entre-eux
- Le jeu est fermé

2° Identification des acteurs et description de leurs principales caractéristiques

a) Démarche de spécification originale

Cette étape permet d'associer à chaque acteur identifié une liste de caractéristiques. On trouvera dans ces listes, et ce pour chaque acteur, des propositions en langage naturel indiquant leur caractère visible ou invisible, leur caractère mobile ou fixe, leur fonction, leur

comportement, la description de leur représentation graphique, ... ainsi que les différents états qui peuvent les caractériser.

Il est cependant un acteur invisible sur lequel nous allons nous pencher plus particulièrement : le meneur de jeu. Comme nous l'avons déjà mentionné au chapitre 2 (point 2.2), le meneur de jeu coordonne le comportement des autres acteurs. Pour ce faire, il connaît les caractéristiques de ces derniers. " Concrètement, son comportement consiste à mettre en place le décor, à initialiser le domaine d'animation et à déclencher le comportement des acteurs, en respectant le scénario. Ces comportements sont essentiellement de deux types : les comportements liés aux déplacements et à tout affichage à l'écran en cours d'animation, et les autres comportements visant à la conduite du scénario. " ([DEL-JOU], p. 40 et 41). Deux fonctions du meneur de jeu répondent à ces deux types de comportements, à savoir les fonctions caméra et conduire scénario.

La fonction **caméra** gère les déplacements et les affichages lors de l'animation. " La caméra est considérée comme un sous-acteur de l'acteur meneur de jeu, et donc comme un acteur à part entière. Elle gère les données relatives à l'affichage et au déplacement des acteurs à l'écran, en respectant les contraintes provenant des autres fonctions du meneur de jeu. " ([DEL-JOU], p. 41). Dans certains systèmes, la caméra peut également à avoir à s'occuper des déplacements simultanés des acteurs.

La fonction **conduire scénario** vise quant à elle à prendre les bonnes décisions au bon moment, en déclenchant les actions prévues des différents acteurs à l'instant requis. La fonction conduire scénario n'est pas un acteur mais constitue l'un des rôles de l'acteur meneur de jeu.

Cette étape ne présentant aucun problème, nous passerons directement à l'illustration.

b) Illustration

Nous illustrerons cette étape par l'identification des comportements du Grand Argentier (faisant office de meneur de jeu) et d'un type de cases, les cases Galerie d'Art. Mais pour commencer, voyons les différents acteurs que nous avons relevés dans le jeu FRIK.

Les acteurs présents dans le jeu FRIK sont:

- les joueurs
- le Grand Argentier
- les cases de Hasard
- la case de Courtage
- les cases de Prêt
- les cases de Galerie d'Art
- les cases de Bourses
- la case de Lingot d'Or
- les cases représentant des titres de propriétés, à savoir les cases de Société, d'Industrie, de Commerce, d'Immeuble ou de Banque

Le **Grand Argentier** est un acteur invisible. C'est lui qui gère la banque FRIK. Son rôle est de :

- proposer une police d'assurance à chaque joueur au début de la partie
- proposer les titres de propriétés invendus lorsqu'un joueur tombe sur une propriété non encore totalement vendue
- proposer des titres de propriétés aux enchères lorsque le joueur qui a accédé à la case correspondante ne désire pas les acheter
- payer aux joueurs les intérêts des prêts qu'ils ont consentis à des personnes fictives
- proposer des actions aux joueurs s'arrêtant sur une des cases Bourse
- proposer des tableaux (un à la fois) aux joueurs s'arrêtant sur une des cases Galerie d'Art
- déposer 100000\$ dans le Fond Boursier à chaque passage d'un joueur par la case départ
- proposer des lingots d'or (un à la fois) aux joueurs qui tombent sur la case Lingots d'Or
- présider les ventes aux enchères
- prêter de l'argent aux joueurs (en exigeant des titres de propriétés pour au moins le double du montant du prêt en garantie)
- exiger le remboursement d'un prêt auprès d'un joueur après 3 tours calculés depuis la case où se trouvait celui-ci lors de l'emprunt

Les cases **Galerie d'Art** sont des acteurs visibles et fixes. Leur rôle est de proposer au joueur qui tombe sur la case d'acheter un tableau au choix (s'il en reste) au prix mentionné sur la carte et/ou de revendre des tableaux que le joueur possédait déjà auparavant au prix également inscrit sur la carte.

J9 : Il ne reste qu'un joueur.

J10 : Le jeu est terminé.

Les jalons J4 à J8 surviennent un nombre quelconque de fois.

2° Découpe en partie du scénario :

P10 (J1 à J2) : Le MJ ouvre le jeu et distribue l'argent selon le nombre de joueurs.

P20 (J2 à J3) : Le MJ propose des polices d'assurance aux joueurs.

P30 (J3 à J4) : Le MJ désigne le joueur courant et celui-ci lance les dés.

P40 (J4 à J5) : Le joueur avance d'autant de cases que de points obtenus au lancé de dés. La case déclenche les comportements qui lui sont associés.

P50 (J5 à J6) : Le MJ élimine éventuellement les personnes qui sont en faillite.

P60 (J6 à J7) : Les joueurs négocient éventuellement avec les autres joueurs.

P70 (J7 à J8) : Les joueurs manipulent éventuellement leur(s) série(s).

P80 (J8 à J9) : Le MJ constate qu'il ne reste qu'un joueur.

P90 (J9 à J10) : Le MJ ferme le jeu.

4° Révision de la description des acteurs et propositions de nouvelles découpes du scénario, et établissement du réseau de transitions entre les jalons du scénario

a) Démarche de spécification originale

" Cette quatrième étape comporte deux sous-étapes distinctes. La première consiste en une suite de raffinements successifs du scénario, en précisant certaines caractéristiques des acteurs, de nouveaux jalons et de nouvelles découpes. (...). La seconde sous-étape consiste en l'élaboration de la dynamique du scénario (réseau de transition). " ([DEL-JOU], p. 48)

Suite à la définition des jalons et à la découpe en partie du scénario (étape 3), de nouvelles caractéristiques des acteurs peuvent apparaître. Celles-ci peuvent à leur tour induire la définition de nouveaux (sous-) jalons ainsi qu'une nouvelle découpe du scénario,

appelée découpe en **séquences**, détaillant chacune des opérations qui constituent les parties du scénario. De la même manière, ces nouveaux sous-jalons et cette nouvelle découpe du scénario peuvent faire apparaître d'autres caractéristiques des acteurs. Celles-ci peuvent également entraîner la définition de nouveaux sous-jalons et une nouvelle découpe du scénario appelée cette fois découpe en **tableaux**. Cette découpe a pour objectif d'affiner la découpe en séquences du scénario. Notons que la découpe en tableaux du scénario constitue le niveau de découpe le plus détaillé. Un tableau décrit en fait une visualisation particulière du domaine d'animation.

" Cette première sous-étape se formalise par :

- la première liste des caractéristiques supplémentaires des acteurs,
- la première liste des jalons supplémentaires,
- la découpe en séquences (présentée de la même manière que la découpe en parties),
- la seconde liste des caractéristiques supplémentaires des acteurs,
- la seconde liste des jalons supplémentaires,
- la découpe en tableaux (présentée de la même manière que la découpe en parties). " ([DEL-JOU], p.49)

Quant au réseau de transition entre les jalons du scénario, il représente en fait les différents cheminements possibles à travers le scénario. Il se situe au niveau le plus fin du scénario, celui de la découpe en tableaux. " Les jalons y sont représentés par des cercles et les tableaux par des flèches reliant les jalons. Le réseau de transitions met en évidence le fait qu'un jalon peut être à la fois situation initiale et situation finale de plusieurs tableaux. Les différents chemins proposés dans ce réseau indiquent les différentes possibilités de scénario. " ([DEL-JOU], p.49 et 50)

b) Problèmes rencontrés

Un des premiers problèmes apparu lors de l'application de cette étape est la difficulté d'exprimer les alternatives. Voyons cela sur un petit exemple. Soient les jalons suivants :

- J1 : Le nombre x est connu
- J2 : Si $x > 20$, l'étudiant a reçu une note d'examen A
- J3 : Si $10 \leq x \leq 20$, l'étudiant a reçu une note d'examen B
- J4 : Si $x < 10$, l'étudiant a reçu une note d'examen C
- J5 : La note de l'étudiant a été attribuée
- J6 : La note de l'étudiant a été imprimée

Cela donnera lieu à la découpe en tableaux suivante :

- T1 (J1 à J2) : Si $x > 20$, donner la note A à l'étudiant
- T2 (J1 à J3) : Si $10 \leq x \leq 20$, donner la note B à l'étudiant
- T3 (J1 à J4) : Si $x < 10$, donner la note C à l'étudiant
- T4 (J2 à J5) : Fin de l'alternative A
- T5 (J3 à J5) : Fin de l'alternative B
- T6 (J4 à J5) : Fin de l'alternative C
- T7 (J5 à J6) : Impression de la note de l'étudiant

Nous nous trouvons donc devant le problème de l'apparition de trois tableaux "bidons" ne servant en fait qu'à nous mener vers le jalon J5, jalon représentant un point de passage obligé du scénario. Pour un petit système tel que celui soulevé dans l'exemple ci-dessus, le problème n'est pas réellement une grosse difficulté car le concepteur connaît (certainement) tout son scénario. Par contre, dans le système que nous nous proposons de spécifier, l'introduction de tableaux "bidons" implique un problème de lisibilité du scénario. En outre, si l'on considère des systèmes mettant en oeuvre une multitude d'acteurs et dont la complexité et l'ampleur sont telles qu'elles requièrent plusieurs concepteurs, il est indispensable de notifier à ceux qui ne se sont pas occupés d'une partie du scénario la nature "fin d'alternative" de certains tableaux.

Si la méthode convient tout à fait pour un scénario séquentiel (c'est-à-dire un scénario dans lequel l'utilisateur n'est autorisé à intervenir qu'en certains endroits bien définis et à la requête du meneur de jeu ou de tout autre acteur autorisé à interagir avec lui), cela n'en est pas de même pour un scénario interactif (c'est-à-dire un scénario dans lequel un des utilisateurs peut demander l'exécution immédiate d'un processus) tel que celui que nous développons. Dans notre système

apparaissent en effet quelques comportements qui doivent pouvoir être activés à divers endroits du scénario. Ainsi, par exemple, un joueur peut réclamer le prêt qu'il a effectué à n'importe quel moment lors de son tour de jeu (pour autant qu'il remplisse les conditions requises bien entendu). De telles situations proviennent dans notre cas du fait qu'il s'agit d'une possibilité offerte au joueur et non une obligation.

c) Solutions proposées

Pour notre premier problème, il n'est pas possible de faire disparaître complètement les tableaux "bidons" introduits suite à une alternative. Ils sont en effet nécessaires à une bonne définition du réseau de transition entre les jalons du scénario, afin que celui-ci représente tous les cheminements possibles à travers le scénario. Nous proposerons donc de représenter les tableaux concernés par des flèches discontinues dans le réseau de transitions entre les jalons du scénario, indiquant ainsi leur caractère "fin d'alternative", de sorte qu'un simple coup d'œil nous informe sur la nature de chaque tableau. Mais nous voudrions en outre que cette distinction importante puisse également être répertoriée dans le scénario lui-même. Ainsi, nous pourrions faire précéder la définition même des tableaux par le terme "Fin d'alternative", comme dans l'exemple précédent. Nous préférons toutefois l'utilisation d'un signe distinctif, beaucoup plus court, et représentant le caractère "Fin d'alternative du tableau. Nous proposons donc de souligner la numérotation de ces tableaux dans la découpe du scénario en tableaux.

La découpe en tableaux de notre exemple devient de cette manière :

T1 (J1 à J2) : Si $x > 20$, donner la note A à l'étudiant

T2 (J1 à J3) : Si $10 \leq x \leq 20$, donner la note B à l'étudiant

T3 (J1 à J4) : Si $x < 10$, donner la note C à l'étudiant

I4 (J2 à J5) : Fin de l'alternative A

I5 (J3 à J5) : Fin de l'alternative B

I6 (J4 à J5) : Fin de l'alternative C

T7 (J5 à J6) : Impression de la note de l'étudiant

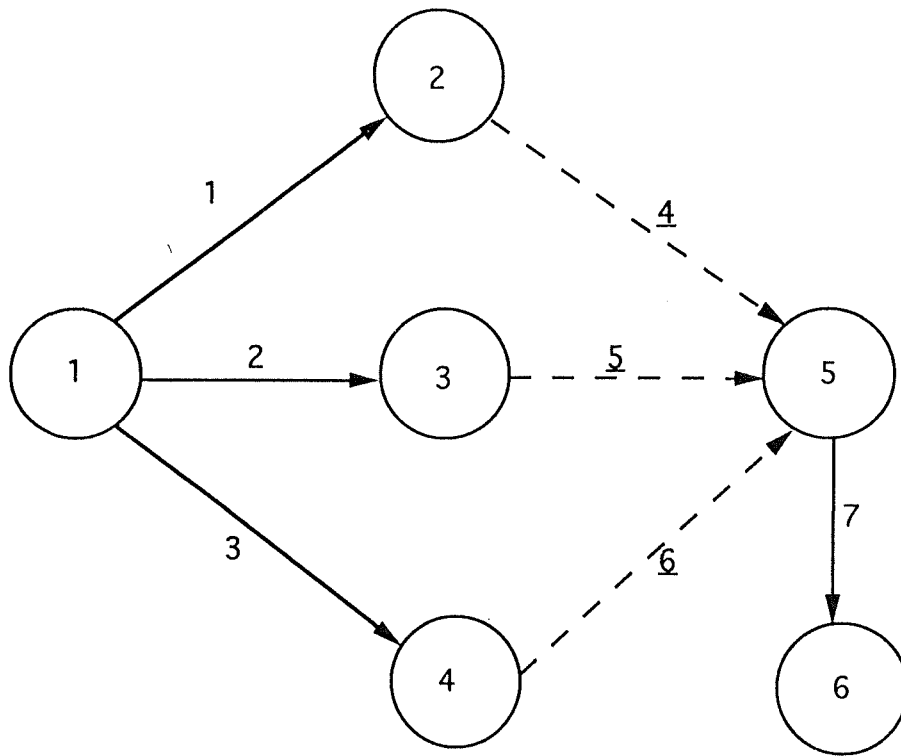


Figure IV.1 - Réseau de transition entre les jalons de l'exemple.

et le réseau de transition entre les jalons du scénario se dessine comme montré à la figure IV.1.

Pour notre second problème, il serait impensable d'introduire en tout point du scénario où cela s'avère nécessaire la possibilité d'activer l'exécution d'une série de tableaux sans alourdir considérablement la lisibilité (et l'ampleur) du dit scénario. D'autre part, il ne faut pas qu'une telle série puisse être développée (plusieurs fois) par des concepteurs différents (cela pourrait être le cas pour de grands systèmes). Il serait donc souhaitable d'introduire un nouveau concept permettant l'utilisation de comportements activables de façon immédiate et indiquant les différents endroits du scénario où ces comportements peuvent être activés. C'est pourquoi nous avons décidé d'introduire le nouveau concept de "jalon étoile" (*). Il s'agit en fait de jalons à partir desquels une série de tableaux correspondant à cette étoile pourront être exécutés. Après exécution des tableaux, on rejoint les jalons initiaux. Ainsi, à chaque jalon du scénario où va apparaître le symbole *i (avec $i > 0$), une séquence i de tableaux pourra être exécutée, si toutefois les conditions requises à son activation sont réunies. D'un point de vue pratique, cette notion de jalon étoile apparaîtra dans le réseau de transition entre les jalons du scénario. La séquence correspondant à ce jalon sera quant à elle développée selon les mêmes conventions que celles employées pour la spécification du scénario principal. Il est donc important de ne pas oublier de développer cette séquence et d'en noter le résultat après le développement total du scénario principal. On obtiendra également un "mini" ⁸ réseau de transition entre les jalons pour chaque "jalon étoile" *i et indiquant, comme pour le scénario principal, les différents cheminements possibles à travers le "mini" scénario.

Enfin, nous proposons d'élargir les listes des caractéristiques supplémentaires des acteurs à des listes, plus complètes, comprenant

⁸ Nous considérons ici que la séquence de tableaux est relativement courte par rapport au scénario principal. Il se pourrait cependant que cette séquence soit très longue et donne, pour cette raison, un réseau de transition entre les jalons bien plus ambitieux.

en outre des acteurs supplémentaires, ainsi que leur caractéristiques, acteurs qui n'avaient pas encore été pris en compte pour la spécification soit par oubli, soit par suite d'un non repérage préalable comme véritable acteur.

d) Illustration

- Première liste des acteurs et caractéristiques des acteurs supplémentaires :

* *Le jeu* : acteur fixe et visible. Il est le cadre de l'animation graphique. Il n'a aucun comportement et est caractérisé par son état "ouvert" ou "fermé" et par le nombre de joueurs.

* *Le joueur* : il est caractérisé par 3 états : "en attente" lorsqu'il n'est pas le joueur courant, "courant" lorsqu'il est le joueur courant, "en faillite" lorsqu'il est éliminé pour faillite.

* *Les cartes de police d'assurance* : elles sont des acteurs visibles. Elles sont caractérisées par l'état "vendue" ou "non vendue".

* *Les dés* : ce sont des acteurs fixes et visibles. Ils sont caractérisés par 7 états : "en lancement", "1", "2", "3", "4", "5" ou "6".

* *Les séries* : ce sont des acteurs (in)visibles. Elles sont caractérisées par 3 états : "disponibles" si les joueurs ont le droit de les modifier, "en cours" si les joueurs sont en train de les modifier et "complétée" quand plus aucun des joueurs ne désire plus modifier ses séries. A partir de ce moment, aucun joueur ne peut plus toucher à une de ses séries.

De cette première liste des acteurs et caractéristiques des acteurs supplémentaires, nous pouvons déduire des sous-jalons supplémentaires.

- Première liste des jalons supplémentaires :

P10 : J1 : Le jeu n'est pas commencé.

J1.1 : Le nombre de joueur est connu.

J1.2 : Le MJ a distribué l'argent aux joueurs.

J.2 : Le jeu est commencé et chaque joueur a reçu une somme d'argent.

P20 : J2 : Le jeu est commencé et chaque joueur a reçu une somme d'argent.

J2.1 : Le MJ a proposé à un joueur d'acheter une police d'assurance (PA).

J2.2 : Le joueur a répondu à l'offre du MJ d'acheter une PA.

J2.3 : Le joueur a éventuellement payé et reçu sa PA.

J3 : Le MJ a proposé des PA à tous les joueurs.

P30 : J3 : Le MJ a proposé des PA à tous les joueurs.

J3.1 : Le MJ a désigné le joueur courant.

J4 : Le joueur a lancé les dés.

P40 : J4 : Le joueur a lancé les dés.

J4.1 : Le joueur a avancé du nombre de cases désigné par les dés en déclenchant des comportements éventuels de cases en passant par dessus.

J5 : Le joueur est tombé sur une case et a éventuellement effectué ce qu'il pouvait ou devait faire.

P60 : J6 : Le MJ a éventuellement éliminé un joueur en faillite.

J6.1 : Un joueur a éventuellement fait une offre de vente, d'achat ou d'échange à un autre joueur.

J6.2 : L'autre joueur a soit accepté, soit refusé l'offre ou le joueur n'a pas fait d'offre.

J6.3 : Plus aucun joueur n'a fait de proposition de vente, d'achat ou d'échange.

J7 : Les joueurs ont éventuellement négocié.

P70 : J7 : Les joueurs ont éventuellement négocié.

J7.1 : Un joueur a éventuellement retiré une ou des cartes d'une ou de plusieurs de ses séries.

J7.2 : Un joueur a éventuellement placé un ou des titres de propriété dans une série.

J7.3 : Plus aucun joueur n'a désiré faire de manipulations

J8 : Les joueurs ont éventuellement manipulé leurs séries.

A l'aide de cette première liste des jalons supplémentaires, nous définissons une nouvelle découpe du scénario.

- Découpe du scénario en séquences :

S10 (J1 à J1.1) : Le MJ demande le nombre de joueurs.

S20 (J1.1 à J1.2) : Le MJ distribue l'argent aux joueurs.

S30 (J1.2 à J2) : Le MJ ouvre le jeu.

S40 (J2 à J2.1) : Le MJ propose à un joueur d'acheter une PA.

S50 (J2.1 à J2.2) : Le joueur répond à l'offre du MJ d'acheter une PA.

S60 (J2.2 à J2.3) : Eventuellement, le joueur paye et reçoit sa PA.

S70 (J2.3 à J2) : Si le MJ n'a pas encore proposé une PA à tout le monde, il la propose au joueur suivant.

S80 (J2.3 à J3) : Cas contraire à la séquence 70.

S90 (J3 à J3.1) : Le MJ désigne le joueur courant.

S100 (J3.1 à J4) : Le joueur courant lance les dés.

S110 (J4 à J4.1) : Le joueur avance d'autant de cases que le nombre obtenu au lancé de dés et déclenche le comportement éventuel d'une case en passant par dessus.

S120 (J4.1 à J4.2) : La case déclenche les comportements et le joueur exécute éventuellement ce qu'il peut ou doit faire.

S140 (J5 à J6) : Le MJ élimine éventuellement les joueurs en faillite.

S150 (J6 à J6.1) : Eventuellement, un joueur fait une offre de vente, d'achat ou d'échange d'un bien ou privilège à un autre joueur.

S160 (J6.1 à J6.2) : L'autre joueur refuse ou accepte l'offre.

S162 (J6.2 à J6) : Un joueur fait à nouveau une offre.

S163 (J6.2 à J6.3) : Plus aucun joueur ne fait d'offre.

S164 (J6.3 à J7) : Fin des négociations.

S170 (J6 à J6.2) : Pas d'offre d'achat, de vente ou d'échange.

S180 (J7 à J7.1) : Un joueur retire un ou des titres d'une ou de plusieurs de ses séries.

S185 (J7.1 à J7) : Un joueur désire manipuler une de ses séries.

S190 (J7.1 à J7.2) : Un joueur place un ou des titres dans une série.

S200 (J7.1 à J7.3) : Plus aucun joueur ne désire faire de manipulation.

S210 (J7.2 à J7.3) : Plus aucun joueur ne désire faire de manipulation.

S220 (J7 à J7.2) : Un joueur place un ou des titres dans une série (sans en retirer).

S225 (J7.2 à J7) : Un joueur désire manipuler une de ses séries.

S228 (J7.3 à J8) : Fin des manipulations de séries.

S230 (J8 à J9) : Le MJ constate qu'il ne reste qu'un joueur.

S240 (J9 à J10) : Le MJ ferme le jeu.

Nous allons maintenant nous intéresser à d'autres acteurs et caractéristiques des acteurs supplémentaires.

• Deuxième liste des acteurs et caractéristiques des acteurs supplémentaires :

* *Les cases de hasard* : elles doivent gérer l'ensemble des cartes de hasard.

* *Les cartes de hasard* : ce sont des acteurs visibles. Elles sont caractérisées par l'état "catastrophe" ou "privilège".

* *La case de courtage* : elle est caractérisée par le montant qu'elle contient.

* *Les cases de prêt* : elles sont caractérisées par les identifiants des joueurs qui ont prêté (ou à qui le prêt a été cédé) et par l'état "autorisation de réclamer" si le joueur vient juste de s'arrêter ou de passer au-dessus de la case de prêt et "interdiction de réclamer" dans le cas contraire.

* *Les cases galerie d'art* : elles doivent gérer l'ensemble des cartes tableaux invendues.

* *Les cartes de tableaux* : ce sont des acteurs visibles caractérisés par l'état "invendue", "vendue" ou "retirée du jeu".

* *Les cases de bourses* : elles doivent gérer l'ensemble des actions.

* *Les cartes d'actions* : ce sont des acteurs visibles. Elles sont caractérisées par l'état "vendue" ou "invendue".

* *La case fonds boursier* : c'est un acteur visible et fixe. Elle est caractérisée par le montant qu'elle contient.

* *La case lingot d'or* : elle doit gérer l'ensemble des lingots d'or.

* *Les lingots d'or* : ce sont des acteurs visibles. Ils sont caractérisés par l'état "vendu" ou "invendu".

* *Les cases de titres de propriétés* : elles gèrent l'ensemble des cartes de propriété se rapportant à leur case. Chaque case est caractérisée par l'état "libre" si aucune partie de la propriété n'est déjà vendue, "semi-libre" si une partie (non entière) de la propriété est libre et "occupée" lorsque l'entièreté des titres de la propriété est vendue.

* *Les cartes de titres de propriétés* : ce sont des acteurs visibles caractérisés par un pourcentage du titre de propriété qu'ils représentent et par l'état "vendu" ou "invendu" ainsi que par l'appartenance à une série qui double, quadruple ou sextuple le revenu du titre.

* *Les cartes de banques* : ce sont des acteurs visibles caractérisés par l'état "vendu" ou "invendu" ainsi que par l'appartenance de plusieurs titres de la même banque à un même joueur (ce qui permet de doubler ou de tripler la somme des revenus des titres).

De cette liste, nous déduisons des sous-jalons supplémentaires.

- Deuxième liste des jalons supplémentaires :

S110 : J4 : Le joueur a lancé les dés.

J4.0.1 : Le joueur a avancé d'une case.

J4.0.2 : Si la case est une case de prêt et que le joueur a prêté de l'argent sur cette case lors d'un tour précédent, la case est passée à l'état "autorisation de réclamer" pour ce joueur.

J4.0.3 : Si la case est la case de départ, le MJ a déposé 100000\$ dans le fonds boursier et le joueur a reçu 5% du fonds boursier par carte action qu'il possède.

J4.1 : Le joueur a avancé du nombre de cases désigné par les dés en déclenchant des comportements éventuels de cases en passant par-dessus.

S120 : J4.1 : Le joueur a avancé du nombre de cases désigné par les dés en déclenchant des comportements éventuels de cases en passant par-dessus.

J4.1.1 : S'il s'agit d'une case de hasard, le MJ a donné une carte.

J4.1.2 : Si la carte de hasard est une carte "catastrophe", le joueur a payé la somme.

J4.1.3 : Si la carte de hasard est une carte "privilège", le joueur en a éventuellement profité.

J4.1.4 : Si la case est la case de courtage, le joueur a reçu la somme contenue dans cette case.

J4.1.5 : Si la case est une case de prêt, le joueur a éventuellement réclaté son argent et ses intérêts.

J4.1.6 : Si la case est une case de prêt, le joueur a éventuellement prêté de l'argent.

J4.1.7 : Si la case est une case galerie d'art, le joueur a éventuellement vendu un ou des tableaux.

J4.1.8 : Si la case est une case galerie d'art, le joueur a éventuellement acheté un tableau.

J4.1.9 : Si la case est une case bourse, le joueur a éventuellement vendu des actions.

J4.1.10 : Si la case est une case bourse, le joueur a éventuellement acheté des actions.

- J4.1.11 : Si la case est une case lingot d'or, le joueur a éventuellement acheté un lingot d'or.
- J4.1.12 : Si la case est une case avec titres de propriété dont l'état est "semi-libre", le joueur a éventuellement payé le revenu du titre à un autre joueur (si aucune partie de la propriété ne lui appartient).
- J4.1.13 : Si la case est une case avec titres de propriété dont l'état est "occupée", le joueur a éventuellement payé le revenu du titre au(x) joueur(s).
- J4.1.14 : Si la case est une case avec titres de propriété dont l'état est "semi-libre" ou "libre", le MJ a proposé au joueur d'en acheter une partie.
- J4.1.15 : Si la case est une case avec titres de propriété dont l'état est "semi-libre" ou "libre", le joueur a éventuellement acheté le titre.
- J4.1.16 : Si la case est une case avec titres de propriété dont l'état est "semi-libre" ou "libre", le MJ propose éventuellement le titre aux enchères.
- J4.1.17 : Le titre de propriété a été éventuellement vendu aux enchères.
- J5 : Le joueur est tombé sur une case et a éventuellement effectué ce qu'il pouvait ou devait faire.

S150 : J6 : Le MJ a éventuellement éliminé un joueur en faillite.

- J6.0.1 : Un joueur a éventuellement fait une offre de vente d'un bien ou privilège qu'il détient.
- J6.0.2 : Un joueur a éventuellement fait une offre d'achat d'un bien ou privilège à un autre joueur.
- J6.0.3 : Un joueur a éventuellement fait une offre d'échange d'un bien ou privilège à un autre joueur.
- J6.1 : Un joueur a éventuellement fait une offre de vente, d'achat ou d'échange à un autre joueur.

Nous aboutissons enfin à la dernière découpe du scénario, dérivée de la deuxième liste des jalons supplémentaires : la découpe en tableaux.

- Découpe du scénario en tableaux :

- T10 (J1 à J1.1) : Le MJ demande le nombre de joueurs.
- T20 (J1.1 à J1.2) : Le MJ distribue l'argent aux joueurs.

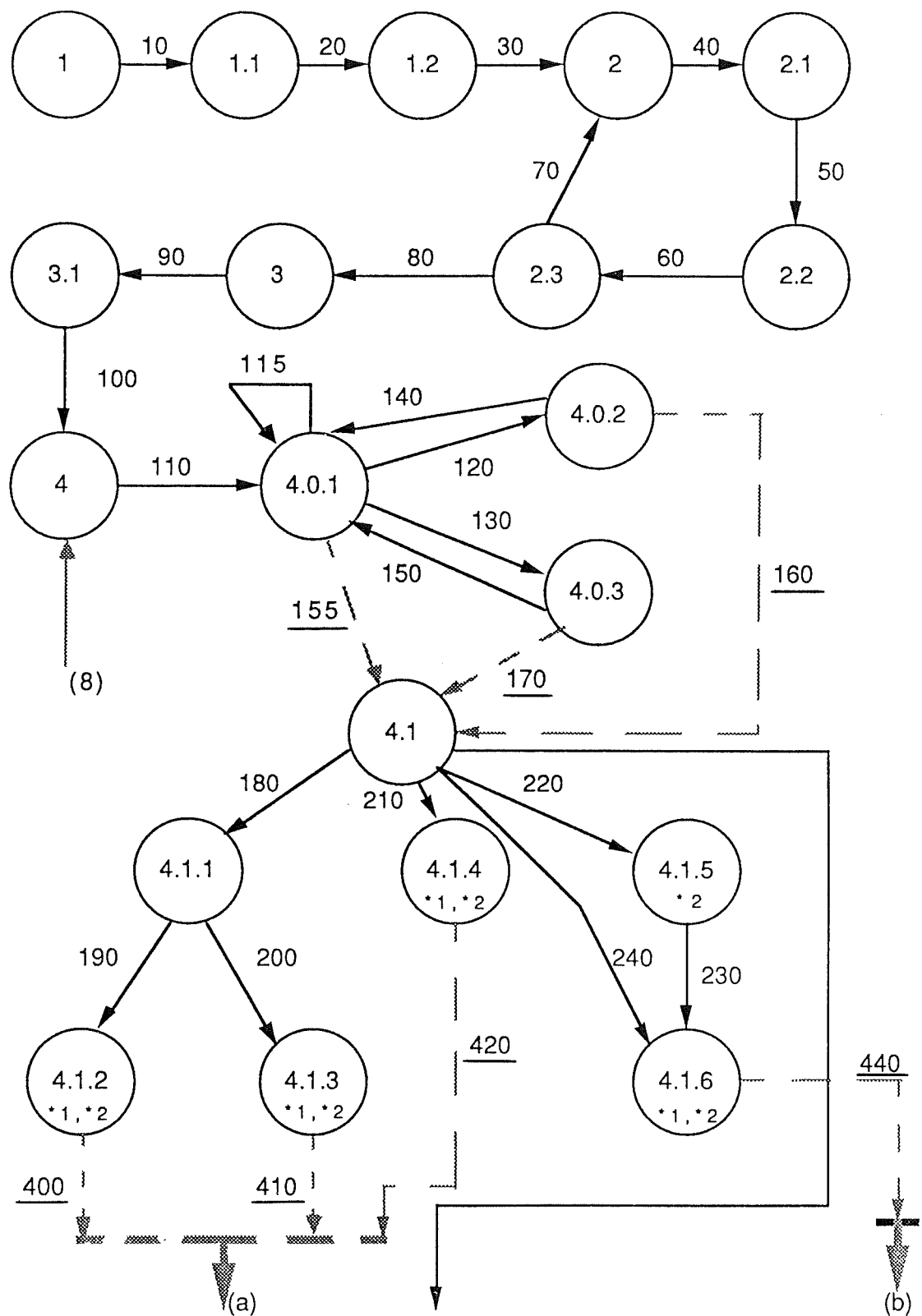


Figure IV.2 (a) - Réseau de transition entre les jalons du scénario du jeu de rôle FRIK.

- T30 (J1.2 à J2) : Le MJ ouvre le jeu.
- T40 (J2 à J2.1) : Le MJ propose à un joueur d'acheter une police d'assurance (PA).
- T50 (J2.1 à J2.2) : Le joueur répond à l'offre du MJ d'acheter une PA.
- T60 (J2.2 à J2.3) : Eventuellement, le joueur paye et reçoit sa PA.
- T70 (J2.3 à J2) : Si le MJ n'a pas encore proposé à tout le monde une PA, il la propose au joueur suivant.
- T80 (J2.3 à J3) : Fin de proposition de PA.
- T90 (J3 à J3.1) : Le MJ désigne le joueur courant.
- T100 (J3.1 à J4) : Le joueur courant lance les dés.
- T110 (J4 à J4.0.1) : Le joueur avance d'une case.
- T115 (J4.0.1 à J4.0.1) : Si la case n'est pas une case de prêt ou de départ et que le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
- T120 (J4.0.1 à J4.0.2) : Si la case est une case de prêt et que le joueur a prêté sur cette case, l'état de la case passe à "autorisation de réclamer" pour ce joueur.
- T130 (J4.0.1 à J4.0.3) : Si la case est la case de départ, le MJ dépose 100000\$ dans le fonds boursier et le joueur reçoit 5% du fonds boursier par carte action qu'il possède.
- T140 (J4.0.2 à J4.0.1) : Si le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
- T150 (J4.0.3 à J4.0.1) : Si le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
- T155 (J4.0.1 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
- T160 (J4.0.2 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
- T170 (J4.0.3 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
- T180 (J4.1 à J4.1.1) : S'il s'agit d'une case de hasard, donner une carte.
- T190 (J4.1.1 à J4.1.2) : S'il s'agit d'une carte de hasard "catastrophe", le joueur paye la somme, ainsi que les copropriétaires.
- T200 (J4.1.1 à J4.1.3) : S'il s'agit d'une carte de hasard "privilège", le joueur en profite ou la garde.
- T210 (J4.1 à J4.1.4) : S'il s'agit d'une case de courtage, le joueur empoche la somme contenue dans la case.

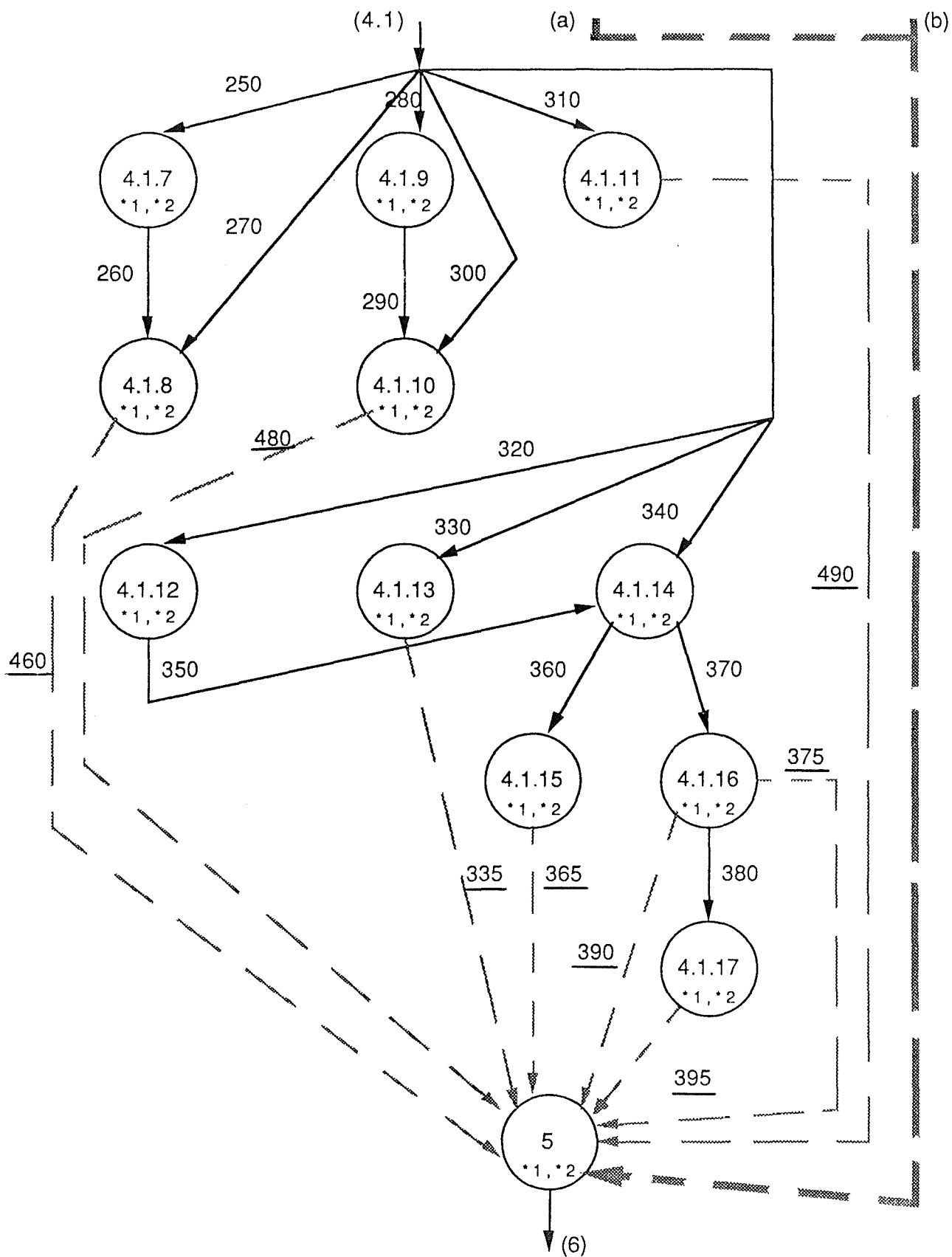


Figure IV.2 (b) - Réseau de transition entre les jalons du scénario du jeu de rôle FRIK.

- T220 (J4.1 à J4.1.5) : S'il s'agit d'une case de prêt et que le joueur a prêté, il réclame éventuellement son argent et ses intérêts.
- T230 (J4.1.5 à J4.1.6) : S'il s'agit d'une case de prêt, le joueur prête éventuellement de l'argent.
- T240 (J4.1 à J4.1.6) : S'il s'agit d'une case de prêt, le joueur prête éventuellement de l'argent (sans avoir réclamé auparavant)
- T250 (J4.1 à J4.1.7) : S'il s'agit d'une case galerie d'art, le joueur vend éventuellement un ou des tableaux.
- T260 (J4.1.7 à J4.1.8) : S'il s'agit d'une case galerie d'art, le joueur achète éventuellement un tableau.
- T270 (J4.1 à J4.1.8) : S'il s'agit d'une case galerie d'art, le joueur achète un tableau (sans en vendre car il n'en a pas).
- T280 (J4.1 à J4.1.9) : S'il s'agit d'une case bourse, le joueur revend éventuellement des actions.
- T290 (J4.1.9 à J4.1.10) : S'il s'agit d'une case bourse, le joueur achète éventuellement des actions.
- T300 (J4.1 à J4.1.10) : S'il s'agit d'une case bourse, le joueur achète des actions (sans en vendre car il n'en a pas).
- T310 (J4.1 à J4.1.11) : S'il s'agit d'une case lingot d'or, le joueur achète éventuellement un lingot d'or.
- T320 (J4.1 à J4.1.12) : S'il s'agit d'une case de propriété avec état = "semi-libre", le joueur paye le revenu du titre à un autre joueur si aucune partie ne lui appartient.
- T330 (J4.1 à J4.1.13) : S'il s'agit d'une case de propriété avec état = "occupée", le joueur paye le revenu du titre à un autre joueur si aucune partie ne lui appartient.
- T335 (J4.1.13 à J5) : Fin si la case est "occupée".
- T340 (J4.1 à J4.1.14) : S'il s'agit d'une case de propriété avec état = "libre", le MJ propose au joueur d'acheter la partie la plus importante.
- T350 (J4.1.12 à J4.1.14) : S'il s'agit d'une case de propriété avec état = "semi-libre", le MJ propose au joueur d'acheter la partie la plus importante qui reste.
- T360 (J4.1.14 à J4.1.15) : S'il s'agit d'une case de propriété avec état = "libre" ou "semi-libre", le joueur achète le titre.
- T365 (J4.1.15 à J5) : Fin de l'achat du titre par le joueur.

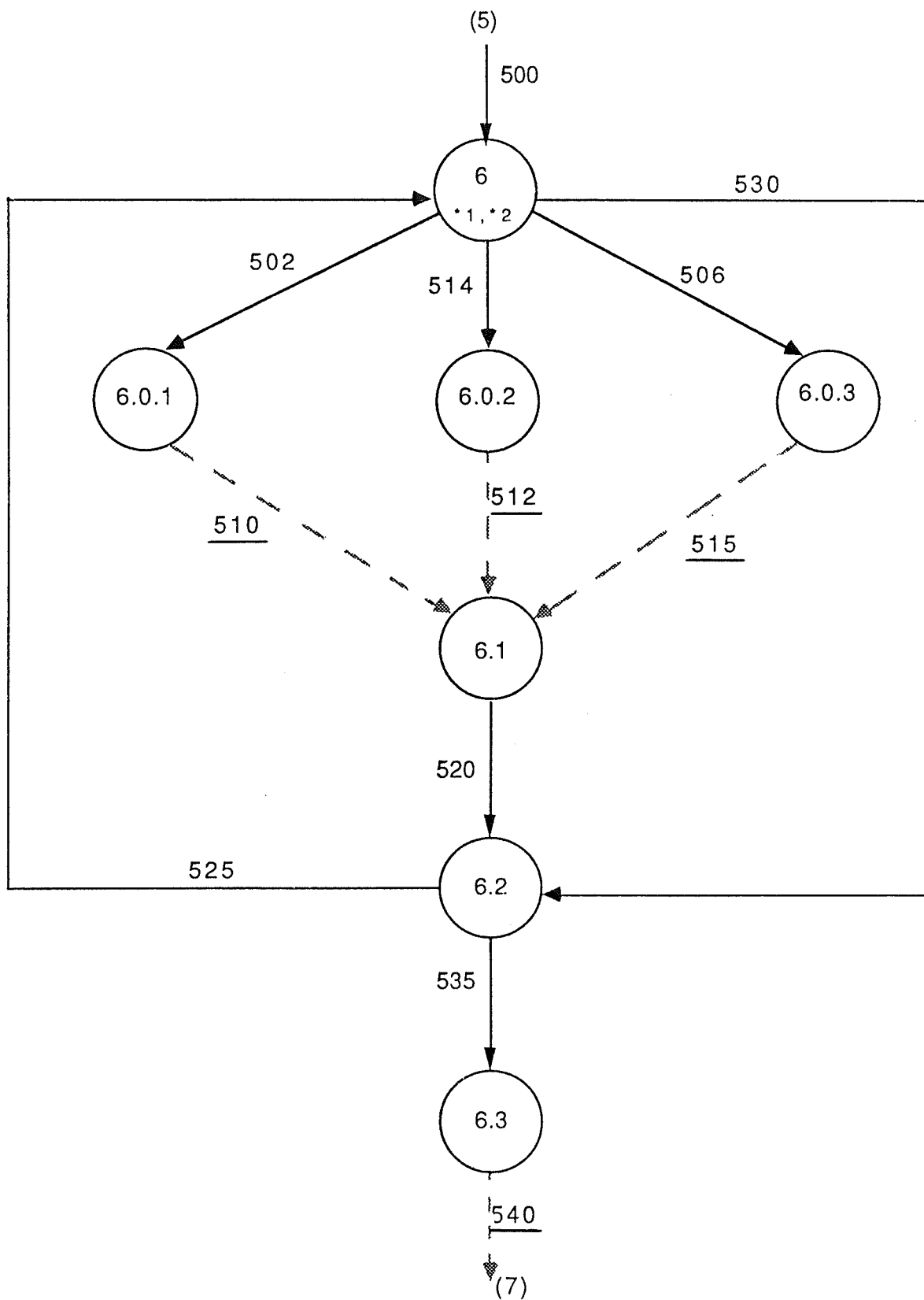


Figure IV.2 (c) - Réseau de transition entre les jalons du scénario du jeu de rôle FRIK.

- T370 (J4.1.14 à J4.1.16) : Cas contraire à 360, le MJ propose le titre aux enchères.
- T375 (J4.1.16 à J5) : Le titre n'intéresse personne et n'est pas vendu.
- T380 (J4.1.16 à J4.1.17) : Le titre est vendu aux enchères.
- T390 (J4.1.16 à J5) : Le titre n'est pas vendu aux enchères.
- T395 (J4.1.17 à J5) : Fin de la vente aux enchères.
- T400 (J4.1.2 à J5) : Fin du comportement de hasard "catastrophe".
- T410 (J4.1.3 à J5) : Fin du comportement de hasard "privilège".
- T420 (J4.1.4 à J5) : Fin du comportement de courtage.
- T440 (J4.1.6 à J 5) : Fin d'une réclamation de prêt éventuelle suivie d'un prêt.
- T460 (J4.1.8 à J5) : Fin d'une vente éventuelle d'un tableau suivie d'un achat.
- T480 (J4.1.10 à J5) : Fin d'un achat éventuel d'actions suivie d'une vente d'actions.
- T490 (J4.1.11 à J5) : Fin d'un achat de lingot d'or.
- T500 (J5 à J6) : Le MJ élimine éventuellement les personnes en faillite.
- T502 (J6 à J6.0.1) : Eventuellement, le joueur fait une offre de vente d'un bien ou privilège qu'il détient.
- T504 (J6 à J6.0.2) : Eventuellement, le joueur fait une offre d'achat d'un bien ou privilège à un autre joueur.
- T506 (J6 à J6.0.3) : Eventuellement, le joueur fait une offre d'échange d'un bien ou privilège.
- T510 (J6.0.1 à J6.1) : Fin d'offre de vente.
- T512 (J6.0.2 à J6.1) : Fin d'offre d'achat.
- T515 (J6.0.3 à J6.1) : Fin d'offre d'échange.
- T520 (J6.1 à J6.2) : Eventuellement, l'autre joueur accepte ou refuse l'offre.
- T525 (J6.2 à J6) : Un joueur fait à nouveau une offre.
- T530 (J6 à J6.2) : Le joueur ne fait pas d'offre de vente, d'achat ou d'échange.
- T535 (J6.2 à J6.3) : Plus aucun joueur ne fait d'offre.
- T540 (J6.3 à J7) : Fin des négociations.
- T550 (J7 à J7.1) : Un joueur retire un ou des titres d'une ou de plusieurs de ses séries.
- T555 (J7.1 à J7) : Un joueur désire manipuler une de ses séries.
- T560 (J7.1 à J7.2) : Un joueur place un ou des titres de propriétés dans une série.

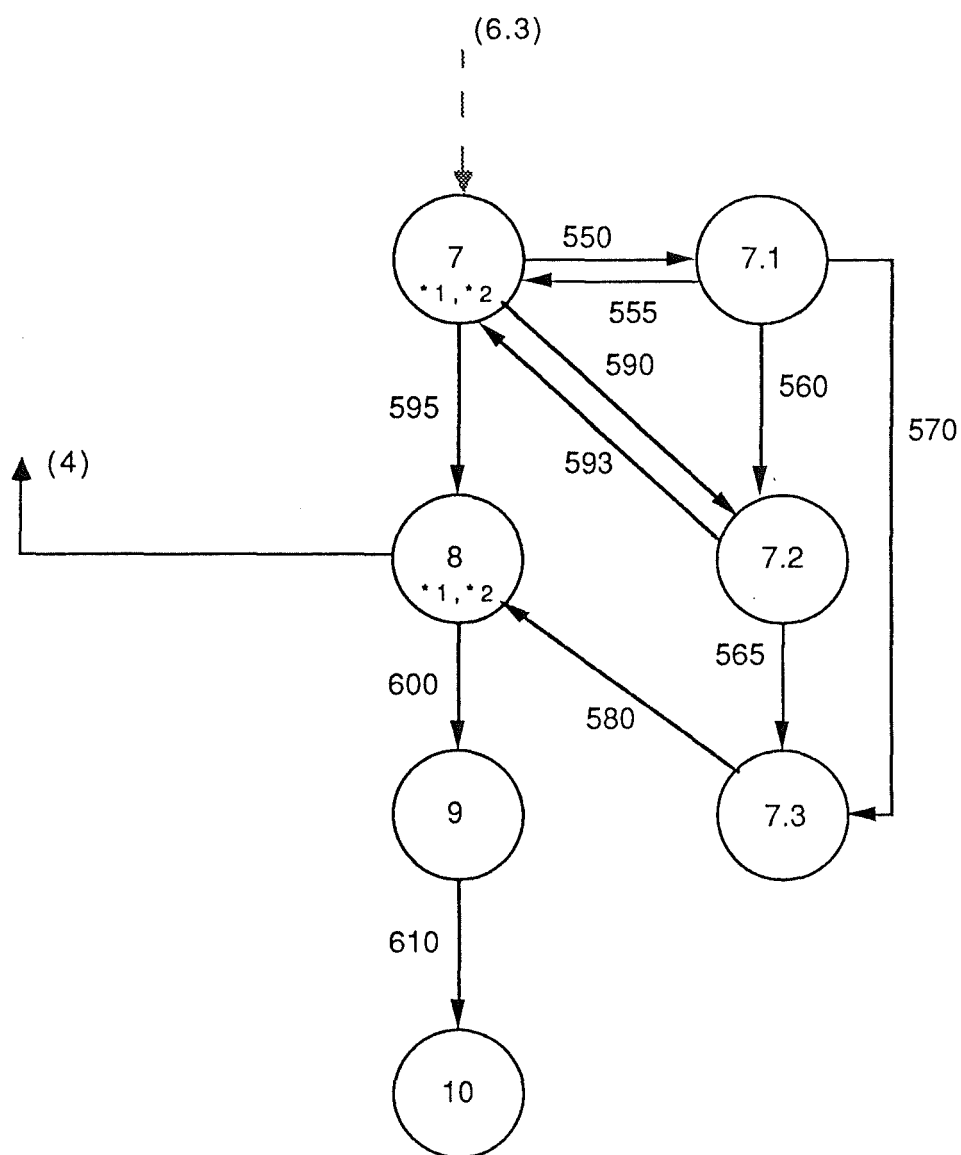


Figure IV.2 (d) - Réseau de transition entre les jalons du scénario du jeu de rôle FRIK.

T565(J7.2	à J7.3	) : Plus aucun joueur ne désire faire de manipulation.
T570 (J7.1	à J7.3	) : Plus aucun joueur ne désire faire de manipulation.
T580 (J7.3	à J8	) : Un joueur termine sa manipulation.
T590 (J 7	à J7.2	) : Un joueur place un ou des titres dans une série.
T593 (J7.2	à J7	) : Un joueur désire manipuler une de ses séries.
T595 (J7	à J8	) : Aucun joueur ne manipule de séries.
T600 (J8	à J9	) : Le MJ constate qu'il ne reste qu'un joueur.
T605 (J8	à J4	) : Il reste plus d'un joueur et le joueur suivant lance les dés.
T610 (J9	à J10	) : Le MJ ferme le jeu.

• Réseau de transition entre les jalons :

On trouvera à la figure IV.2 le réseau de transition entre jalons tel qu'il apparaît à la fin de cette étape.

On remarquera que des "jalons étoiles" *1 et *2 ont été introduits, ceux-ci correspondant respectivement au remboursement d'un prêt et à la vente d'un lingot d'or. Le développement de ces mini-scénarios se trouve ci-dessous. On trouvera à la figure IV.3 leur réseau de transition respectifs.

*1 : (Remboursement d'un prêt)

J*100 : Le joueur courant a demandé le remboursement de son prêt.

J*101 : Le MJ a remboursé le prêt.

J*102 : Le MJ a débouté la demande.

J*103 : Le joueur courant a éventuellement remboursé la demande.

T*110 (J*100 à J*101) : Si la case pour laquelle le prêt est réclamé est à l'état "autorisation de réclamer" pour ce joueur, le MJ rembourse le prêt et les intérêts.

T*120 (J*100 à J*102) : Si la case pour laquelle le prêt est réclamé est à l'état "interdiction de réclamer" pour ce joueur, le MJ déboute sa demande.

T*130 (J*101 à J*103) : Fin du remboursement.

T*140 (J*102 à J*103) : Fin du déboutement de remboursement.

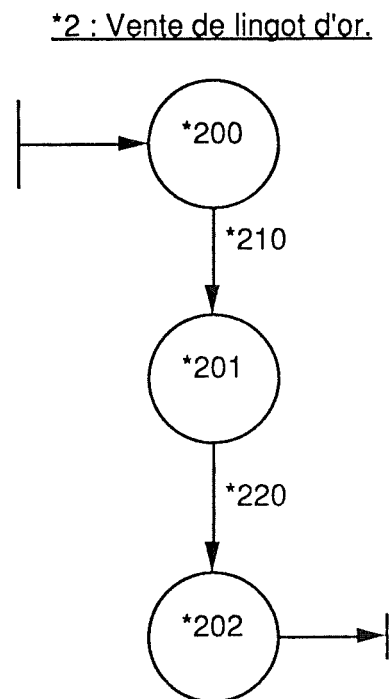
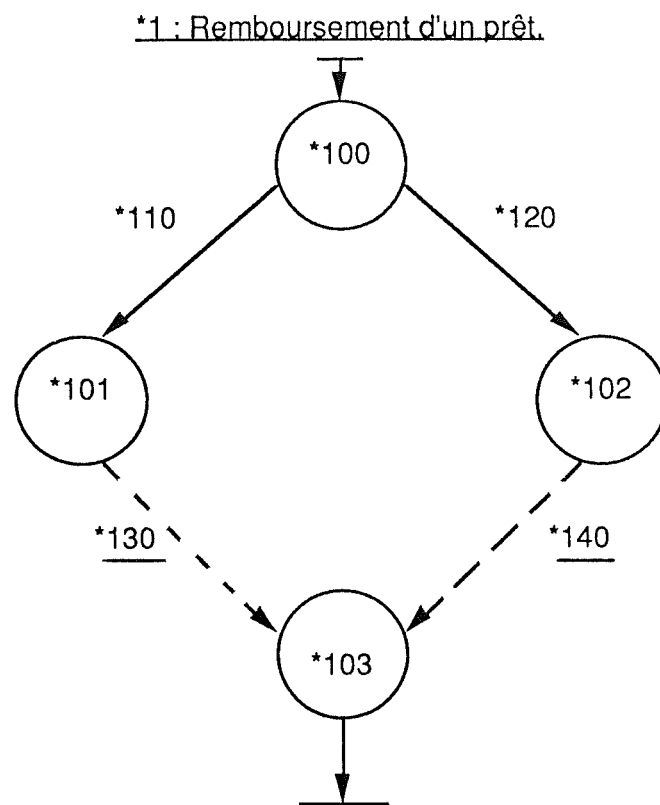


Figure IV.3 - Réseau de transition entre les jalons *1 et *2 du scénario.

*2 : (vente de lingot d'or).

J*200 : Un joueur a demandé la vente de son or.

J*201 : Le joueur a lancé les dés.

J*202 : Le MJ a payé le joueur selon les points obtenus au lancé de dés.

T*210 (J*200 à J*201) : Le joueur lance les dés.

T*220 (J*201 à J*202): Le MJ paie le joueur.

5° Spécification des acteurs : formalisation en terme d'attributs, d'une part, et cycle de vie et règles de couplage, d'autre part

a) Démarche de spécification originale

Le but de cette étape est de spécifier formellement les acteurs. Elle est divisée en deux sous-étapes distinctes. La première de ces deux sous-étapes, comme l'indiquent de Looz et Jourquin dans [DEL-JOU], consiste à faire une synthèse des données déjà relevées à propos des caractéristiques des acteurs. Nous pourrions ainsi dégager la structure interne des acteurs et regrouper ces derniers en classes. Le résultat attendu est une liste des attributs définitifs de chaque acteur. Pour chaque acteur, nous disposerons ainsi d'un ensemble de propositions définissant ses attributs. Les propositions sont bâties sous la forme suivante :

$\langle \text{nom-attr} \rangle [\langle \text{nom-acteur} \rangle] = \langle \text{domaine} \rangle .$

où $\langle \text{nom-attr} \rangle$ indique le nom de l'attribut considéré,

$\langle \text{nom-acteur} \rangle$ indique le nom de l'acteur décrit,

$\langle \text{domaine} \rangle$ indique le domaine de définition des valeurs de l'attribut

de Looz et Jourquin utilisent dans [DEL-JOU] les huit propositions suivantes :

vitesse [acteur] = entier.

Si l'entier vaut 0, indique que l'acteur n'est pas mobile ; sinon, indique la vitesse de l'acteur pour ses déplacements à l'écran.

nombre [acteur] = entier.

Indique le nombre d'occurrences possibles de l'acteur.

type [acteur] = {strings}.

Indique, sous forme de chaînes de caractères ("strings"), les différents types que l'acteur peut prendre. Un type d'un acteur est une caractéristique propre à un ensemble d'occurrences de cet acteur.

état [acteur] = {strings}.

Indique, sous forme de "strings", les différents états par lesquels peut passer l'acteur.

repr [acteur] = {forms}.

Indique les différentes représentations graphiques que peut prendre un acteur, soit sous la forme d'un dessin ("form"), soit sous la forme d'un "string" (dans ce cas les dimensions sont mentionnées).

loc [acteur] = point.

Indique la localisation de l'acteur dans le domaine d'animation. Cet attribut n'a de sens que si l'acteur est visible et fixe, sinon "point" vaut 0. La valeur de "point" s'exprime sous la forme d'une coordonnée (abscisse, ordonnée).

fct [acteur] = string.

Indique, en une phrase en langage naturel, quelle est la fonction de l'acteur.

compt [acteur] = { (pré, action, post) }.

(où "pré" et "post" sont des jalons et "action" un groupe verbal).

Indique le rôle, dans l'ensemble des comportements de l'acteur, sous forme de triplets "préconditions - action - postconditions" ou, autrement dit, sous forme "situation initiale - action - situation finale". Ces triplets sont représentés à trois niveaux d'indentation différents : le premier correspond aux comportements apparaissant lors de la découpe en parties, le

deuxième à celui de la découpe en séquences et le troisième à ceux de la découpe en tableaux. Nous représentons en caractères gras les comportements non décomposés afin de pouvoir visualiser immédiatement l'ensemble des comportements atomiques.

" La seconde sous-étape présente le cycle de vie des acteurs et les règles de couplage entre ces acteurs. (...).

Le cycle de vie des acteurs spécifie, sous la forme d'un graphe de transitions, toutes les transitions possibles d'états des acteurs. (...).

Les règles de couplages spécifient les liens entre les attributs des différents acteurs. Elles sont libellées sous la forme :

R : **SI** attr [acteur] = val
ALORS attr' [acteur'] = val'.

Les règles de couplages ne sont pas nécessairement toutes vérifiées à la fin de l'exécution d'une séquence (ou d'un tableau), mais leur éventuels non-respects sont des incohérences à éliminer dès qu'elles se présentent. Ainsi, en constatant que attr [act] = val, le meneur de jeu doit déclencher immédiatement une séquence (ou un tableau) qui établisse attr' [act'] = val' . " ([DEL-JOU], p. 54)

b) Problèmes rencontrés

Il est apparu très rapidement que les attributs proposés par de Looz et Jourquin ne pouvaient suffire à spécifier complètement les différents acteurs. Ceci est dû au fait que, dans leur système, les acteurs (c'est-à-dire principalement les clients de la banque) n'avaient pas à être différenciés l'un par rapport à l'autre, si ce n'est par un identifiant. En effet, aucun des acteurs n'avait une valeur (ou un ensemble de valeurs) d'attribut (ou d'attributs) qui lui était propre (par exemple, les différents clients de la banque n'étaient pas caractérisés par un solde de leur compte à vue, par un solde de leur livret d'épargne, par la somme d'argent qu'il comptait retirer (ou déposer) de (ou sur) l'un de ces comptes, etc.). Or, dans notre cas, les différents acteurs sont

caractérisés par un certain nombre de valeurs d'attributs qui leurs sont propres.

c) Solution proposée

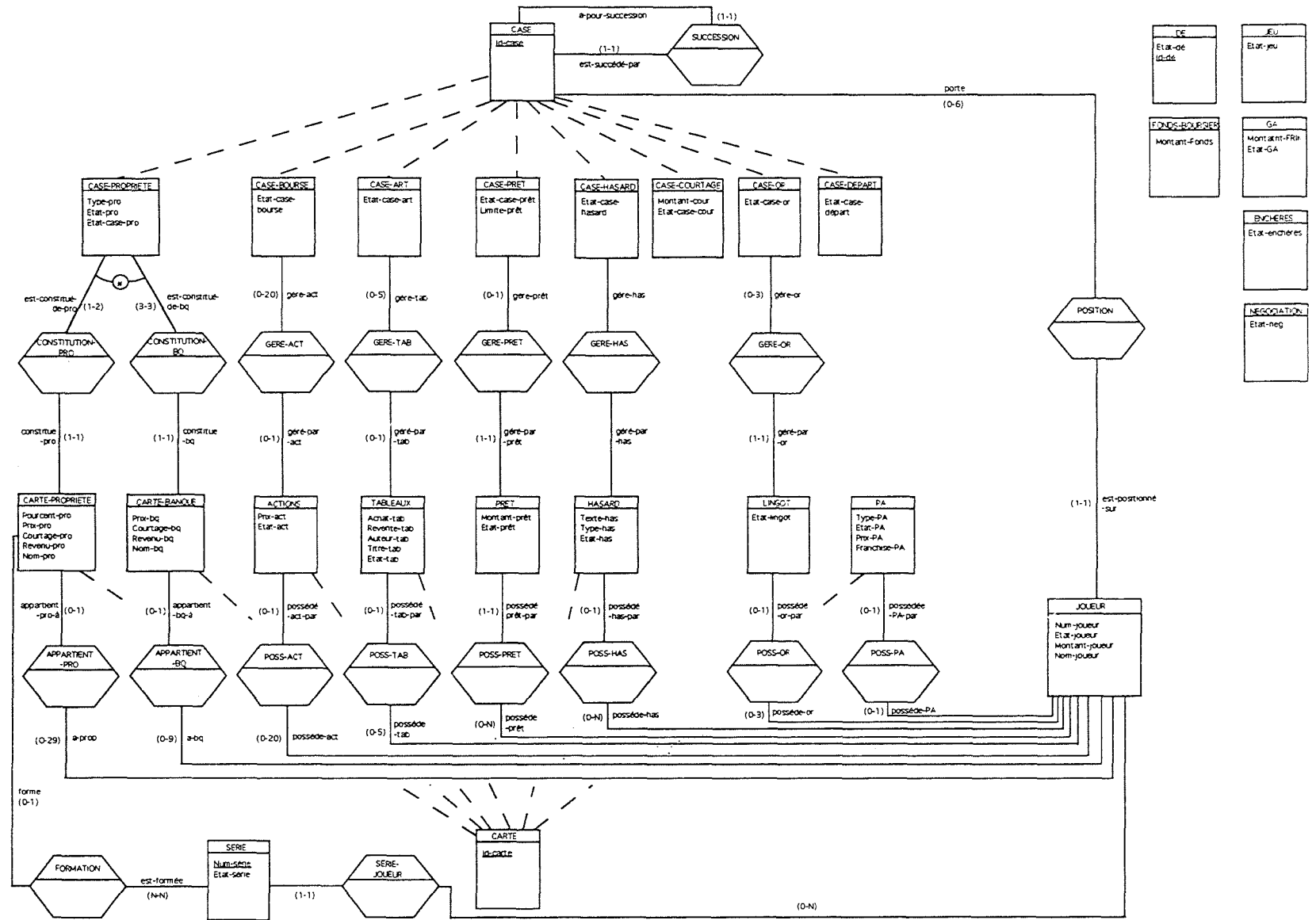
Au vu des problèmes évoqués ci-dessus, une analyse complète des données s'est avérée nécessaire. C'est en effet à ce stade que nous devons détenir la liste de tous les acteurs et de leurs caractéristiques car les étapes ultérieures considèrent que celle-ci est acquise. Cette analyse a été effectuée à l'aide de la méthode Entité-Relation développée dans [BODA-89].

Toutefois, une distinction entre deux principaux types d'attribut a été opérée à des fins pratiques. Ainsi, les attributs d'instances et les attributs de classe ou de système des acteurs ont été séparés.

Les attributs d'instances sont des attributs qui prennent des valeurs propres à chaque acteur, c'est-à-dire des valeurs pouvant différer d'un acteur à l'autre. Ainsi, par exemple, dans le jeu de rôle FRIK, les joueurs sont caractérisés par un numéro identifiant, par la somme d'argent qu'ils détiennent, etc. On retrouvera dans cette liste des attributs d'instances les attributs `type[acteur]` et `état[acteur]` définis par de Looz et Jourquin. Elle contiendra en outre tous les attributs figurant dans le schéma Entité-Relation.

Les attributs de classe ou de système sont, pour leur part, des attributs prenant des valeurs identiques pour tous les acteurs de la classe. Ils ne sont de ce fait représentés qu'une seule fois pour toute la classe. Ainsi, par exemple, tous les joueurs du jeu de rôle FRIK ont la même vitesse et possèdent les mêmes comportements. Le terme "attribut de système" correspond plus spécifiquement aux attributs nécessaires au bon fonctionnement du système. On retrouvera dans cette liste des attributs de classe ou de système les attributs `vitesse[acteur]`, `nombre[acteur]`, `repr[acteur]`, `fct[acteur]` et `compt[acteur]` définis par de Looz et Jourquin. On ajoutera également, pour tout acteur concerné, l'attribut `compt*[acteur]` correspondant aux comportements que cet acteur pourrait avoir dans les séries de tableaux définissant les "jalons étoiles" $*i$ ($i > 0$).

Figure IV.4 - Schéma Entité-Relation du jeu de rôle FRIK.



La raison principale de la distinction entre attributs d'instance et attributs de classe ou de système est que celle-ci s'accorde parfaitement avec le langage de classe dans lequel le système final sera implanté, à savoir Smalltalk/V.

d) Illustration

Nous illustrerons cette étape en formalisant les acteurs JOUEUR, CARTE et TABLEAU en terme d'attributs, en établissant leur cycle de vie et en exposant l'une des règles de couplage relevée dans le système. Le schéma Entité-Relation correspondant au jeu de rôle FRIK se trouve à la figure IV.4.

- Attributs d'instance des acteurs JOUEUR, CARTE et TABLEAU :

JOUEUR : *Num-joueur* : numéro du joueur { J1, J2, J3, J4, J5, J6 }. Il s'agit en fait d'un identifiant de JOUEUR.

Etat-joueur : état dans lequel se trouve le joueur {"en attente", "courant", "en faillite", "en négociation", "créé", "prêt à jouer", "offre de vente", "offre d'achat", "offre d'échange", "plus d'offre", "en manipulation", "plus de manipulation" }.

Montant-joueur : somme d'argent que détient le joueur.

Nom-joueur : nom du joueur.

Num-carte-pro-joueur : liste des numéros de cartes propriété que détient le joueur.

Num-carte-bq-joueur : liste des numéros de cartes de banque que détient le joueur.

Num-série-joueur : liste des numéros de séries que détient le joueur.

Poss-act-num : liste des actions que possède le joueur.

Poss-tab-num : liste des tableaux que possède le joueur.

Poss-prêt-num : liste des prêts que possède le joueur.

Poss-lingot-num : liste des lingots que possède le joueur.

Poss-PA-num : numéro de la PA que possède le joueur. S'il n'en possède pas, la valeur de *Poss-PA-num* est "0".

Poss-has-num : liste des cartes de hasard que possède le joueur.

Position-joueur : position courante du joueur sur le plateau de jeu. Elle vaut "0" si le jeu est fermé.

Réponse-joueur : réponse du joueur {"oui", "non"}

Réponse-joueur-offre : réponse du joueur à l'offre d'un autre joueur

CARTE : *Id-carte* : identifiant du générique CARTE.

TABLEAU : (spécialisation de CARTE)

Achat-tab : valeur d'achat du tableau.

Revente-tab : valeur de revente du tableau.

Auteur-tab : nom de l'auteur du tableau.

Titre-tab : titre du tableau.

Etat-tab : état de la carte { "vendue", "invendue", "retirée du jeu" }.

Num-joueur-tab : numéro du joueur à qui appartient le tableau. Si la carte n'est pas vendue ou est retirée du jeu, la valeur de *Num-joueur-tab* vaut "0".

Passons maintenant en revue les différents attributs de classe ou de système de ces trois acteurs.

- Attributs de classe ou de système des acteurs JOUEUR, CARTE et TABLEAU :

JOUEUR :

Vitesse[joueur] = *v*, où $v > 0$

Nombre[joueur] = 6.

Repr[joueur] = { cercle dans lequel est inscrit le numéro du joueur }.

Fct[joueur] = "partie prenante au jeu".

Compt[joueur] = {

(J1, participer-ouverture-jeu, J2)

(J1.1, **recevoir-argent-début**, J1.2)

(J2, participer-proposition-PA, J3)

(J2.1, **répondre-offre-PA**, J2.2)

(J2.2, **payer-recevoir-PA**, J2.3)

(J3, participer-lancement-dés, J4)

(J3, **devenir-joueur-courant**, J3.1)

(J3.1, **lancer-dés**, J4)

(J4, participer-avancement-case, J5)

(J4, **avancer-case-déclenchement-par-dessus**, J4.1)

(J4, **avancer-1-case**, J4.0.1)

(J4.0.1, **avancer-1-case-à-nouveau**, J4.0.1)

(J4.0.1, **déclencher-état-autorisation-réclamer**, J4.0.2)

- (J4.0.1, **participer-passage-départ**, J4.0.3)
- (J4.0.2, **avancer-1-case-à-nouveau**, J4.0.1)
- (J4.0.3, **avancer-1-case-à-nouveau**, J4.0.1)
- (J4.1, **faire-ce-que-peut-doit-faire-case**, J5)
 - (J4.1.1, **payer-somme-catastrophe**, J4.1.2)
 - (J4.1.1, **profiter-privilège**, J4.1.3)
 - (J4.1, **empocher-argent-courtage**, J4.1.4)
 - (J4.1, **réclamer-argent-prêt**, J4.1.5)
 - (J4.1.5, **prêter-argent**, J4.1.6)
 - (J4.1, **prêter-argent**, J4.1.6)
 - (J4.1, **vendre-tableaux**, J4.1.7)
 - (J4.1.7, **acheter-tableau**, J4.1.8)
 - (J4.1, **acheter-tableau-sans-vendre**, J4.1.8)
 - (J4.1, **vendre-actions**, J4.1.9)
 - (J4.1.9, **acheter-actions**, J4.1.10)
 - (J4.1, **acheter-actions-sans-vendre**, J4.1.10)
 - (J4.1, **acheter-lingot**, J4.1.11)
 - (J4.1, **payer-revenu-titre-semi-libre**, J4.1.12)
 - (J4.1, **payer-revenu-titre-occupée**, J4.1.13)
 - (J4.1.14, **acheter-titre-prop-libre-semi-libre**, J4.1.15)
 - (J4.1.16, **participer-vente-aux-enchères**, J4.1.17)
- (J5, **être-éliminé**, J6)
- (J6, **participer-négociation**, J7)
 - (J6, **faire-offre-vente-achat-échange**, J6.1)
 - (J6, **faire-offre-vente**, J6.0.1)
 - (J6, **faire-offre-achat**, J6.0.2)
 - (J6, **faire-offre-échange**, J6.0.3)
 - (J6.1, **refuser-accepter-offre**, J6.2)
 - (J6.2, **faire-nouvelle-offre**, J6)
 - (J6.2, **ne-plus-faire-offre**, J6.3)
- (J7, **manipuler-séries**, J8)
 - (J7, **retirer-titres-séries**, J7.1)
 - (J7.1, **désirer-manipuler-séries**, J7)
 - (J7.1, **placer-titres-séries**, J7.2)
 - (J7.1, **ne-plus-désirer-manipuler**, J7.3)
 - (J7.2, **ne-plus-désirer-manipuler**, J7.3)
 - (J7, **placer-titres-séries-sans-retirer**, J7.2)
 - (J7.2, **désirer-manipuler-séries**, J7)

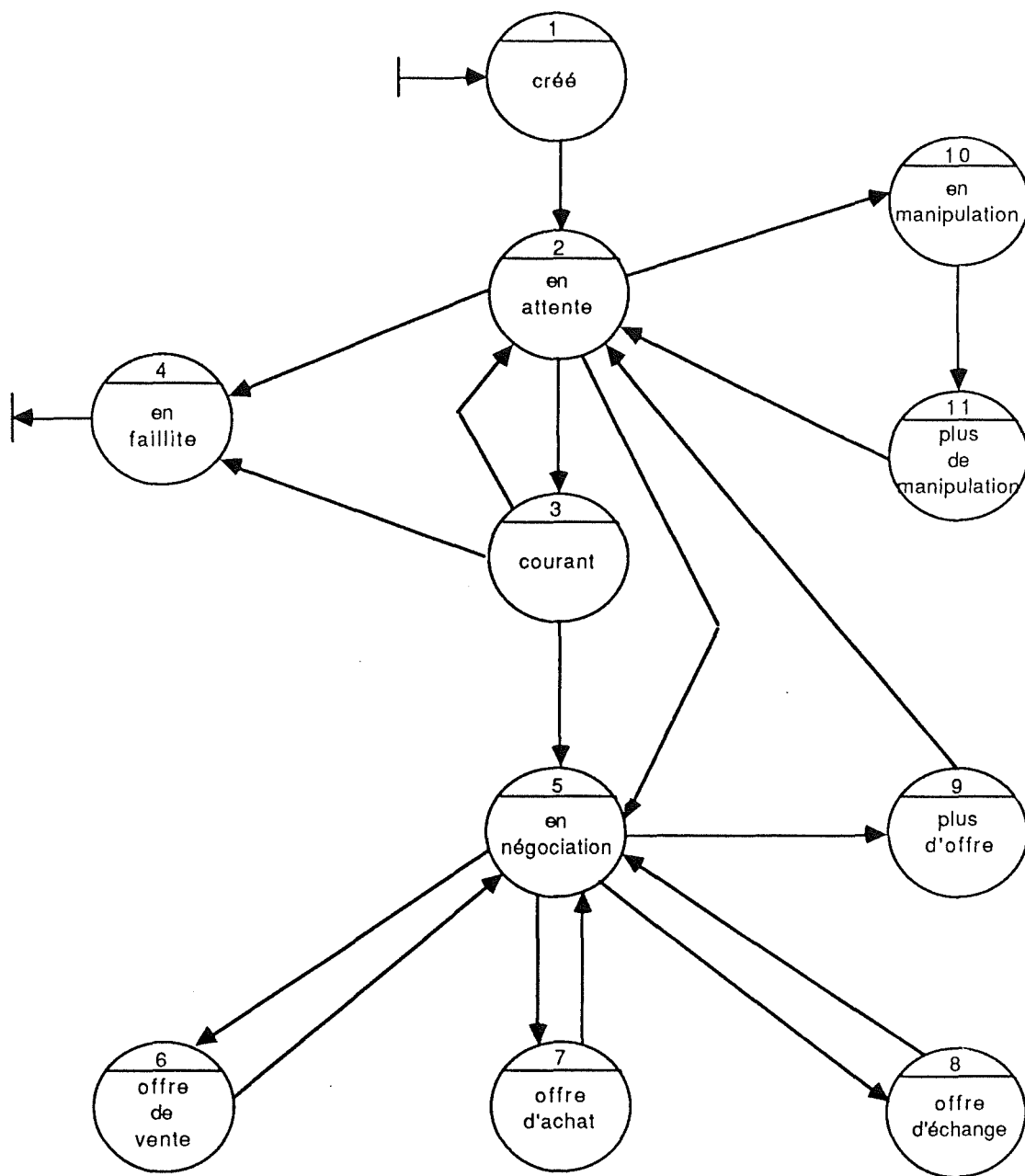


Figure IV.5 - Cycle de vie du JOUEUR.

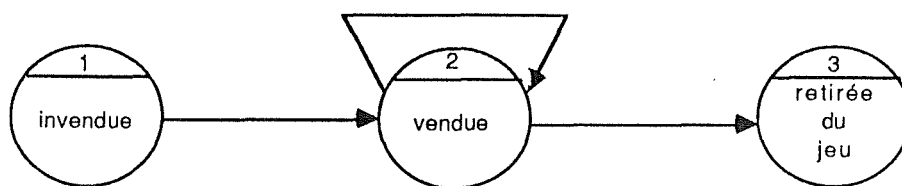


Figure IV.6 - Cycle de vie du TABLEAU.

(J8, **participer-tour-suivant**, J4) }.

Compt[joueur]* = { (*1, **demander-remboursement-prêt**, J*100)
(J*100, **recevoir-argent-intérêts**, J*101)

(*2, **demander-vente-or**, J*200)
(J*200, **lancer-dés**, J*201)
(J*201, **recevoir-argent-or**, J*202) }.

CARTE :

Vitesse[carte] = 0.
Nombre[carte] = 84.
Repr[carte] = "rectangle".
Fct[carte] = "représenter différents objets".
Compt[carte] = .

TABLEAU :

Vitesse[tableau] = 0.
Nombre[tableau] = 5.
Repr[tableau] = "rectangle avec TABLEAU inscrit dedans".
Fct[tableau] = "indiquer un placement d'argent dans un tableau".
Compt[tableau] = .

Nous pouvons maintenant élaborer les différents cycles de vie de ces trois acteurs.

- Cycle de vie des acteurs JOUEURS, CARTE et TABLEAU :

On trouvera aux figures IV.5 et IV.6 les cycles de vie des acteurs JOUEURS et TABLEAU. L'acteur CARTE n'ayant pas d'état, il ne possède donc pas de cycle de vie.

Enfin, nous allons exposer une des règles de couplage relevée dans le système.

- Exemple de règle de couplage relevée dans le système :

SI Etat-jeu = "fermé" ALORS

Etat-GA = "normal" ET Etat-propriété = "libre" ET Etat-carte-pro = "invendue"
ET Etat-carte-bq = "invendue" ET Etat-act = "invendue" ET Etat-tab =
 "invendue" ET Etat-PA = "invendue" ET Etat-has = "non distribuée" ET Etat-
 lingot = "invendu" ET Etat-négociation = "terminée" ET Etat-enchères =
 "terminée".

6° Spécification formelle des jalons

a) Démarche de spécification originale

Cette sixième étape consiste en la spécification formelle des jalons du scénario par la conjonction et/ou la disjonction de certaines propriétés des acteurs.

Nous disposerons de cette manière d'un outil formel pour l'activation des méthodes associées aux objets. Grâce au pré- et post-conditions associées aux comportements des acteurs, nous pourrons ainsi déduire à quel moment on peut envoyer un message à un objet et quel sera l'état du système après que la méthode associée ait été activée.

Pour des questions de lisibilité, nous préconisons l'utilisation d'abréviations pour désigner certaines valeurs clés des attributs des acteurs, abréviations qui seront énumérées et définies avant d'entamer la spécification formelle des jalons proprement dite.

Le résultat attendu lors de cette étape est, éventuellement, une liste d'abréviations et, dans tous les cas, une liste des jalons exprimés sous la forme de conjonction et/ou de disjonction de propositions.

b) Problème rencontré

Si cette sixième étape s'avère être indispensable en vue d'une implantation (cf ci-dessus), son application n'en est pas pour autant évidente. Il n'existe en effet pas de démarche systématique qui

permettrait d'éviter ou, en tout cas, de minimiser les erreurs ou oublis que le concepteur aurait pu commettre. C'est pourquoi nous proposerons une telle démarche systématique ci-dessous.

c) Solution proposée

Cette démarche, résultant d'une observation systématique des comportements du concepteur, se divise en deux parties : la première permet d'obtenir une spécification assez précise, en ce sens qu'elle comprend tous les acteurs jouant un rôle dans le jalon et tous les attributs pertinents de ces acteurs par rapport au jalon, tandis que la seconde conduit à une spécification formelle contrôlée et complétée le cas échéant ⁹.

Nous exposerons tout d'abord la démarche de spécification formelle des jalons sous la forme d'un algorithme et, dans un deuxième temps, nous le commenterons

⁹ Les chiffres entre parenthèses renvoient aux commentaires de l'algorithme.

Première partie :

Pour chaque jalon

Repérer les acteurs qui jouent un rôle dans le jalon (1)

Pour chaque acteur

Repérer les attributs pertinents par rapport au jalon (2)

Pour chaque attribut repéré

Repérer les valeurs (3) et/ou ensembles de valeurs pertinents que cet attribut peut prendre (4)

Fin-pour

Fin-pour

Classer les attributs repérés en ordre croissant du nombre de valeurs ou d'ensembles de valeurs qu'ils peuvent prendre (tout acteur confondu) (en cas de répétition dans le critère d'ordonnancement, classer les attributs concernés dans un ordre quelconque) (5)

Combiner les valeurs et/ou ensembles de valeurs de tous les attributs entre-eux (6)

Eliminer les lignes correspondant à des valeurs incompatibles d'attributs et/ou ne correspondant pas à l'état du jalon (7)

Fin-pour

A la fin de cette première partie, nous possédons une spécification assez précise, mais pas nécessairement correcte. C'est pourquoi nous proposons maintenant une seconde partie permettant de contrôler et de compléter cette spécification.

Seconde partie :

Etape 1 :

Pour chaque jalon spécifié conformément à la première partie

Cocher dans les cycles de vie des acteurs les états
présentés dans le jalon

Examiner les attributs et leurs valeurs (ou une partie
d'entre-eux) et voir si ça ne correspondrait pas plutôt à un
nouvel état d'un des acteurs (8)

Si la réponse est oui

Ajouter dans le cycle de vie de l'acteur ce nouvel état

Remplacer dans la spécification formelle du jalon les
attributs et valeurs impliqués par le nouvel état

Voir s'il n'y a pas d'implications dues à cette
introduction et en tenir compte le cas échéant (9)

Fin-si

Fin-pour

Etape 2 :

Examiner les cycles de vie des acteurs et voir si tous les
états ont été marqués

Si la réponse est oui

Rien à faire

Sinon (c'est-à-dire si la réponse est non)

Voir s'il s'agit d'un oubli dans les jalons ou s'il s'agit d'un
état non utilisé

Si l'état est non utilisé

Le retirer de l'ensemble des états de l'acteur (10)

Sinon (c'est-à-dire s'il s'agit d'un oubli)

Ajouter cet état dans le ou les jalons incriminés (11)

Recommencer l'étape 1 pour ces jalons

Fin-si

Fin-si

Commentaires de l'algorithme :

La première partie de la démarche commence par un repérage de
tous les acteurs jouant un rôle dans le jalon à spécifier formellement
(1). A cette fin, nous pouvons nous aider des attributs "Compt" contenus

dans la liste des attributs de classe ou de système des acteurs. En effet, tout acteur dont un des comportements a pour postcondition le jalon étudié joue un rôle dans le dit jalon. Il est cependant important de signaler que l'on ne retrouvera pas forcément tous les acteurs impliqués dans le jalon de cette manière : un simple changement d'état d'un acteur n'est en général pas référencé dans cette liste. Nous conseillons donc d'utiliser cette liste comme un premier contrôle de complétude.

Il nous faut ensuite repérer les attributs pertinents par rapport au jalon à formaliser (2), c'est-à-dire les attributs qui peuvent ou doivent changer de valeur pour correspondre au jalon. Afin de ne pas en oublier, on passera successivement en revue tous les attributs de chaque acteur repéré précédemment

Il nous faut encore repérer les valeurs et/ou ensembles de valeurs pertinents que cet attribut peut prendre par rapport au jalon (4), c'est-à-dire les valeurs que peut prendre l'attribut pour correspondre à l'état du jalon. Notons que ces valeurs peuvent être représentées par des formules de calcul.

Maintenant que nous possédons une liste des attributs pertinents, ainsi que les valeurs et/ou ensembles de valeurs pertinents par rapport au jalon, il reste à spécifier ce dernier de manière formelle. Afin de n'oublier aucune combinaison valable des valeurs et/ou ensembles de valeurs d'attributs, nous combinerons systématiquement toutes les valeurs et /ou ensembles de valeurs entre-eux (6) dans une matrice que nous appellerons **matrice de combinaison**¹⁰. Cette matrice possède autant de colonnes que d'attributs pertinents repérés et compte autant de lignes que le produit des nombres de valeurs et/ou ensembles de valeurs pertinents que ces attributs peuvent prendre. Vu la taille que la matrice de combinaison peut prendre, nous désirons réduire le nombre d'écritures à y effectuer. C'est pourquoi nous procéderons d'abord à un classement des attributs repérés en ordre croissant du nombre de valeurs ou d'ensembles de valeurs qu'ils peuvent prendre (5). Cette

¹⁰ Chaque valeur ou ensemble de valeurs d'attribut doit être combiné avec toutes les autres valeurs et ensembles de valeurs des autres attributs.

recommandation n'est pas obligatoire, mais est vivement conseillée. En effet, supposons que nous ayons les attributs pertinents suivant :

attr1, attr2, attr3.

Supposons en outre que attr1 puisse prendre 2 valeurs pertinentes distinctes (a et b), attr2 puisse en prendre 3 (c,d et e), et attr3 puisse en prendre 4 (f,g,h,i).

Supposons enfin que nous choissions l'ordre attr1-attr2-attr3 pour classer ces différents attributs.

La matrice de combinaison résultante (comptant trois colonnes et 24 lignes) est pour ce classement :

	attr1	attr2	attr3
1.	a	c	f
2.			g
3.			h
4.			i
5.		d	f
6.			g
7.			h
8.			i
9.		e	f
10.			g
11.			h
12.			i
13.	b	c	f
14.			g
15.			h
16.			i
17.		d	f
18.			g
19.			h
20.			i
21.		e	f
22.			g
23.			h
24.			i

Ce classement induit $2 + (2 \times 3) + (2 \times 3 \times 4)$ écritures dans la matrice, soit 32 écritures.

Supposons maintenant que l'ordre choisi est l'ordre attr3-attr2-attr1. La matrice de combinaison est alors :

	attr1	attr2	attr3
1.	f	c	a
2.			b
3.		d	a
4.			b
5.		e	a
6.			b
...			
19.	i	c	a
20.			b
21.		d	a
22.			b
23.		e	a
24.			b

Ce classement induit donc $4 + (4*3) + (4*3*2)$ écritures dans la matrice, soit 40 écritures.

Le classement permettant un minimum d'écritures est donc le classement des attributs en ordre croissant du nombre de valeurs qu'ils peuvent prendre (les autres classements auraient donné les nombres d'écritures suivants :

attr1-attr3-attr2 : $2 + (2*4) + (2*4*3) = 34$ écritures;

attr2-attr1-attr3 : $3 + (3*2) + (3*2*4) = 33$ écritures;

attr2-attr3-attr1 : $3 + (3*4) + (3*4*2) = 39$ écritures;

attr3-attr1-attr2 : $4 + (4*2) + (4*2*3) = 36$ écritures.).

En cas de répétition dans le critère d'ordonnancement, les attributs concernés peuvent être classés dans un ordre quelconque.

Après que ce classement ait éventuellement été effectué, nous pouvons combiner les valeurs et/ou ensembles de valeurs de tous les attributs entre-eux (6) afin d'obtenir la matrice de combinaison. Il ne nous reste plus alors qu'à éliminer les lignes de la matrice correspondant à des valeurs incompatibles d'attributs (dans ce sens qu'elles se contredisent totalement ou partiellement) et/ou ne correspondant pas à l'état du jalon (7).

Nous sommes maintenant arrivé à la fin de la première partie. Nous disposons donc à cet instant d'une spécification des jalons assez

précise. Il nous faut encore la contrôler et la compléter : telle est la finalité de cette deuxième partie de la démarche de spécification formelle des jalons.

Cette deuxième partie est divisée à son tour en deux étapes. L'étape 1 vise à repérer de nouveaux états des acteurs, tandis que l'étape 2 vise pour sa part à compléter la définition formelle des jalons et à supprimer les états des acteurs non nécessaires au système.

Examinons pour commencer l'étape 1. Nous devons tout d'abord cocher dans les cycles de vie des acteurs les états présentés dans chaque jalon (il s'agit en fait d'une préparation à l'étape 2). Il s'agit maintenant de repérer les éventuels nouveaux états des acteurs. A cette fin, nous allons examiner, pour chaque jalon, les attributs et leurs valeurs (ou une partie d'entre-eux) et voir si la combinaison de ceux-ci ne correspondrait pas plutôt à un nouvel état d'un des acteurs (8). En effet, une partie (ou l'entièreté) des valeurs (ou ensembles de valeurs) des attributs peut en fait représenter un nouvel état (que l'on n'avait pas relevé) d'un acteur, beaucoup plus simple celui-là que la combinaison dont il est issu. Dans le cas d'une réponse positive, il faut bien entendu ajouter dans le cycle de vie de l'acteur concerné ce nouvel état et remplacer dans la spécification formelle du jalon examiné les attributs et valeurs impliqués par ce nouvel état. Il reste cependant à voir s'il n'y a pas d'implications dues à cette introduction dans d'autres jalons et, dans ce cas, en tenir compte (9). Il est en effet possible que la combinaison des valeurs (ou ensembles de valeurs) des attributs que l'on a remplacée par un nouvel état d'un acteur puisse également apparaître dans la spécification formelle d'autres jalons. Le remplacement doit donc aussi être effectué dans ceux-ci. En outre, il se pourrait que ces remplacements fassent apparaître à nouveau un nouvel état d'un acteur. Dans ce cas, il faudra reprendre l'étape 1 de la seconde partie depuis le début.

Il nous faut encore compléter la définition formelle des jalons et supprimer les états superflus. C'est ce que nous ferons à l'étape 2.

Celle-ci débute par un examen des cycles de vie des différents acteurs et un repérage des états qui n'auraient pas été marqués lors de l'étape 1 de la seconde partie. Si tous les états ont été marqués, nous

pouvons considérer que la spécification formelle des jalons est correcte. Dans le cas contraire, il reste à voir pourquoi chacun de ces états n'a pas été marqué. S'il s'agit d'un état non utilisé, il est souhaitable de le retirer des états de l'acteur (10) ...et donc de son cycle de vie! Par contre, s'il s'agit d'un oubli dans la spécification formelle d'un ou de plusieurs des jalons, il faut ajouter cet état dans le ou les jalons incriminés (11). En effet, s'il s'agit d'un oubli, c'est qu'à un moment ou un autre la spécification formelle d'un (ou plusieurs) des jalons doit contenir cet état. Autrement dit, le scénario passe obligatoirement à un moment ou un autre dans un état particulier comprenant cet état de l'acteur. Il faut donc que cet état soit ajouté dans la spécification formelle de ces jalons. Cette dernière a cependant été modifiée; il faut donc recommencer l'étape 1 de la seconde partie pour ces jalons.

A l'issue de cette étape 2 de la seconde partie, nous pouvons considérer que la spécification formelle des jalons est correcte ou, en tout cas, contient un minimum d'erreurs ou d'oublis.

Pour des raisons de lisibilité, généralement seuls les attributs subissant une modification de valeur par rapport à celle qu'ils avaient dans leur(s) jalon(s) prédécesseur(s) sont mentionnés dans la spécification formelle des jalons.

d) Illustration

- Première partie

Pour effectuer l'illustration de cette première partie, on se basera sur le jalon 4.1 :

J4.1 : Le joueur a avancé du nombre de cases désignés par les dés en déclenchant des comportements éventuels de case en passant par-dessus.

Les acteurs qui jouent un rôle dans le jalon sont le joueur, les dés, le jeu et la case. Le joueur n'ayant reçu aucune modification pour

arriver à cet état, il peut être ignoré. Les dés peuvent être ignorés pour la même raison que le joueur. Il reste donc le jeu et la case.

Les attributs pertinents de jeu par rapport au jalon sont Nbre-avancé et Somme-dé (car c'est par comparaison de leur valeur réciproque que l'on se rend compte si oui ou non le joueur a avancé du nombre de cases voulu); celui de case est Etat-case (car c'est grâce à cet attribut que l'on se rend compte que l'on a bel et bien déclenché le comportement de la case sur laquelle on aboutit finalement).

L'attribut Nbre-avancé peut prendre une valeur et un ensemble de valeurs distincts, à savoir la valeur de la somme des dés ou un nombre x strictement inférieur à la valeur des dés. Quant à l'attribut Somme-dé, la seule valeur qu'il puisse prendre est la somme réelle des dés. Cette valeur n'ayant pas changée, cet attribut peut être ignoré.

Les valeurs pertinentes que peut prendre l'attribut Etat-case sont "normale" et "déclenchée".

On constate alors que chaque attribut possède le même nombre de valeurs distinctes. Le classement des attributs résulte donc d'un choix personnel. Nous choisirons plutôt l'ordre Nbre-avancé ** Etat-case car l'attribut considéré le plus important s'avère être Nbre-avancé. En effet, quelle que soit la valeur de Etat-case, celle-ci ne sera intéressante que dans la mesure où l'on aura d'abord examiné la valeur de Nbre-avancé pour savoir si oui ou non on a avancé du nombre de cases voulu.

Nous allons maintenant combiner ces différentes valeurs entre-elles :

	Nbre-avancé	Etat-case
1.	$x < \text{Somme-dé}$	normale
2.		déclenchée
3.	$x = \text{Somme-dé}$	normale
4.		déclenchée

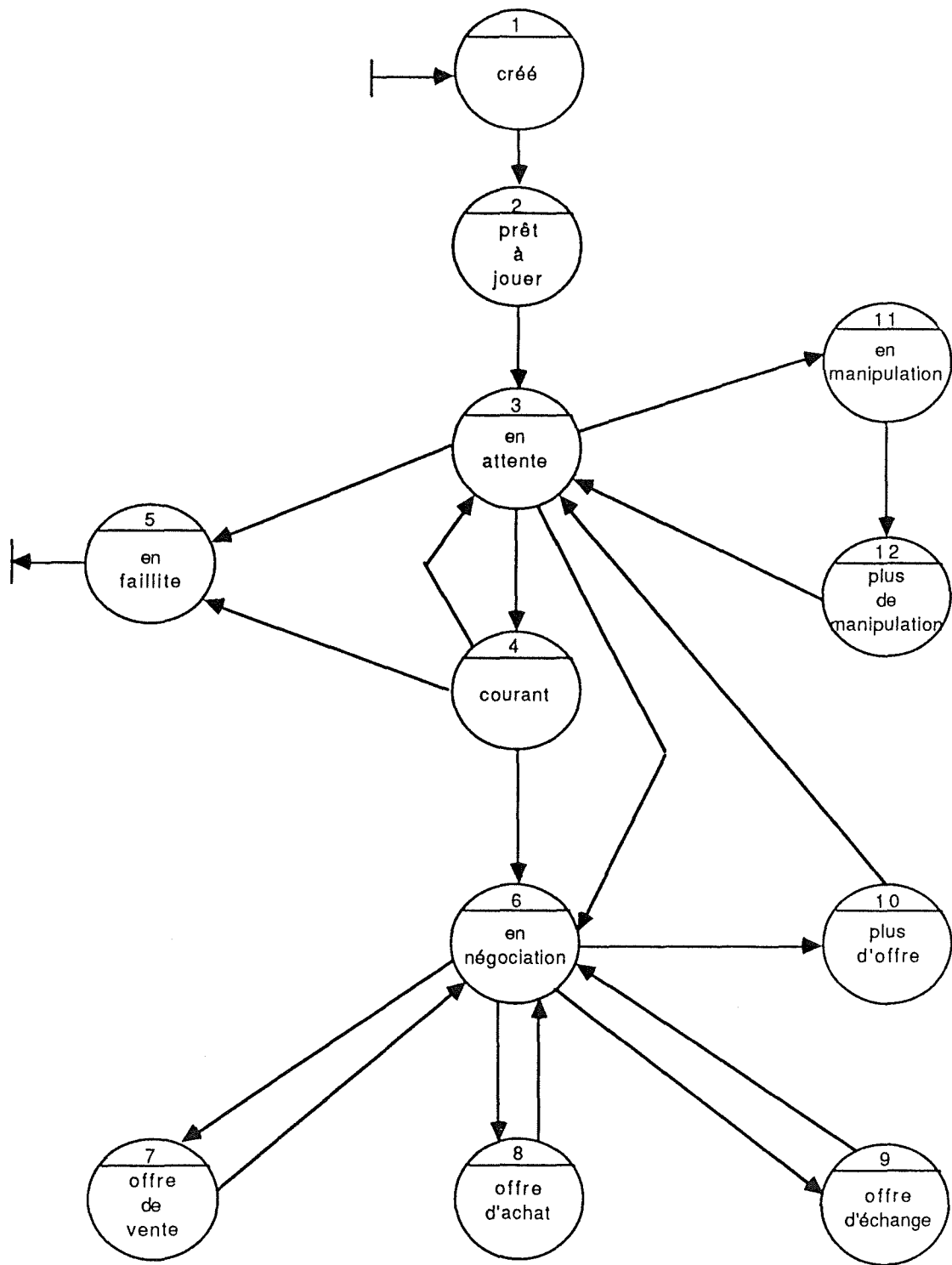


Figure IV.7 - Nouveau cycle de vie du JOUEUR.

Les lignes 1 et 2 ne correspondent pas au jalon puisque la valeur de Nbre-avancé est strictement inférieure à la valeur de Somme-dé. Il reste donc les lignes 3 et 4.

Cependant, la ligne 3 ne correspond pas non plus au jalon puisque le joueur est maintenant arrêté sur la case voulue, ce qui a pour conséquence de changer l'état d'Etat-case à "déclenchée". Il ne reste finalement que la ligne 4.

La spécification formelle du jalon est donc :

$$\text{Nbre-avancé}[\text{Jeu}] = \text{Somme-dé}[\text{Jeu}] \text{ ET } \text{Etat-case} = \text{"déclenchée"}$$

Nous illustrerons la deuxième partie de la démarche en l'appliquant aux jalons pour lesquels une modification a été opérée et en relevant les états qui n'avaient pas été marqués.

- Deuxième partie :

Le jalon J1.2 (J1.2 : Montant-joueur = agd) correspond plutôt à un nouvel état de joueur, à savoir que le joueur est "prêt à jouer". Cela nécessite à nouveau la création d'un autre état : celui de "créé" pour pouvoir définir de manière formelle l'état des joueurs lorsqu'ils viennent d'être créés (cf J1.1 : Nbre-joueur-jeu = x, où $2 \leq x \leq 6$). Le nouveau cycle de vie du joueur est présenté à la figure IV.7. Les spécifications des deux jalons deviennent :

$$\text{J1.2 : } \text{Etat-joueur}[\text{Ji}] = \text{"prêt à jouer"} \text{ où } 2 \leq i \leq \text{nbrej}$$

$$\text{J1.1 : } \text{Nbre-joueur-jeu} = x, \text{ où } 2 \leq x \leq 6$$

$$\text{ET } \text{Etat-joueur}[\text{Ji}] = \text{"créé"} \text{ avec } 2 \leq i \leq \text{nbrej}$$

L'état "vendue" de Etat-PA n'a pas été coché. Il s'agit en fait d'un oubli dans le jalon J2.3. L'état est donc ajouté dans le jalon et sa spécification formelle devient :

J2.3 : [Réponse-joueur[Ji] = "oui"

ET Montant-joueur[Ji] = mj - Prix-PA[PA tirée]

ET Poss-PA-num = Id-carte[PA tirée]

ET Montant-FRIK[GA] = Montant-FRIK[GA] + Prix-PA[PA tirée]

ET Etat-PA[PA-tirée] = "vendue"]

OU

[Réponse-joueur[Ji] = "non"]

Les états "invendue" de Etat-tab, "invendue" de Etat-PA, "non distribuée" de Etat-has, "invendu" de Etat-lingot n'ont pas été cochés. Cela fait apparaître le problème de l'initialisation. La spécification formelle du jalon J1 devient donc :

J1 : Etat-jeu = "fermé"

ET Etat-GA = "normal"

ET Etat-propriété = "libre"

ET Etat-carte-pro = "invendue"

ET Etat-carte-bq = "invendue"

ET Etat-act = "invendue"

ET Etat-tab = "invendue"

ET Etat-PA = "invendue"

ET Etat-has = "non distribuée"

ET Etat-lingot = "invendu"

ET Etat-négociation = "terminée"

ET Etat-enchères = "terminée"

7° Pour chaque acteur, description du plan de système et du diagramme de transition

a) Démarche de spécification originale

Il s'agit d'une étape de modélisation. Elle consiste à modéliser séparément chaque acteur doté d'un comportement car ces acteurs échangent des messages avec d'autres acteurs et peuvent éventuellement déclencher un comportement d'un autre acteur. Les notions de plan de système et de diagramme de transition sont empruntés à la méthode E.P.A.S..

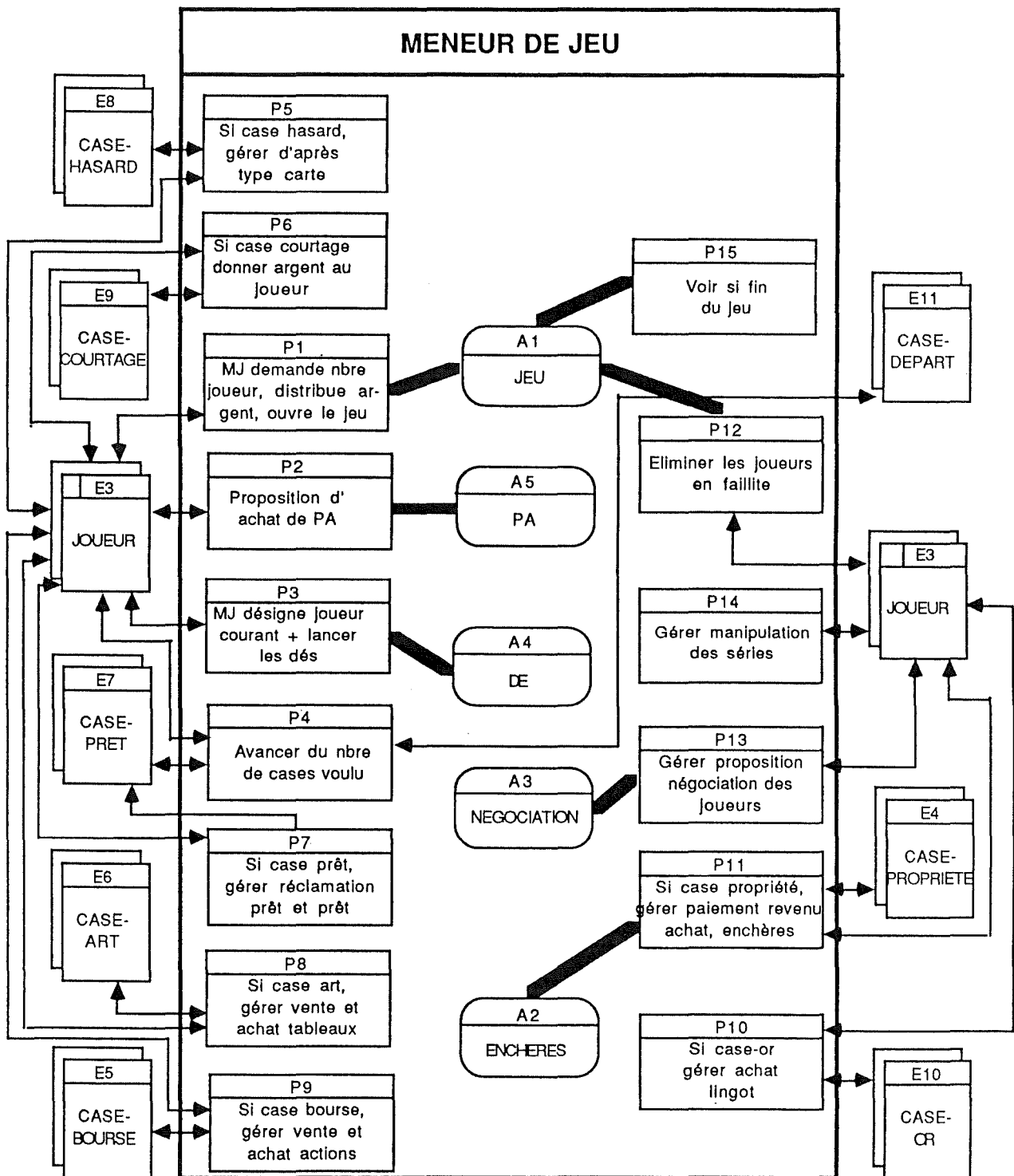


Figure IV.8 - Plan de système du MENEUR DE JEU.

" Chaque jalon du scénario correspond à une situation du domaine d'animation, qui est fonction de tous ses acteurs. Chacun des tableaux du scénario implique la mise en oeuvre par les acteurs d'une stratégie dictée par le meneur de jeu visant à faire passer le système d'une situation initiale à une situation finale, toutes deux exprimées par des jalons. Un processus peut être associé à un tableau ou à un ensemble de tableaux. Chaque processus représente un ensemble de comportements du système, mais ne décrit ni quand ni comment ces comportements seront menés à bien, puisque la modélisation en plans de système est purement conceptuelle et atemporelle. Le plan de système décrit donc un cadre d'intégration d'un ensemble d'activités. " [DEL-JOU], p. 62

" Le but de cette étape est également de dégager un modèle des comportements assez fin pour pouvoir présenter des diagrammes de transition, qui font intervenir la dimension temporelle. Les diagrammes de transition permettent de représenter, pour un scénario donné, les activités qui doivent être mises en oeuvre, les états qui sont atteints lorsque l'activité est terminée et les informations en provenance ou à destination des acteurs externes. " [DEL-JOU], p. 62

Deux perspectives intéressantes s'ajoutent lors de cette étape. Il s'agit des flux de communication entre les acteurs (plan de système) et la dimension dynamique de l'enchaînement des processus (diagrammes de transition). " Les plans de systèmes et les diagrammes de transition permettent de regrouper en processus les comportements solidaires, de connaître les transitions possibles entre ces processus, ainsi que les caractéristiques des acteurs que chaque processus doit mettre à jour. Ces informations n'apparaîtraient pas si on omettait cette dernière étape. " [DEL-JOU], p. 63.

Les concepts d'accumulations, d'acteurs externes et de processus sont, tous, définis à l'annexe 1.

Cette étape ne posant aucun problème, on passera directement à l'illustration.

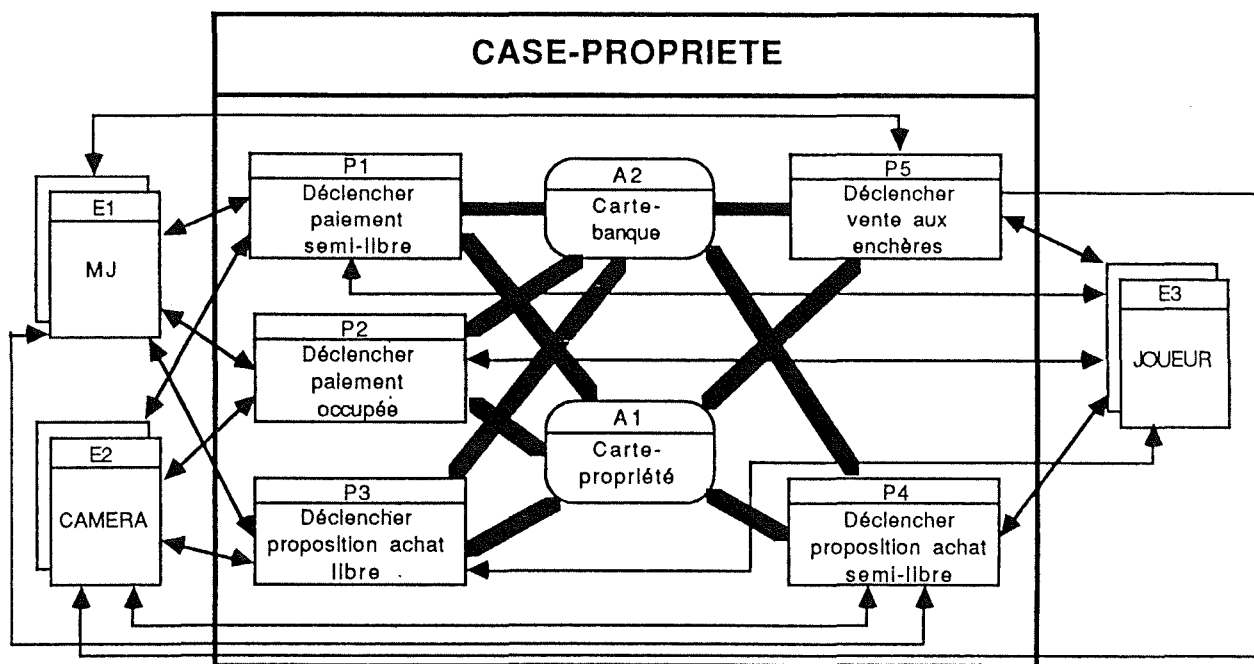


Figure IV.10 - Plan de système de la CASE-PROPRIETE.

b) Illustration

Pour illustrer cette ultime étape de la démarche de spécification originale, nous exposerons les plans de système et les diagrammes de transition des acteurs MENEUR DE JEU et CASE-PROPRIETE.

• Le MENEUR DE JEU :

Les accumulations relevées pour le système du meneur de jeu sont :

- A1 : Jeu
- A2 : Enchères
- A3 : Négociation
- A4 : Dé
- A5 : PA

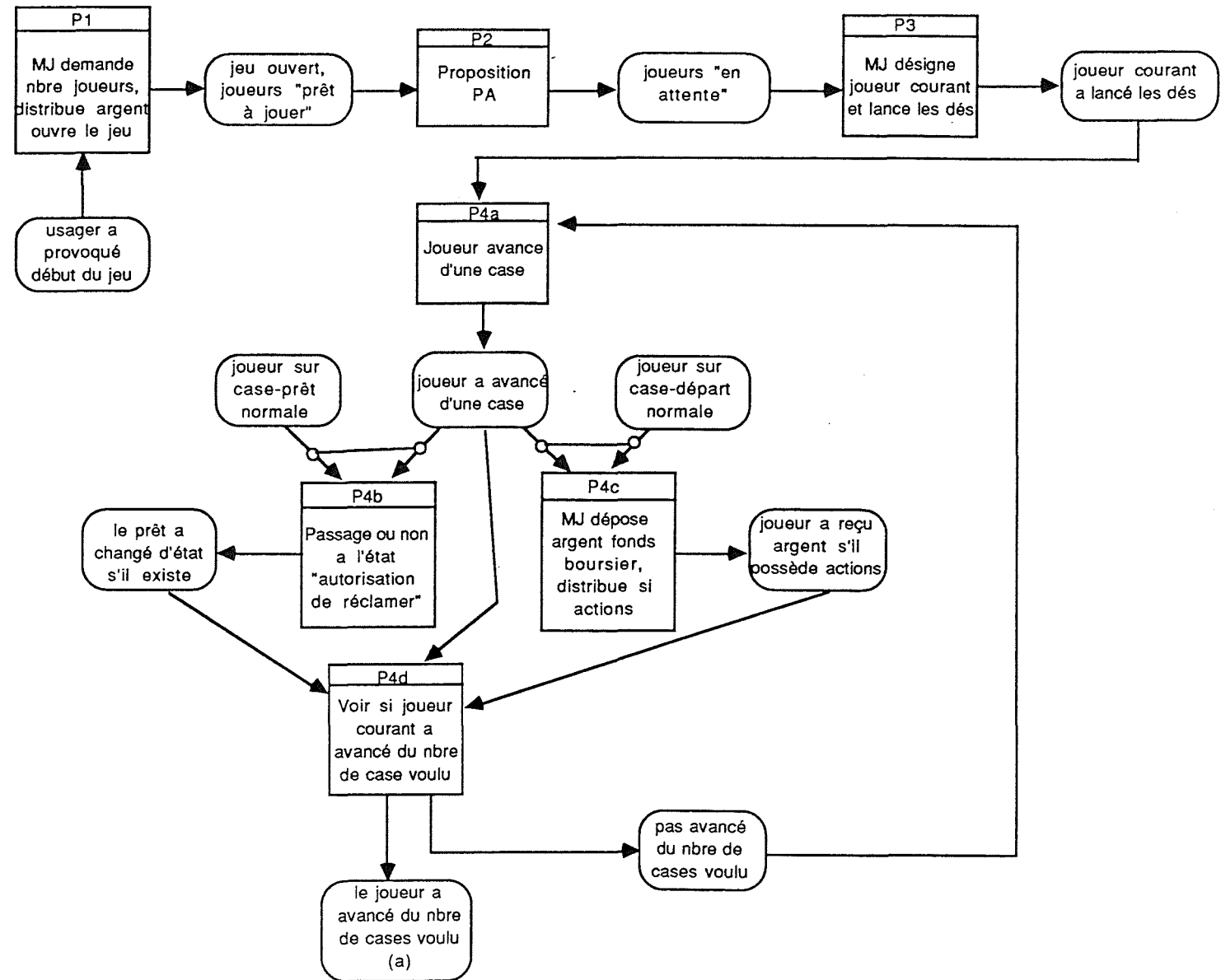
Les acteurs externes qui interagissent avec le système sont :

- E3 : Joueur
- E4 : Case-Propriété
- E5 : Case-Bourse
- E6 : Case-Art
- E7 : Case-Prêt
- E8 : Case-Hasard
- E9 : Case-Courtage
- E10 : Case-Or
- E11 : Case-Départ

Les 15 processus apparaissant dans les diagrammes sont :

- P1 : Le MJ demande le nombre de joueurs, distribue l'argent et ouvre le jeu (T10, T20, T30)
- P2 : Le MJ propose aux joueurs d'acheter une PA (T40, T60, T70, T80)
- P3 : Le MJ désigne le joueur courant et celui-ci lance les dés (T90, T100)
- P4 : Le joueur avance du nombre de cases voulu (T110, T115, T120, T130, T140, T150, T155, T160, T170)
- P5 : Si case hasard, gérer d'après le type de la carte (T180, T190, T200, T400, T410)

Figure IV.9 (a) - Diagramme de transition du MENEUR DE JEU.



- P6 : Si case courtage, donner au joueur la somme contenue dans la case (T210, T420)
- P7 : Si case prêt, gérer réclamation argent et prêt du joueur (T220, T230, T240, T440)
- P8 : Si case art, gérer vente et achat de tableaux (T250, T260, T270, T460)
- P9 : Si case bourse, gérer vente et achat actions (T280, T290, T300, T480)
- P10 : Si case or, gérer achat d'un lingot d'or (T310, T490)
- P11 : Si case propriété, gérer paiement revenu, achat de titre par joueur, vente de titre aux enchères (T320, T330, T335, T340, T350, T360, T365, T370, T375, T380, T390, T395)
- P12 : Elimination des joueurs en faillite (T500)
- P13 : Gérer proposition de négociation des joueurs (T502, T504, T506, T510, T512, T515, T525, T530, T535, T540)
- P14 : Gérer manipulation des séries (T550, T555, T560, T565, T570, T580, T590, T593, T595)
- P15 : Voir si fin du jeu par le nombre de joueurs restant (T600, T605, T610)

On trouvera respectivement le plan de système et le diagramme de transition du MENEUR DE JEU aux figures IV.8 et IV.9.

• La CASE-PROPRIETE

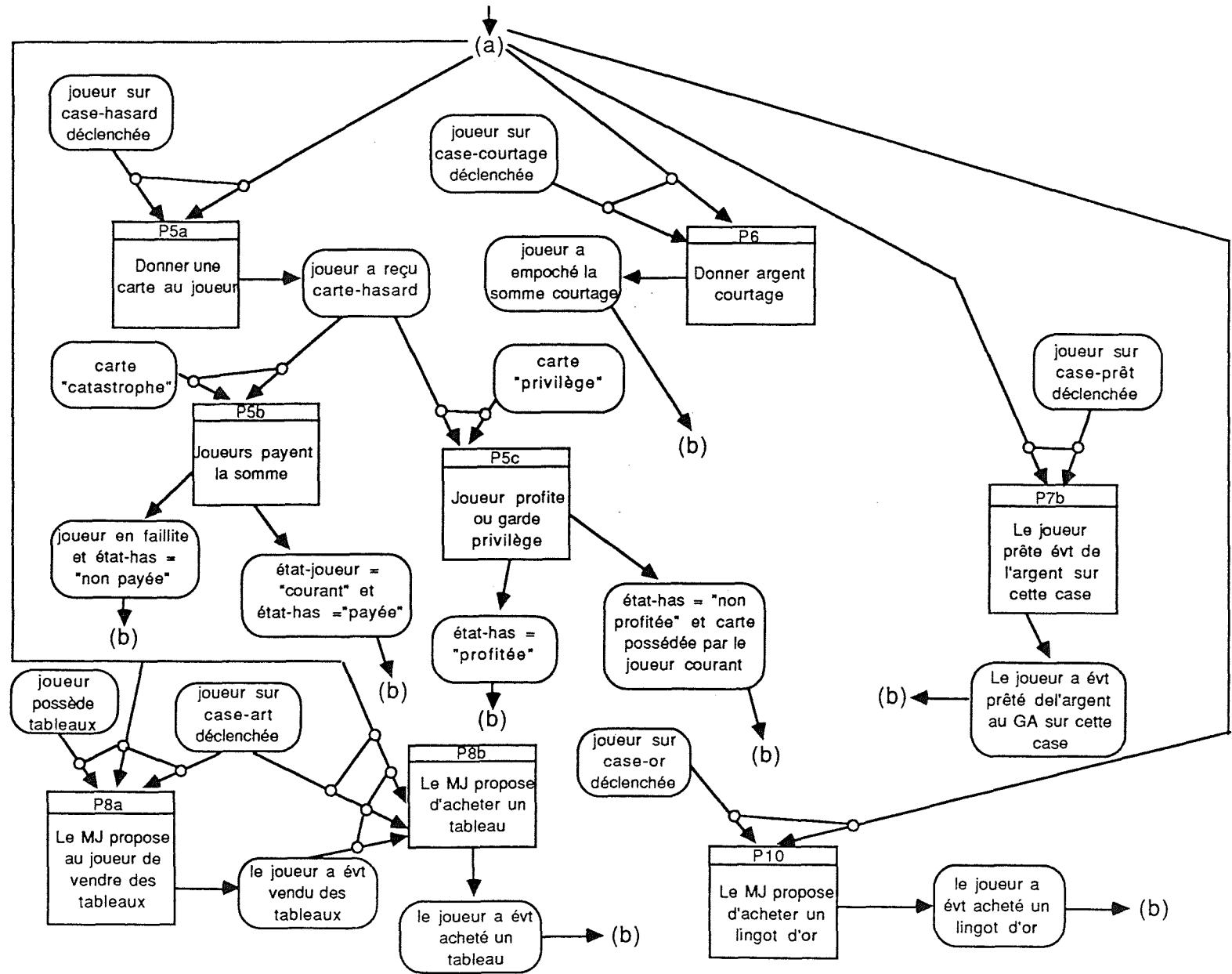
Les accumulations relevées pour la case propriété sont :

- A1 : Carte-Propriété
- A2 : Carte-Banque

Les acteurs externes qui interagissent avec le système sont :

- E1 : MJ
- E2 : Caméra
- E3 : Joueur

Figure IV.9 (b) - Diagramme de transition du MENEUR DE JEU.



Les 5 processus apparaissant dans les diagrammes sont :

- P1 : Déclencher le paiement si la propriété est semi-libre (T320)
- P2 : Déclencher le paiement si la propriété est occupée (T330)
- P3 : Déclencher proposition d'achat du titre si la propriété est libre (T340, T360, T365)
- P4 : Déclencher proposition d'achat du titre si la propriété est semi-libre (T350, T360, T365)
- P5 : Déclencher la vente aux enchères si le titre n'est pas vendu (T370)

On trouvera respectivement le plan de système et le diagramme de transition de la CASE-PROPRIETE aux figures IV.10 et IV.11.

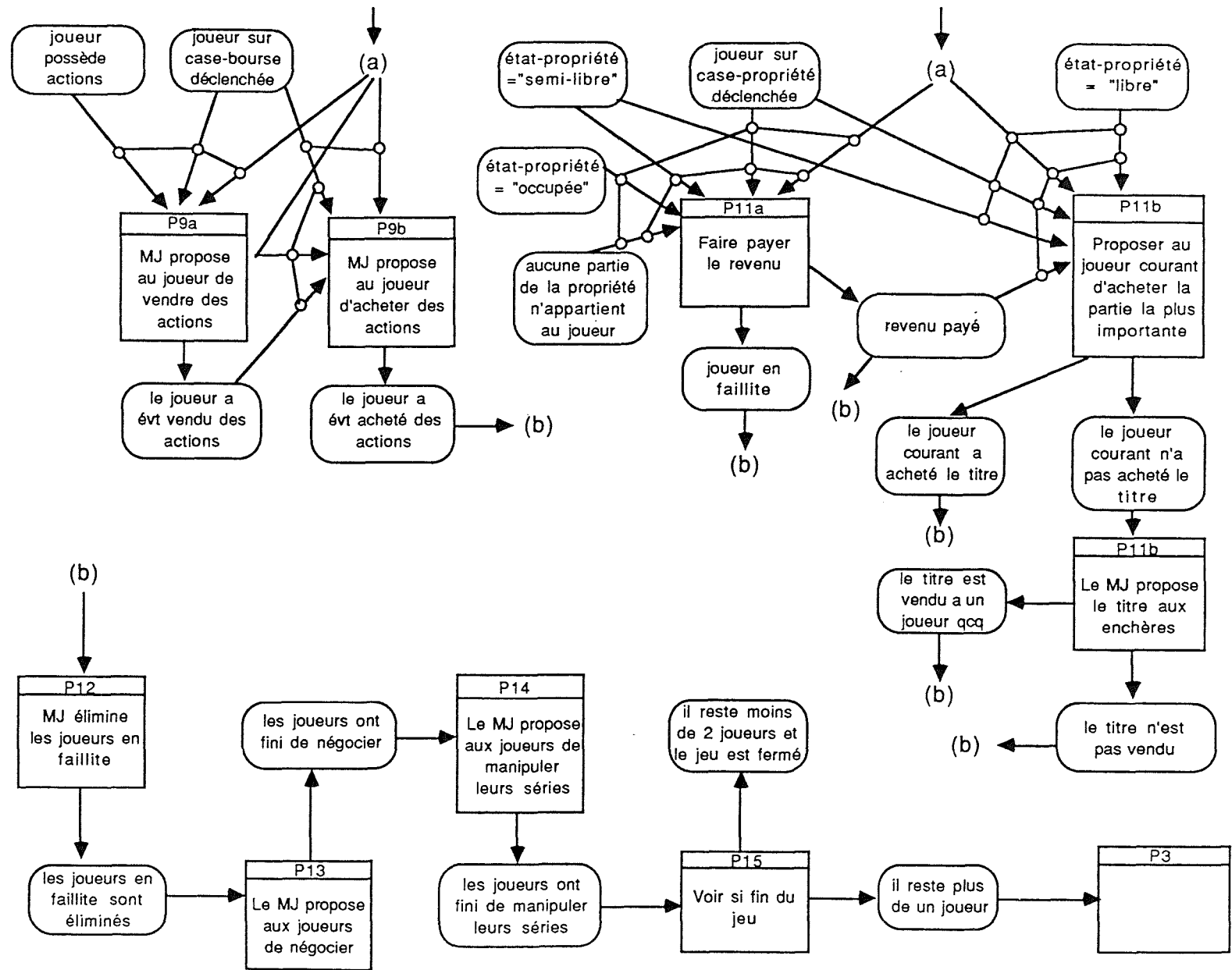
Nous disposons maintenant d'une démarche de spécification que nous avons adaptée à nos besoins. Nous avons cependant un sentiment de grande redondance dans la méthodologie. La question qui vient directement à l'esprit est dès lors : une rationalisation ne serait-elle pas possible? C'est ce que nous tenterons de voir au point suivant.

4.3. ETUDE D'UNE RATIONALISATION DE LA METHODE

La première question que l'on pourrait se poser est : que rationaliser? Afin de pouvoir y répondre, essayons de repérer dans la méthode les étapes où l'on croit constater une redondance avec d'autres étapes ou dans lesquelles une simplification serait peut-être possible. Ainsi, nous avons pu identifier trois domaines candidats potentiels à une rationalisation.

Le premier vise à un rapprochement du réseau de transition entre jalons et des cycles de vie des acteurs; le second à une rationalisation dans la découpe du scénario; et le troisième à une analyse complète des

Figure IV.9 (c) - Diagramme de transition du MENEUR DE JEU.



données du problème en une seule étape au lieu de trois. Passons maintenant en revue et de manière plus précise ces trois domaines.

4.3.1. Rapprochement du réseau de transition entre jalons et des cycles de vie des acteurs

On peut se demander pourquoi penser à un tel rapprochement. La raison en est très simple. Un coup d'oeil sur les deux types de graphes suffit à comprendre : l'armature de représentation semble particulièrement se ressembler.

Avant de voir si un rapprochement est possible entre ces graphes, rappelons en quelques mots ce qu'ils représentent. Le réseau de transition entre les jalons représente les différents cheminements possibles à travers le scénario. Les cycles de vie, pour leur part, spécifient toutes les transitions possibles d'états des acteurs. Nous avons donc d'un côté (celui du réseau de transition entre les jalons) les notions de jalon et de tableau, et de l'autre côté (celui des cycles de vie des acteurs) les notions d'état des acteurs et de transition entre ces états.

Afin de voir si le rapprochement désiré est plausible, regardons tout d'abord si les notions évoquées ci-dessus sont compatibles entre elles. En effet, si ces dernières sont incompatibles, nous pourrions abandonner immédiatement toute tentative de fusion car les concepts représentés seraient totalement différents. Par contre, si les concepts représentés sont compatibles, et par conséquent se ressemblent fortement, une étude plus approfondie permettrait peut-être de réaliser le rapprochement voulu.

Dans le réseau de transition entre jalons, un jalon constitue, comme nous l'avons vu au point 4.4.1., un état particulier d'une partie ¹¹ du scénario. Il faut entendre ici par "état particulier d'une partie du scénario", non pas un état particulier d'un des acteurs à un moment particulier du scénario, mais un ensemble d'états des différents

¹¹ Le terme "partie" est à comprendre ici comme un fragment quelconque du scénario.

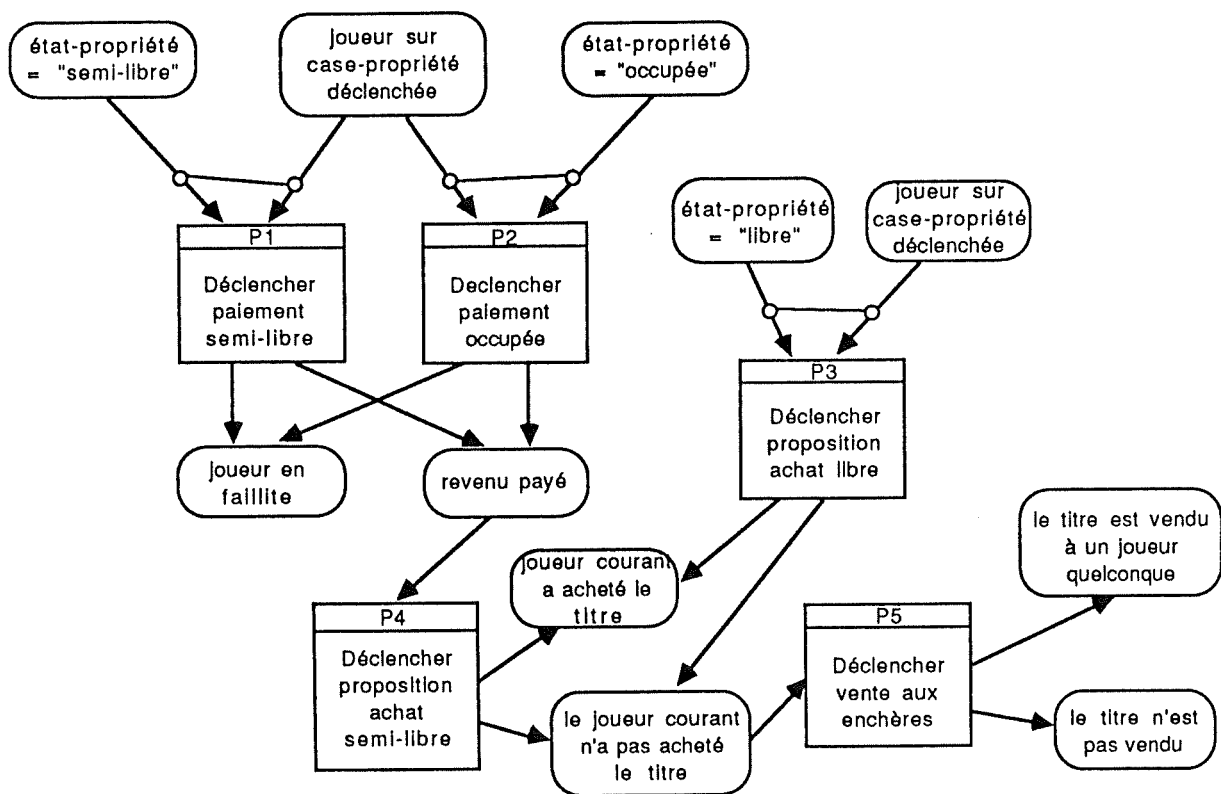


Figure IV.11 - Diagramme de transition de la CASE-PROPRIETE.

acteurs, sachant que le terme "état" doit être compris ici comme une valeur d'un attribut (au sens Entité-Relation [BODA-89]). Un tableau exprime pour sa part l'action à effectuer à partir d'une situation initiale (c'est-à-dire un jalon) pour aboutir à une situation finale (également un jalon). Il vise donc à modifier un état (au sens large) particulier d'une partie du scénario en un autre.

Dans les cycles de vie des acteurs, on trouve pour chaque acteur les différents états par lesquels il peut passer. Dans ce cas, nous considérons le terme "état" au sens restreint de valeur de l'attribut "état" de cet acteur. Les flèches du graphe ne représentent pour leur part que les transitions possibles entre ces différents états.

Nous identifions donc des deux parts des notions d'états (étendue en un sens large pour le réseau de transition entre les jalons et restreinte pour les cycles de vie des acteurs). En outre, nous trouvons également une notion de transition entre états. Cependant, dans les cycles de vie des acteurs, les flèches ne représentent pas une action bien précise, comme c'est le cas pour le réseau de transition entre les jalons (les tableaux). Il est toutefois possible de remédier à ce problème en désignant pour chacune des transitions le ou les tableaux qui provoque(nt), entre autre, le changement d'état de l'acteur. Ainsi, les concepts manipulés deviendraient compatibles.

Nous préférons cependant abandonner cette proposition, et par là même renoncer au rapprochement du réseau de transition entre les jalons et les cycles de vie des acteurs. Ce rapprochement nous occasionnerait un surplus de travail et n'apporterait aucun avantage. En effet, les différents états (au sens restreint) des acteurs représentés dans les cycles de vie sont également représentés dans les états particuliers (au sens large) des parties du scénario, autrement dit par les jalons.

Un rapprochement du réseau de transition entre les jalons et les cycles de vie des acteurs, et par conséquent intégration de tous les états (au sens restreint) de tous les acteurs dans le même graphe y créerait une surcharge et induirait de ce fait l'illisibilité de celui-ci. Or nous recherchons par cette rationalisation une simplification de la

procédure de spécification et une lisibilité accrue, choses que cette proposition n'amène manifestement pas.

4.3.2. Rationalisation dans la découpe du scénario

Comme nous l'avons vu au point 4.2.3. (cf étapes 3° et 4°), l'idée maîtresse de la méthode de spécification proposée par de Looz et Jourquin dans [DEL-JOU] est celle d'un scénario obtenu par raffinement progressif. Nous avons également remarqué qu'une séquence et un tableau peuvent représenter la même chose ¹². Il suffit pour cela que la séquence originale ne soit pas raffinée. Dès lors, on peut considérer qu'il existe une redondance entre ces différentes découpes puisque ces séquences sont automatiquement muées en tableaux et à nouveau ajoutée dans la liste de la découpe du scénario en tableaux. Sur les 29 séquences relevées dans notre système, seules 3 ont été affinées et, par conséquent, 26 furent automatiquement muées en tableaux. On imagine aisément l'ampleur que cette redondance pourrait prendre dans d'énormes systèmes.

Afin d'éviter cette redondance, nous avons imaginé d'opérer une découpe en tableaux uniquement pour les séquences ayant été raffinées. La découpe en tableaux reprendrait donc seulement quelques fragments du scénario devant être raffinés. Toutes les autres séquences ne subiraient aucune modification. Le document de base ne serait plus dans ce cas la découpe en tableaux mais la découpe en séquences et la découpe en tableaux opérée sur certaines séquences. Il est donc aussi nécessaire de repérer dans la découpe en séquences celles qui ont été raffinées et dont on trouvera une découpe en tableaux. Pour cette raison, nous les feront précéder dans la découpe en séquences par un signe distinctif (§ par exemple).

Cette solution n'apporte pas seulement des avantages. Nous allons ci-dessous repérer les inconvénients qu'elle occasionne.

¹² Ceci vaut également pour le couple partie-séquence, mais dans une moindre mesure

Le premier, et le plus facilement repérable, est évidemment la nécessité de consulter deux documents au lieu d'un seul. Ainsi, la lecture commence sur le document contenant la découpe en séquences du scénario pour se poursuivre autant de fois qu'il existe de séquences développées sur celui contenant la découpe en tableaux.

Un second problème est celui posé par la numérotation. En effet, comment réaliser celle-ci de manière non ambiguë et de telle manière que l'on puisse repérer facilement si l'on a affaire à une séquence ou à un tableau? Nous voudrions en outre que cette numérotation indique le fait qu'un ensemble de tableaux est le raffinement d'une séquence (et vice versa) ¹³. Il semble donc que celle-ci serait complexe, ou du moins assez lourde.

Au delà de ce second problème se trouve un autre, plus fondamental celui-là. Le réseau de transition entre jalons que nous serions amené à construire mêlerait à la fois des séquences et des tableaux. Si l'on accepte cependant cet écart par rapport à la démarche originale, un autre concept va devoir à nouveau être créé afin de distinguer, dans ce réseau, les séquences des tableaux. Toutefois, ce concept pourrait certainement être introduit (et résolu) par le biais de la numérotation. Cette dernière semble donc être un des problèmes cruciaux.

Nous allons proposer, à titre d'exemple, une numérotation répondant à notre volonté de rationalisation. Nous nous baserons sur la séquence 110 de la découpe du scénario en séquences :

S110 (J4 à J4.1) : Le joueur avance d'autant de cases que le nombre obtenu au lancé de dés et déclenche le comportement éventuel d'une case en passant par dessus.

Ayant décidé que, entre autre, la séquence 110 sera affinée, la découpe du scénario en séquences devient :

¹³ Ceci s'avère nécessaire puisque nous travaillons avec deux documents

...

S90 (J3 à J3.1) : Le MJ désigne le joueur courant.

S100 (J3.1 à J4) : Le joueur courant lance les dés.

‡ S110 (J4 à J4.1) : Le joueur avance d'autant de cases que le nombre obtenu au lancé de dés et déclenche le comportement éventuel d'une case en passant par dessus.

...

La deuxième liste des jalons supplémentaires reste, elle, identique à celle que nous avons obtenue lors de l'étape 4° de la démarche de spécification :

S110 : J4 : Le joueur a lancé les dés.

J4.0.1 : Le joueur a avancé d'une case.

J4.0.2 : Si la case est une case de prêt et que le joueur a prêté de l'argent sur cette case lors d'un tour précédent, la case est passée à l'état "autorisation de réclamer" pour ce joueur.

J4.0.3 : Si la case est la case de départ, le MJ a déposé 100000\$ dans le fonds boursier et le joueur a reçu 5% du fonds boursier par carte action qu'il possède.

J4.1 : Le joueur a avancé du nombre de cases désigné par les dés en déclenchant des comportements éventuels de cases en passant par-dessus.

Les tableaux résultant des transitions entre ces nouveaux jalons seront numérotés de la façon suivante :

T<num-séquence-aff>.<num-tab>

où <num-séquence-aff> indique le numéro de la séquence affinée,
<num-tab> indique le numéro du tableau.

Ainsi, la numérotation, quoique assez lourde, répond à nos attentes, à savoir : - elle est non ambiguë (car bien définie) et permet de repérer que l'on a affaire à un tableau;

- elle permet d'indiquer que le tableau provient du raffinement de la séquence <num-séquence-aff> (et vice versa).

La découpe en tableaux résultante est donc :

- T110.10 (J4 à J4.0.1) : Le joueur avance d'une case.
- T110.20 (J4.0.1 à J4.0.1) : Si la case n'est pas une case de prêt ou de départ et que le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
- T110.30 (J4.0.1 à J4.0.2) : Si la case est une case de prêt et que le joueur a prêté sur cette case, l'état de la case passe à "autorisation de réclamer" pour ce joueur.
- T110.40 (J4.0.1 à J4.0.3) : Si la case est la case de départ, le MJ dépose 100000\$ dans le fonds boursier et le joueur reçoit 5% du fonds boursier par carte action qu'il possède.
- T110.50 (J4.0.2 à J4.0.1) : Si le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
- T110.60 (J4.0.3 à J4.0.1) : Si le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
- T110.70 (J4.0.1 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
- T110.80 (J4.0.2 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
- T110.90 (J4.0.3 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
- ... (indiquant les autres découpes en tableaux résultant des différents affinements de séquences)

La distinction dans le réseau de transition entre jalons des séquences et des tableaux sera également faite grâce à cette numérotation.

Mais au vu des inconvénients qu'apporte cette proposition, nous préférons, une fois encore, abandonner cette idée. Nous conservons donc la numérotation simple et une découpe finale unique du scénario (celle en tableaux) sur un seul document.

4.3.3. Analyse des données

C'est après être passé par plusieurs étapes différentes que l'on peut considérer avoir obtenu une analyse complète des données. En effet, lors de l'étape 2 de la démarche de spécification, on commence

par chercher, pour chaque acteur identifié, une liste de ses principales caractéristiques. Cette liste est révisée et une nouvelle description des acteurs et de leurs caractéristiques est rédigée à l'étape 4. Enfin, nous aboutissons, à l'étape 5, à une spécification complète des données.

Nous avons donc l'impression de traîner un problème en longueur, d'autant plus qu'une analyse complète des données du problème est de toute façon opérée en dernier recours (cf étape 5). Dès lors, ne vaudrait-il pas mieux commencer directement par celle-ci afin de partir d'emblée avec la totalité des données (de toute façon nécessaires pour l'étape d'implantation)?

Cette proposition semble d'autant plus intéressante qu'elle devrait avoir des répercussions sur les cinq premières étapes de la démarche de spécification. Supposons donc que cette analyse complète soit réalisée au préalable et voyons les implications que celle-ci entraîne sur les cinq premières étapes (les autres n'étant pas concernées car partant toutes deux d'une analyse des données supposée complète).

La première étape (c'est-à-dire la description de l'idée originale ou d'un synopsis pour le scénario) ne subit vraisemblablement aucune modification car elle ne vise qu'à décrire de manière sommaire les grandes étapes du problème à spécifier. Mais elle peut cependant être influencée dans ce sens qu'une connaissance approfondie des données peut induire la rédaction d'un synopsis plus détaillé que prévu.

La deuxième étape (c'est-à-dire l'identification des acteurs et la description de leurs principales caractéristiques) n'a plus de raison d'être puisque tous les acteurs (et leurs caractéristiques) ont déjà été identifié(e)s. Cependant, comme nous l'avons vu dans la spécification du jeu de rôle FRIK, cette étape permet en outre de spécifier sous forme de phrases les rôles des différents acteurs dans le système. Cet aspect de l'étape pourrait donc être conservé pour des raisons évidentes de clarté.

La troisième étape (c'est-à-dire la spécification du scénario en définissant les jalons et la découpe en parties du scénario) risque elle aussi d'être plus approfondie car certaines caractéristiques (et certains acteurs) qui normalement auraient du être trouvées plus tard dans la démarche le sont déjà, et par conséquent influencent la découpe en partie du scénario.

Pour la même raison, la quatrième étape (c'est-à-dire la révision de la description des acteurs et les propositions de nouvelles découpes du scénario, et l'établissement du réseau de transitions entre les jalons du scénario) risque, elle aussi, d'être influencée par l'analyse préalable des données. En outre, il n'y aura évidemment plus de liste des acteurs et caractéristiques des acteurs supplémentaires (induisant la définition de nouveaux sous-jalons), et donc, les affinements successifs se feront sur base de choix personnels puisque ceux-ci ne seront plus guidés par celles-ci.

Enfin, la cinquième étape (c'est-à-dire la spécification des acteurs) est elle aussi influencée par cette analyse complète des données. La plupart des attributs (d'instances, et de classe ou de système) auront déjà été retirés de cette analyse, mais c'est seulement à cette étape que l'on pourra donner les valeurs qu'ils peuvent prendre. C'est en effet uniquement après la quatrième étape que le concepteur est à même de fournir les différents états que peut prendre tout acteur. Il en est de même pour l'attribut `compt[acteur]` (et `compt*[acteur]`) puisqu'il requiert la connaissance de tous les jalons du scénario (et des mini-scénarios des "jalons-étoiles") et de tous les tableaux.

Faut-il donc procéder de préférence à une analyse complète et préalable des données du problème à spécifier à l'analyse progressive de celles-ci à travers plusieurs étapes? Là est en effet la question. Celle-ci permettrait d'aborder la spécification en toute connaissance de cause (du moins pour les données). Les cinq premières étapes se verraient ainsi simplifiées (cf ci-dessus) et effectuées avec un maximum de renseignements. Mais l'on peut rétorquer à une telle argumentation que le propre de la démarche de spécification est d'être progressive. Elle préconise le raffinement et permet de cette manière d'aboutir également à une analyse des données complète en temps voulu (les données complètes ne sont requises qu'à partir de la cinquième étape).

Nous préconiserons pour notre part une analyse des données préalable. Nous pourrions de ce fait partir dès le début d'une bonne base de connaissance des données du problème à spécifier. Mais nous préconisons la rédaction d'un document qui ne se veut pas définitif et auquel il est possible d'ajouter à tout moment des renseignements supplémentaires au fur et à mesure des étapes de la démarche de

spécification rencontrées ¹⁴. Nous préservons ainsi l'esprit à l'origine de la méthode de spécification, celui d'une démarche progressive.

Nous avons présenté et adapté dans ce chapitre la méthode de spécification originale proposée par de Looz et Jourquin dans [DEL-JOU]. Nous avons en outre tenté de la rationaliser. Nous allons maintenant proposer au point suivant une étape supplémentaire à celle-ci, se situant au terme des sept étapes exposées jusqu'à présent : la spécification par prédicats.

4.4. UNE ETAPE SUPPLEMENTAIRE : LA SPECIFICATION PAR PREDICATS

Dans un premier temps, nous donnerons les explications théoriques à la base de cette nouvelle étape. Ensuite, dans un deuxième temps, nous l'illustrerons à l'aide d'exemples puisés dans la spécification du jeu de rôle FRIK. Dans un troisième temps, enfin, nous proposerons une réflexion basée sur cette spécification par prédicats.

4.4.1. Explication théorique

Nous commencerons par donner une définition d'un prédicat. Nous expliquerons ensuite pourquoi nous jugeons nécessaire d'introduire cette nouvelle étape dans la méthodologie de spécification et nous verrons sur quoi celle-ci porte. Finalement, nous examinerons la forme du résultat de cette étape supplémentaire.

¹⁴ Rappelons que de toute façon il n'est pas possible de définir toutes les valeurs de certains attributs avant la cinquième étape de la démarche de spécification.

a) Définition d'un prédicat

Un **prédicat** est une proposition ou une expression qui prend la valeur vraie ou fausse en fonction de ses arguments.

b) Pourquoi une spécification par prédicats?

Quelles sont les raisons qui nous ont poussé à introduire cette nouvelle étape dans la méthodologie de spécification?

Nous possédons, à l'issue de la septième étape de la méthodologie présentée dans ce chapitre, deux outils extrêmement importants pour l'implémentation. Il s'agit des plans de système et des diagrammes de transition des différents acteurs.

Les premiers permettent une vision globale pour l'implantation des classes des acteurs puisqu'ils présentent pour chaque processus de chaque acteur les flux d'informations existant entre lui et les autres acteurs (c'est-à-dire avec son environnement). En outre, ils indiquent également pour chaque processus les canaux d'informations existant avec les différentes accumulations relevées dans le système de l'acteur.

Les seconds, pour leur part, aident à l'implantation et à l'agencement des méthodes. C'est à ces derniers que nous nous intéresserons plus particulièrement. Nous partirons principalement d'eux pour l'implantation car ils rendent compte de la dynamique d'évolution du système via leur dimension temporelle.

Nous retrouvons dans ceux-ci des états exprimés à l'aide de petites phrases (souvent en style télégraphique), états permettant l'activation des processus s'ils sont vérifiés, ou résultant de l'activation d'un de ces processus (et dans ce cas, ils sont évidemment vérifiés). Il est donc nécessaire de les spécifier plus formellement. C'est pourquoi nous avons décidé de les formaliser en termes de prédicats prédéfinis.

c) Forme du résultat de la spécification par prédicat

Le résultat obtenu à l'issue de cette étapes est constitué de trois parties. La première, appelée "**Dictionnaire des prédicats utilisés**", est constituée de la liste de tous les prédicats utilisés dans les deux autres parties et de leur définition. La deuxième est constituée d'une liste contenant la spécification en termes de prédicats prédéfinis des états communs aux différents acteurs, c'est-à-dire apparaissant dans les diagrammes de transition d'au moins deux acteurs différents. La troisième, enfin, est constituée d'une liste pour chaque acteur spécifiant par prédicats ses états propres.

4.4.2. Illustration

Pour illustrer cette nouvelle étape, nous proposons de spécifier par prédicats les états communs aux acteurs suivants : "Le joueur courant a lancé les dés" et "Joueur sur case-prêt normale". Nous poursuivrons par la spécification des états propres à l'acteur MENEUR DE JEU suivants : "Jeu ouvert, joueurs prêts à jouer" et "Il reste moins de 2 joueurs et le jeu est fermé". Mais avant toute chose, définissons les prédicats dont nous aurons besoin.

- Extrait du dictionnaire des prédicats utilisés :

Etat-Dé (i, j) : l'état du Dé-1 = i et l'état du Dé-2 = j , $1 \leq i, j \leq 6$.

Etat-GA (Etat-x) : le GA est dans l'état Etat-x.

Etat-Jeu (Etat-x) : le jeu est dans l'état Etat-x.

Etat-Joueur (Joueur-x, Etat-y) : le joueur Joueur-x est dans l'état Etat-y.

Joueur-Désigné-GA (Joueur-x) : le joueur Joueur-x est désigné par le GA.

Nbre-Avancé (l) : le nombre de case dont le joueur courant a déjà avancé est l.

Nbre-Jeu (Nbre) : il y a un nombre Nbre de joueurs dans le jeu.

Somme-Dés (k) : la somme des dés (=i+j) vaut k.

- Extrait de la spécification par prédicats des états communs aux acteurs :

Le joueur courant a lancé les dés :

Joueur-Désigné-GA (joueur-x)

ET Etat-Joueur (joueur-x, courant)

ET Etat-Dés (i, j)

ET Somme-Dés (k)

ET Nbre-Avancé (0)

Joueur sur case-prêt normale :

Etat-Joueur (joueur-x, courant)

ET Position (joueur-x, case-prêt, p)

ET Etat-Case (p, normale)

- Extrait de la liste de spécification par prédicats des états propres aux acteurs (ici, le meneur de jeu)

Jeu ouvert, joueurs prêts à jouer :

Etat-Jeu (ouvert)

ET Etat-Joueur (joueur-x, prêt à jouer) avec $2 \leq x \leq \text{Nbre-joueur}$

ET Etat-GA (en proposition)

Il reste moins de 2 joueurs et le jeu est fermé :

[Nbre-Jeu (0) ET Etat-Jeu (fermé)]

OU [Nbre-Jeu (1) ET Etat-Jeu (fermé)]

4.4.3. Réflexions sur les prédicats

Alors que la spécification formelle des jalons est utile pour repérer tous les attributs ainsi que pour retrouver de nouveaux états, la spécification par prédicats se situe, elle, à un niveau d'abstraction plus élevé. Dans cette dernière, tous les changements d'attributs ne sont pas mentionnés s'ils ne sont pas nécessaires à l'expression même de l'état. On pourrait donc se poser, dans une certaine mesure, la question de savoir pourquoi ne pas faire la spécification par prédicats avant la spécification formelle des jalons, celle-ci devenant par la force des choses un simple raffinement de celle-là.

La question pourrait cependant être posée de manière différente : pourrait-on, suite à une analyse du texte (et, dans notre cas, du

règlement du jeu) sortir tels quels les différents prédicats ? La réponse est, semble-t-il, non, et ceci parce que le texte est très mal structuré. Peut-être serait-ce possible si les règles étaient elles-mêmes déjà un scénario, c'est-à-dire si elles étaient énoncées dans l'ordre dans lequel elles apparaissent dans le jeu.

En outre, rappelons que cette spécification se base sur la description des diagrammes de transition des différents acteurs. Ceux-ci mettent en oeuvre des processus associés chacun à un ou plusieurs tableaux. Or les tableaux sont des actions visant à faire passer le système d'une situation initiale à une situation finale, c'est-à-dire d'un jalon à un autre jalon. Les jalons doivent donc être définis (et non nécessairement spécifiés formellement) avant que les diagrammes de transition puissent être dessinés, et donc qu'une spécification par prédicat puisse être opérée.

D'autre part, nous avons besoin d'une structure de données complète. Elle est supposée telle à l'issue de la cinquième étape. Mais c'est lors de la sixième étape (c'est-à-dire la spécification formelle des jalons) qu'un contrôle supplémentaire permet, le cas échéant, de la compléter. En effet, si l'on n'arrive pas à spécifier formellement un (ou plusieurs) des jalons, cela signifie qu'il manque des données. Et n'oublions pas non plus que cette formalisation permet aussi de retrouver des états que l'on n'avait pas relevés, ou de supprimer ceux qui étaient superflus.

Il semble donc que notre première opinion s'avérait correcte : la spécification formelle des jalons est un préalable à la spécification par prédicats, les deux spécifications étant en outre l'une et l'autre nécessaires.

4.5. INTERET DES DIFFERENTS DOCUMENTS REDIGES POUR L'IMPLANTATION

Nous allons relever dans ce point l'intérêt que suscite chaque document généré pour l'implantation.

Le cheminement dans la méthode nous amène en effet, étape après étape, à une spécification complète du système et utile lors de l'implantation.

Ainsi, le synopsis nous permet de comprendre quel est l'objet de l'implantation. L'identification des acteurs permet une première approche de l'ensemble des intervenants du système. La découpe en tableaux du scénario permet d'avoir un bon agencement des comportements. Les listes d'attributs nous permettent de connaître la structure interne des acteurs et de regrouper ces derniers en classes, tandis que les cycles de vie définissent des contraintes de changements d'état possibles d'une méthode à l'autre. La spécification formelle des jalons fournit quant à elle un outil pour l'activation des méthodes associées aux objets. Enfin, les plans de système permettent une vision globale pour l'implantation des classes des acteurs tandis que les diagrammes de transition aident quant à eux à l'implantation et à l'agencement des méthodes.

Nous constatons donc que toutes les étapes traversées ne le sont pas en vain. Au contraire, elles jouent toutes un rôle dans le cadre de l'implantation du système.

Nous avons exposé et adapté dans ce chapitre une méthode de spécification d'un jeu de rôle. Nous avons ensuite tenté de la rationaliser. Nous avons en outre proposé une huitième étape à cette méthode : la spécification par prédicats.

Nous possédons donc, à l'issue de ces huit étapes, une spécification complète du jeu de rôle qui nous a servi d'illustration, le jeu FRIK. (cf annexe 3). Il nous reste maintenant à décrire la démarche d'implantation. Elle fera l'objet du chapitre suivant.

Chapitre V : APPROCHE THEORIQUE D'IMPLANTATION BASEE SUR LA NOTION DE SCENARIO

5.1. INTRODUCTION

Au terme des huit étapes proposées au chapitre précédent, nous possédons une spécification complète du jeu de rôle. Le chapitre présent aborde, pour sa part, la méthode mise en oeuvre quant à l'implantation du système résultant de celle-ci.

Nous allons donc présenter ci-dessous la démarche d'implantation (basée bien entendu sur la notion de scénario puisque nous partons des résultats obtenus précédemment) proposée par de Looz et Jourquin dans [DEL-JOU] et largement inspirée de [SVT].

Nous commencerons par exposer la démarche proprement dite, démarche qui reste valable quel que soit le langage de classe choisi (le choix des langages de classe a été opéré au chapitre 3).

Nous terminerons par une étude un peu plus approfondie concernant l'implantation des méthodes mêmes.

5.2. DEMARCHE D'IMPLANTATION

La démarche générale d'implantation (en un langage de classe), telle que proposée par de Looz et Jourquin, se base sur les résultats obtenus par la méthode de spécification exposée au chapitre précédent. Elle comporte cinq étapes :

- 1° Initialisation du système;
- 2° Identification des classes;
- 3° Description de l'état des objets;
- 4° Etablissement de la liste des interfaces des objets;
- 5° Implantation des méthodes.

Nous présenterons, pour chaque étape, une explication théorique. Celle-ci sera suivie d'une illustration puisée, une fois encore, dans le jeu de rôle FRIK.

Passons donc maintenant en revue les différentes étapes de cette démarche.

5.2.1. Initialisation du système

a) Explication théorique

Cette première étape se divise en deux sous-étapes.

La première consiste en la définition du domaine d'animation en terme de fenêtres à l'écran. On définira donc la fenêtre et les sous-fenêtres dans lesquelles l'animation graphique s'inscrira, en s'inspirant de l'idée originale issue de la première étape de la démarche de spécification.

La deuxième sous-étape consiste, pour sa part, à créer les formes graphiques associées aux acteurs mobiles. On s'aidera pour ce faire de l'attribut de classe repr[acteur] qui contient toutes les données nécessaires.

b) Illustration

Nous proposons de diviser la fenêtre principale en deux sous-fenêtres. La plus grande, dans la partie inférieure de l'écran, sera une fenêtre de type graphique et sera le siège de l'animation. La plus petite, dans la partie supérieure de l'écran, sera une fenêtre de type texte et décrira ce qui se passe dans la fenêtre graphique.

Nous avons cependant dû revoir la présentation de la fenêtre graphique. Il est en effet impensable d'afficher le plateau de jeu complet sur celle-ci. Nous avons donc opté pour la solution suivante : seule la case sur laquelle se situe le joueur courant sera affichée à l'écran (qu'il soit arrêté ou en progression). De ce fait, plus aucun

DescriptionPane : fenêtre de type texte		
AnimationPane : fenêtre graphique	CartePane : fenêtre de type texte donnant les données relatives à la case	TbordPane : fenêtre de type texte affichant les cartes que possède le joueur courant
ArgentPane : fenêtre de type texte permettant d'afficher divers renseignements		

Figure V.1 - Fenêtres à l'écran.

déplacement n'aura lieu et la fonction de la caméra se résumera à de "simples" affichages.

Nous n'avons donc plus à créer de formes graphiques pour les acteurs mobiles puisqu'ils seront tous considérés comme fixes. La représentation de ces derniers se fera sous forme de méthodes implantées.

Nous décidons en outre que les différentes cartes n'apparaîtront pas dans la fenêtre graphique. Elles apparaîtront dans des fenêtres de type texte.

Ceci nous amène donc à redessiner la fenêtre principale de la manière suivante. Nous aurons toujours dans la partie supérieure de l'écran la fenêtre de type texte décrivant ce qui se passe dans les autres fenêtres. La fenêtre graphique sera fortement réduite et sera le siège de l'animation. Une fenêtre de type texte affichera les différentes données relatives à la case sur laquelle le joueur courant se sera arrêté (si une telle carte existe, bien entendu). Une troisième fenêtre de type texte servira quant à elle à afficher toutes les cartes que le joueur courant possède déjà. Enfin, une dernière fenêtre de type texte permettra d'afficher divers renseignements, comme la somme d'argent que possède le joueur courant, etc. La figure V.1 présente cette fenêtre principale.

5.2.2. Identification des classes

a) Explication théorique

Cette seconde étape vise à identifier les classes issues de la démarche de spécification et ne figurant pas dans l'environnement initial de programmation.

Nous rangerons les classes dans les grandes catégories suivantes :

- le décor
- les acteurs mobiles
- les autres acteurs

- le meneur de jeu
- la caméra

Le décor est constitué de tous les acteurs visibles fixes et des éléments décoratifs. Les acteurs mobiles s'identifient généralement aux personnages. D'autres acteurs peuvent cependant en faire partie. Les autres acteurs sont les acteurs du domaine d'animation n'entrant pas en compte dans les deux catégories précitées (en général, ce sont des acteurs invisibles). Le meneur de jeu et la caméra répondent aux définitions qui leurs ont été données précédemment. Rappelons toutefois que la caméra est un sous-acteur du meneur de jeu. Elle correspond également à une de ses fonctions. C'est pour cette raison que celle-ci sera plutôt implantée via une méthode. On ne la retrouvera donc pas dans la liste des classes.

b) Illustration

Le décor est constitué du seul plateau de jeu. Celui-ci est en fait représenté par l'acteur jeu.

Comme nous l'avons vu précédemment, il n'y a pas d'acteurs mobiles. Nous créerons cependant une classe appelée `Prototype1` et rassemblant tous les acteurs visibles mais que l'on ne peut considérer comme fixes car non-toujours présent à l'écran (il ne peuvent de ce fait être inclus dans le décor). Les acteurs entrant dans cette classe sont : les joueurs, les différentes cases et cartes, les lingots, les séries, les dés et le fonds-boursier.

Dans la classe des autres acteurs (intitulée `ActeurInvisible`), nous trouverons les acteurs prêts, enchères et négociations.

Enfin, les caractéristiques du meneur de jeu (et du GA !), y compris celles de la caméra, seront prises en compte au sein de la classe `MeneurDeJeu`, qui sera la super-classe de toutes les autres classes du système car le meneur de jeu est en quelque sorte un acteur possédant toutes les données du système.

La hiérarchie de classes dégagée de cette identification des classes est la suivante :

MeneurDeJeu
DecorPrototype1
Jeu
Prototype1
Joueur
Case
CasePropriete
CaseBourse
CaseArt
CasePret
CaseCourtage
CaseOr
CaseDepart
CaseHasard
Carte
CartePropriete
CarteBanque
Action
Tableau
PA
Hasard
Lingot
Serie
De
FondsBoursier
ActeurInvisible
Pret
Encheres
Negociations

5.2.3. Description de l'état des objets

a) Explication théorique

Lors de cette étape, nous allons définir l'ensemble des variables d'instance et de classe liées aux objets identifiés précédemment.¹⁵

D'un point de vue pratique, tous les attributs d'instance relevés lors de la cinquième étape de la démarche de spécification basée sur la notion de scénario seront implantés comme des variables d'instance.

Un tableau référençant chaque occurrence de la classe sera pour sa part implanté sous forme de variable de classe. Son nombre d'éléments est donné par l'attribut de classe nbre[acteur] issu également de la cinquième étape de la démarche de spécification (cf point 4.2.3., 5° et annexe 3). Les autres attributs de classe ou de système ne sont pas pris en compte ici.

" D'autres variables d'instance ou de classe peuvent également être implantées. Il en est ainsi de variables permettant de faire des références à des objets de la nouvelle hiérarchie du système, de variables correspondant à des contraintes de création de fenêtre, de gestion du système, etc. " ([DEL-JOU], p. 76). Ainsi, selon le cas, une ou des variable(s) d'instance et/ou de classe du Meneur de Jeu seront implantées afin de référencer la ou les fenêtre(s) graphique(s), la ou les fenêtre(s) de texte, ainsi que les différents supports des fenêtres graphiques.

b) Illustration

Nous illustrerons cette étape en définissant l'état des objets de la classe Joueur.

¹⁵ Rappelons qu'une variable de classe est une variable qui est unique pour tous les objets de la classe. Une variable d'instance est propre à une instance d'une classe et existe donc conceptuellement autant de fois qu'il y a d'objets dans la classe. ([DEL-JOU])

- Définition des variables d'instance :

NumJoueur, EtatJoueur, MontantJoueur, NomJoueur, NumCarteProJoueur, NumCarteBqJoueur, NumSerieJoueur, PossActNum, PossTabNum, PossPretNum, PossLingotNum, PossPAnum, PossHasNum, PositionJoueur.

- Définition des variables de classe :

Nous définissons TabJoueur référençant toutes les occurrences de Joueur.

5.2.4. Etablissement de la liste des interfaces des objets

a) Explication théorique

Cette quatrième étape vise à établir la liste des messages auxquels chacun des objets identifiés précédemment doit pouvoir répondre. Nous pouvons rassembler ces différents messages en trois grandes catégories :

- " Les messages liés au système adopté lui-même; ce sont généralement des messages de classe permettant de créer une instance, de la mettre à jour, de la référencer, etc.
- Les messages hérités des "super-classes" de l'objet en question.
- Les messages propres à l'objet en question et qui traduisent un comportement. " ([DEL-JOU], p. 77)

Concrètement, on établira pour chaque classe des messages permettant de créer une instance, de la mettre à jour (c'est-à-dire qu'on créera un message pour chaque variable de classe et d'instance susceptible d'être modifiée), de la référencer, etc.

En outre, pour chaque comportement non-décomposé relevé dans les attributs compt[acteur] et compt*[acteur] issu de la cinquième étape de la démarche de spécification, un message sera créé. On veillera toutefois à ne pas définir un tel message s'il correspond seulement à une modification (c'est-à-dire à une mise à jour) d'une des variables de l'objet puisqu'un tel message a déjà été établi (cf ci-dessus).

Enfin, il en sera de même pour tous les processus dégagés à l'étape sept de la démarche de spécification. Ceux-ci correspondent en effet à

un ou plusieurs tableaux et représentent donc un ensemble de comportements du système.

A l'issue de cette étape, on disposera donc d'une liste de messages liés à chaque classe du système. Cette liste ne se veut pas définitive car, pour des raisons de lisibilité, de facilité ou de nécessité, d'autres méthodes (et donc messages) peuvent apparaître en cours d'implantation. On veillera cependant à les intégrer à la liste initiale des messages, afin que la documentation soit la plus complète possible.

b) Illustration

Nous allons énumérer ici quelques-uns des messages de la classe Joueur (les étoiles indiquent les messages de classe).

+ nouveau*	: crée une nouvelle instance de Joueur.
+ etat:	: met à jour l'état du receveur.
+ initTab*	: déclare la variable de classe TabJoueur.
+ cartePro:	: ajoute une carte de propriété au receveur.
+ payerPA:	: payer et recevoir la PA proposée par le MJ.
+ passerDépart	: participer au passage sur la case départ.
+ argentOr:	: recevoir l'argent de la vente de l'or et rendre le lingot.
+ courLancer:	: devenir courant et lancer les dés.
+ ...	

5.2.5. Implantation des méthodes

a) Explication théorique

Cette ultime étape consiste à définir le contenu des méthodes associées aux messages identifiés lors de l'étape précédente. Il s'agit donc d'une étape de pure programmation dans le langage de classe choisi.

Cette étape peut être assez longue et nécessite en tout cas un esprit de synthèse assez élevé puisqu'il faut tenir compte à la fois de la spécification formelle des jalons, des diagrammes de transitions, des cycles de vie des acteurs, des règles de couplage et de la spécification

par prédicat. Nous verrons au point 5.3. comment agencer toutes ces informations obtenues lors de l'application de la démarche de spécification.

b) Illustration

Nous ne pourrions illustrer cette dernière étape de la démarche d'implantation sans recourir à un langage de programmation. Le choix s'est porté sur Smalltalk/V. On trouvera à l'annexe 4 une description sommaire de ses principales caractéristiques et de sa syntaxe.

Nous décrivons tout d'abord le contenu de la méthode "nouveau" de la classe Joueur et créant une nouvelle instance de Joueur :

```
nouveau
    "Crée une instance de Joueur"
    ^self new etat: #créé.
```

Nous prendrons comme deuxième exemple l'implantation du message "lancerDe" correspondant à un comportement du Joueur :

```
lancerDe
    "Faire lancer les dés"
    | varAl1 varAl2 |
    varAl1 := (SmallInteger random : 6) + 1.
    varAl2 := (SmallInteger random : 6) + 1.
    De1 etat : varAl1.
    De2 etat : varAl1.
    SommeDe etat : (varAl1 + varAl2).
    NbreAvance etat : 0.
```

5.3. ETAPE D'IMPLANTATION DES METHODES : UNE DEMARCHE?

Comme nous l'avons vu au point 5.2.5., l'étape d'implantation des méthodes induit la manipulation d'énormément d'informations en même temps. Nous allons essayer de voir ci-dessous s'il existe un moyen facile d'implémenter les méthodes. Nous indiquerons également la

manière de se servir des différents documents générés lors de l'application de la démarche de spécification.

Nous allons passer en revue la façon d'implanter les messages liés au système adopté lui-même et les messages propres à l'objet en question et traduisant un comportement. Les messages hérités des "super-classes" de l'objet entrent, pour leur part, dans une des deux catégories précitées.

5.3.1. Les messages liés au système adopté lui-même

L'implantation des méthodes associées à ces messages est assez évidente. Elle tient en effet de la structure même des données et est de ce fait assez technique. Aucun des documents générés précédemment ne nous servira, sauf, peut-être, la liste des attributs des acteurs afin de repérer facilement les variables (d'instance ou de classe) à mettre à jour ou à référencer.

5.3.2. Les messages représentant un comportement

a) Les comportements non-décomposés relevés dans les attributs `compt[acteur]` et `compt*[acteur]`

Le comportement concerné provenant d'un des deux attributs `compt[acteur]` ou `compt*[acteur]`, il possède donc une précondition et une postcondition puisqu'il correspond à un tableau du scénario.

De là, la spécification formelle des jalons nous indique quelles sont exactement les pré- et post-conditions attachées à ce comportement (rapellons que pré- et post-conditions sont toutes deux des jalons). Nous relevons en plus de cette manière les acteurs impliqués par la méthode.

En outre, les cycles de vie des acteurs nous indiquent les seules transitions possibles entre états. Nous considérons donc ceux-ci comme des vérificateurs supplémentaires.

Il ne reste plus au concepteur qu'à faire en sorte que la postcondition soit vérifiée à l'issue de la méthode. Il veillera en plus à ce que les règles de couplages soient appliquées, le cas échéant.

Afin d'illustrer ceci, reprenons le message "lancerDe" dont le texte Smalltalk/V figure en point 5.2.5. et voyons comment nous en sommes arrivé là.

Ce message correspond au comportement "lancer-dés" apparaissant dans l'attribut de classe compt[joueur]. Il a pour précondition le jalon J3.1 et pour postcondition le jalon J4.

La spécification formelle des jalons nous révèle que les jalons ci-dessus correspondent à :

J3.1 : Etat-joueur[Ji] = "courant"
ET Etat-joueur[Jj] = "en attente" avec $j \neq i$
ET Joueur-désigné[GA] = Ji

J4 : Etat-dé = i, $1 \leq i \leq 6$
ET Somme-dé[Jeu] = s
ET Nbre-avancé[Jeu] = 0

Il en résulte donc le texte Smalltalk/V figurant au point 5.2.5.

```
lancerDe
    "Faire lancer les dés"
| varA1 varA2 |
varA1 := (SmallInteger random : 6) + 1.
varA2 := (SmallInteger random : 6) + 1.
De1 etat : varA1.
De2 etat : varA1.
SommeDe etat : (varA1 + varA2).
NbreAvance etat : 0.
```

b) Les messages correspondant aux processus

La façon d'implanter les processus est quelque peu différente. Nous savons en effet qu'un processus correspond à un ou plusieurs tableaux et représente donc un ensemble de comportements du système. Nous connaissons déjà de cette manière les messages (correspondant à des comportements non-décomposés) dont nous aurons besoin.

Le plan de système indique les flux de communication existant entre acteurs. Il indique donc l'existence d'un message à adresser à chacun de ces acteurs dans le cadre de l'exécution de ces processus.

Mais pour qu'un processus puisse être activé (et donc exécuté), certains états doivent être vérifiés. Ceux-ci apparaissent dans le diagramme de transition de l'acteur concerné par le processus. Il n'appartient cependant pas au processus de vérifier ces états mais bien à la fonction "conduire scénario" du Meneur de Jeu. Par contre, c'est bien à lui qu'incombe la tâche de vérification des états en sortie.

C'est donc ici qu'intervient la huitième étape de la démarche de spécification, la spécification par prédicats. Avant de déclencher les processus des différents acteurs, le Meneur de Jeu doit donc vérifier que les états permettant l'activation du processus sont bien vérifiés. En outre, le concepteur devra veiller à ce qu'après l'exécution d'un processus, les états en sortie soient bien atteints.

Le diagramme de transition de l'acteur concerné permet en outre de relever les informations en provenance ou à destination des acteurs externes (c'est-à-dire de son environnement).

Nous disposons donc maintenant de toutes les informations nécessaires à l'implantation du processus. Il appartient maintenant au concepteur de la réaliser.

Pour l'illustration, nous nous inspirerons de l'implantation du processus P3 "Devenir le joueur courant et lancer les dés" du Joueur (cf annexe 3).

Ce processus correspond aux tableaux T90 et T100 (cf annexe 3). Nous aurons donc besoin des messages "etat:" et "lancerDe".

Le plan de système indique qu'il y a un flux d'information avec le Meneur de Jeu (c'est normal car c'est lui qui déclenchera le processus P3 et surtout c'est lui qui indiquera le joueur qui sera courant).

Afin que le processus puisse être activé, il faut que les états "le joueur a éventuellement acheté une PA" et "MJ a désigné ce joueur courant" soient vérifiés, mais ces informations n'intéressent que le Meneur de Jeu. Par contre, à l'issue de l'exécution du processus, l'état "joueur courant a lancé les dés" doit être vérifié (cf la spécification par prédicats).

Nous pouvons donc maintenant implanter la méthode correspondant au processus P3. Le texte en Smalltalk/V est le suivant :

```
courLancerDe
    "Permet d'indiquer le joueur courant et de lancer les dés"
    | jouCour |
    jouCour := (MeneurDeJeu joueurDesigné) asInteger.
    (Joueur occ: jouCour) etat: #courant.
    Joueur lancerDe.
```

A l'issue de l'exécution du processus, l'état "le joueur courant a lancé les dés" est bien vérifié. La méthode répond donc bien à ses spécifications.

On trouvera, à titre d'exemple, une partie de l'implantation des méthodes relatives à la caméra à l'annexe 5.

Il est maintenant temps de prendre un peu de recul par rapport à la (double) démarche proposée aux chapitres IV et V, de poser un regard critique sur celle-ci et de proposer des prolongements possibles. Ceci fera l'objet de notre ultime chapitre, le chapitre VI.

Chapitre VI : REFLEXIONS ET CRITIQUES

6.1. INTRODUCTION

Ce chapitre a pour objectif d'évaluer la (double) méthode proposée dans les chapitres précédents et d'exposer les perspectives et prolongements envisageables à ce mémoire.

6.2. EVALUATION DE LA METHODE

Notre étude a démontré que la notion de scénario pouvait servir de support méthodologique en matière de conception d'animation graphique d'un jeu de rôle sur ordinateur. Tous les éléments nécessaires à une telle animation nous sont en effet fournis par la double démarche couvrant à la fois les aspects de spécification et d'implantation.

Nous avons vu que la méthode originale présentée par de Looz et Jourquin dans [DEL-JOU] était déjà assez puissante. Suite aux problèmes rencontrés (notamment dans la méthode de spécification), nous avons proposé des solutions permettant de les résoudre, tout en conservant, bien entendu, l'idée originale de la démarche "orientée-scénario". Nous avons en outre apporté des précisions concrètes quant à la démarche d'implantation en un langage (de classe) orienté objet.

Certaines difficultés se sont cependant présentées lors de l'application de cette méthode, indiquant les différents points faibles de celle-ci.

Il s'est ainsi avéré qu'il est parfois assez ardu de représenter un concept. Ce fut le cas des "Séries" dans notre système. Fallait-il les représenter par une relation (Entité-Relation entendu) ou par une entité? Etaient-elles des acteurs visibles ou invisibles?

D'autre part, l'oubli d'un certain nombre de détails en cours de spécification peut poser d'énormes inconvénients. En effet, la démarche est constituée d'une série d'étapes à appliquer successivement. Toute modification dans l'une de celles-ci entraîne des changements dans celles qui lui sont ultérieures. Ces dernières doivent donc être réappliquées. Toute modification radicale implique donc souvent une perte de temps et une perte de tout (dans le pire des cas) ou partie du travail effectué depuis lors. Ainsi, par exemple, l'apparition d'un acteur important ou d'une caractéristique importante d'un des acteurs lors de la spécification formelle des jalons (étape 6° de la démarche de spécification) peut obliger le concepteur à modifier les différentes découpes du scénario (étapes 3° et 4° de la démarche de spécification).

S'il subsiste encore quelques problèmes, il est un avantage indéniable à attribuer à la démarche que l'on ne pourrait cacher, celui de permettre de retrouver des acteurs oubliés, ainsi que des caractéristiques de ces derniers qui ne sont pas visibles de prime abord. Ceci constitue bien entendu un élément crucial et capital pour une démarche de spécification, quelle qu'elle soit.

6.3. PERSPECTIVES ET PROLONGEMENTS ENVISAGEABLES

Le premier prolongement possible est de terminer l'implantation complète du système spécifié. Celle-ci demande cependant un travail considérable. Nous aurions ainsi la preuve que la méthode exposée est digne d'intérêt et permet d'arriver à ses fins.

Un deuxième prolongement possible serait de gérer les déplacements des joueurs sur le plateau de jeu, et donc sur les cases, de manière déterministe, en fonction d'un scénario spécifié par des expérimentateurs, préalablement à la session de jeu. Les situations de jeu seraient de cette manière reproductibles. Les joueurs ne connaissant pas ces scénarios (tout se passe évidemment à leur insu) auront l'impression que les déplacements sont déterminés aléatoirement.

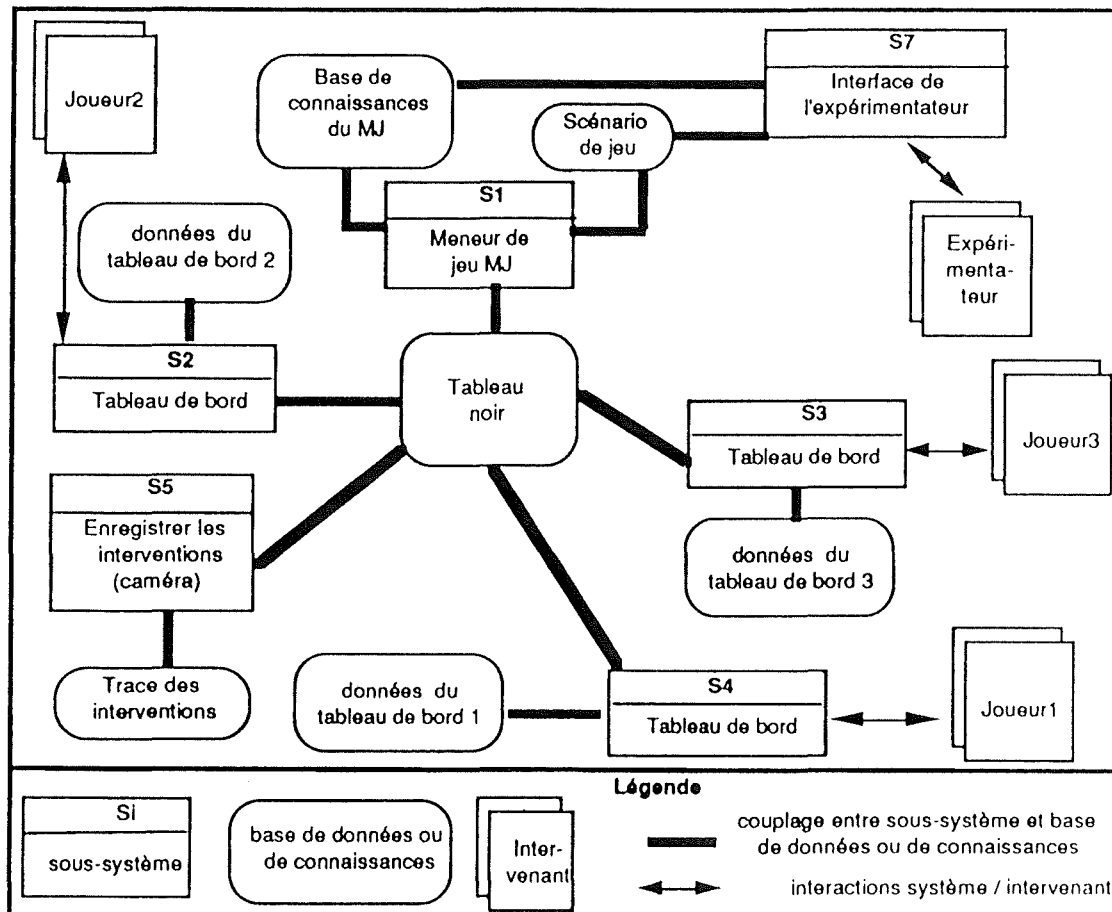


Figure VI.1 - Architecture générale du système envisagé.

Ce deuxième prolongement ne serait pas très intéressant s'il n'entrait dans le cadre d'une perspective plus générale. Ainsi, outre la gestion des déplacements et des affichages, la caméra pourrait se voir confier une fonction supplémentaire, celle d'enregistrer toutes les interactions entre les joueurs (informatisés ou humains), et avec le meneur de jeu. Les expérimentateurs pourraient de cette manière analyser les résultats enregistrés par la caméra et les comparer avec d'autres résultant d'une même situation de jeu (c'est-à-dire provenant d'un même scénario déterministe). Sachant que chaque joueur disposera d'une interface appelée "tableau de bord" lui permettant d'interagir avec le système, l'architecture générale du système envisagé est présentée à la figure VI.1 ¹⁶. Mais vu le nombre de données à afficher pour chaque tableau de bord, seul l'un d'entre-eux peut figurer à la fois à l'écran. Cela pose évidemment un problème de consultation vis-à-vis de la totalité des données. C'est la combinaison de la caméra enregistreuse et du scénario déterministe qui donne à cette perspective son caractère général.

Enfin, une autre perspective intéressante, mais de grande envergure, pourrait consister en l'élaboration d'une version du jeu de rôle informatisé tournant sur un réseau. Chaque joueur posséderait de cette manière son propre tableau de bord (avec possibilité de consulter ceux des autres joueurs et le plateau de jeu en entier) sur son propre écran. L'expérimentateur pour sa part posséderait sa propre interface, lui permettant à tout instant de modifier certains paramètres du système et d'analyser les résultats au fur et à mesure du déroulement de la partie. Cette version semble particulièrement se rapprocher de la réalité.

Ainsi, l'architecture matérielle et logicielle globale s'inscrivant dans cette perspective (cf figure VI.2) pourrait consister en un réseau reliant plusieurs serveurs pour un même client (l'application représentant le jeu de rôle). L'expérimentateur pourrait, via son serveur spécial, changer les paramètres du système et visualiser les

¹⁶ Cette perspective et l'architecture du système envisagé sont inspirées de [MOUL-90].

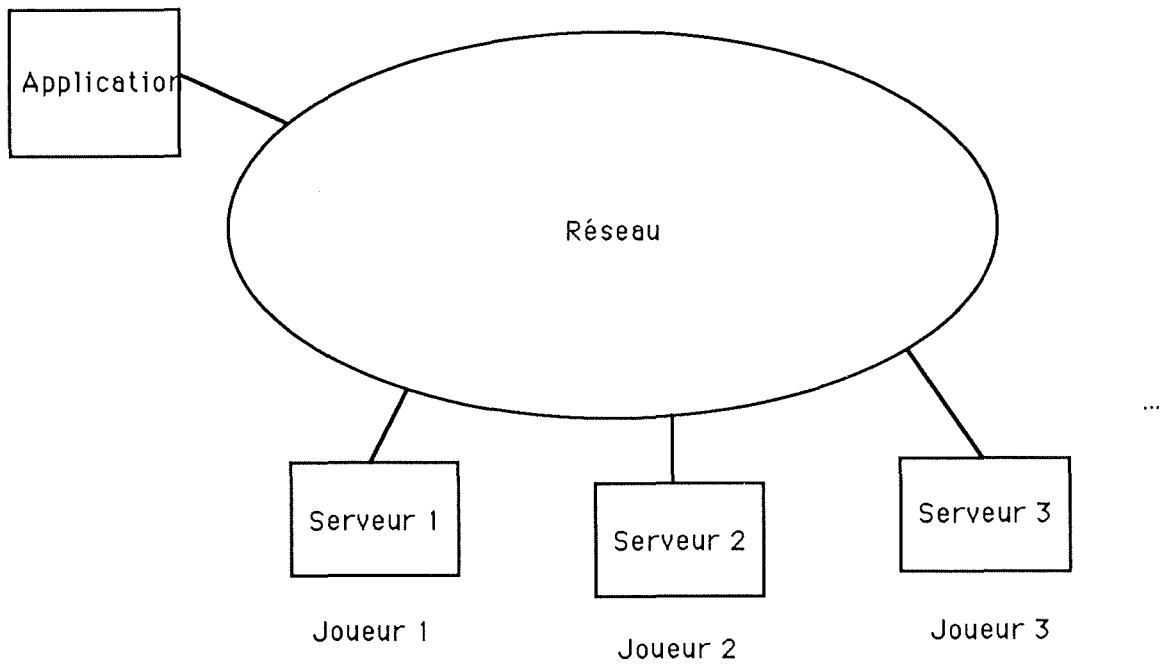


Figure VI.2 - Architecture matérielle pour la version réseau du jeu de rôle.

enregistrements effectués par la caméra. Les différents joueurs pourraient pour leur part interagir avec le système chacun à leur tour et pourraient visualiser leur tableau de bord ou celui d'un adversaire à tout instant. Un système de multi-fenêtrage et un système indiquant les priorités seraient bien entendu requis pour l'accomplissement de cette perspective. Nous n'entrerons cependant pas ici dans les détails, ceux-ci dépassant le cadre de ce mémoire.

CONCLUSION

L'idée instigatrice de ce mémoire était d'explorer l'utilisation d'une méthode "orientée-scénario" développée par de Looz et Jourquin pour la spécification d'un jeu de rôle en Smalltalk/V (cf chapitre I).

Pour mener à bien ce projet, nous avons exploré au chapitre II la littérature existante. Ce chapitre nous a permis de découvrir l'analogie existant entre le monde de l'audio-visuel et le problème d'animation graphique nous concernant, analogie déjà développée dans [DEL-JOU].

Ensuite, au chapitre III, nous avons posé un choix quant au type de langages de programmation le mieux adapté au problème : les langages de classes.

Le chapitre IV nous a permis d'exposer la méthodologie de spécification basée sur la notion de scénario et de lui adjoindre des modifications à partir des problèmes rencontrés lors de son utilisation. Nous avons en outre envisagé une rationalisation de celle-ci. Enfin, nous avons proposé d'ajouter une étape supplémentaire à la démarche de spécification originale : la spécification par prédicats.

Disposant de cette spécification complète, le chapitre V nous a permis d'exposer une approche théorique d'implantation basée, elle aussi, sur la notion de scénario. De plus, nous avons dégagé les informations (issues de la démarche de spécification) nécessaires à l'implantation même des méthodes.

Les chapitres IV et V ont été abondamment illustrés à l'aide d'exemples tirés du jeu de rôle FRIK.

Enfin, le chapitre VI a proposé quelques réflexions et critiques par rapport à la double démarche exposée aux chapitres IV et V. Nous avons notamment constaté que la méthode convenait pour la spécification d'un jeu de rôle en Smalltalk/V. Nous avons en outre considéré une première proposition de généralisation de notre travail. Celle-ci consiste en une combinaison de l'enregistrement des différentes interactions entre les joueurs et le meneur de jeu avec des déplacements déterministes des joueurs. Nous avons enfin envisagé la perspective d'une version du jeu de rôle informatisé tournant sur un réseau local.

BIBLIOGRAPHIE

[BODA-89]

Bodart F. et Pigneur Y., Conception assistée des applications informatiques, Edition Masson, 1989.

[DEL-JOU]

de Looz-Corswarem G. et Jourquin P., Méthode de spécification et d'implantation de systèmes d'animation graphique (volumes I et II), Institut d'Informatique, Facultés Universitaires Notre-Dame de la Paix, Namur, 1990.

[GUTT-77]

Gutttag J., Abstract data types and the development of data structures, Communications of the ACM, 20(6) p. 396 à 404, 1977.

[J&S]

Paunin T., Rosenthal P. et Sagot G., N'ayez plus peur des jeux de rôle!, in Jeux & Stratégie, n° 38, p. 32 à 43, Avril-Mai 1986.

[LEGA-75]

Le Garff A., Dictionnaire de l'informatique, Presses Universitaires de France, Paris, 1975.

[LISK-74]

Liskov B., Zilles S., Programming with abstract data types, Proc. SIGPLAN Symposium on Very High Level Languages, p. 50 à 59, 1974.

[MINS-75]

Minsky M., A Framework for Representing Knowledge, in P. Winston, editor, The Psychology of Computer Vision, p. 211 à 281, McGraw-Hill, New-York, 1975.

[MOUL-88]

Moulin B., La méthode E.P.A.S. pour la modélisation et la conception de systèmes (volume de cours et étude de cas), Département d'Informatique, Université Laval, Québec, 1988.

[MOUL-90]

Moulin B., Sujet de stage Automne 90 proposé par Bernard Moulin, feuillet volant, Septembre 1990.

[ROCH-87]

Roche C. et Laurent J.-P., Les approches objets et le langage LRO2 (KEOPS), TSI, vol. 8, n° 1, p. 21 à 39, 1989.

[STV]

Smalltalk/V Mac - Tutorial and Programming Handbook, Digitalk inc., Los Angeles, California, 1989.

ANNEXES

Annexe 1 : La méthode E.P.A.S.

Annexe 2 : Règlement du jeu de rôle FRIK.

Annexe 3 : Spécification complète du jeu de rôle FRIK.

Annexe 4 : Caractéristiques du langage Smalltalk/V.

Annexe 5 : Implantation de la caméra.

Annexe 1 :

La méthode E.P.A.S. pour la modélisation et la conception de systèmes (cf [DEL-JOU])

1. Introduction

Cette annexe consiste en une présentation de la méthode E.P.A.S. pour la modélisation et la conception de systèmes d'information. Cette méthode a été développée de 1982 à 1985 par l'équipe du professeur Moulin à l'Université Laval (Québec - Canada). Elle est présentée dans [MOUL-88].

Nous définirons d'abord les principaux concepts utilisés dans la méthode. Ensuite, nous analyserons le cheminement de modélisation et de conception proposé. Enfin, nous présenterons les principales techniques relatives à la modélisation. Nous terminerons par quelques remarques relatives à la méthode.

2. Définition des principaux concepts

La méthode E.P.A.S. a pour but de **modéliser** et de **concevoir** des **systèmes**. Tout d'abord, pour bien fixer les idées, nous allons définir ces termes selon la méthode E.P.A.S.. Ensuite, nous verrons ce que la notion de **plan** peut apporter par rapport au but de la méthode. Enfin, nous examinerons quelques principes sous-jacents à la méthode.

2.1. Système

Un système est un ensemble d'éléments en interaction dynamique, organisés en fonction d'un but. Il est vu comme un organisme, composé d'un ensemble de sous-systèmes.

2.2. Modéliser

Modéliser, c'est représenter une certaine réalité en utilisant un langage de description (graphique, mathématique, programmable, ...), choisi en fonction des objectifs de la modélisation.

2.3. Concevoir

Concevoir un système consiste, à partir des objectifs fixés et de l'analyse de la réalité, à modéliser le système et à valider son comportement pour pouvoir réaliser et mettre en oeuvre le système.

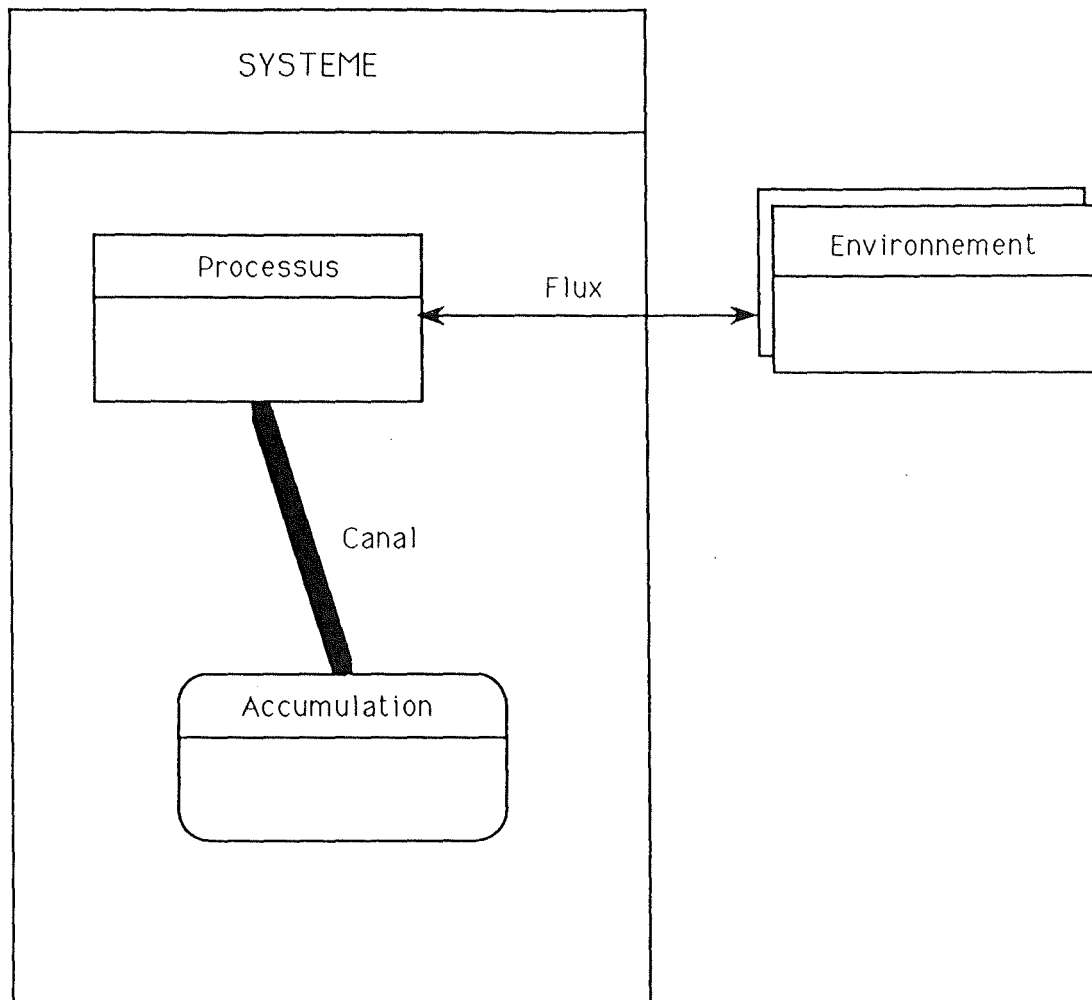


Figure A1.1. : plan (ou diagramme) E.P.A.S.

2.4. Plan

Le plan est ce qui permet d'agréger les concepts de modélisation, de conception et de système. En effet, un plan est un modèle présentant l'ensemble des composantes et leurs interactions. Le système est représenté par des plans, à partir desquels on peut détailler les caractéristiques de la mémoire et du comportement du système.

Un plan comporte les 5 types de composantes suivants :

1. des **processus**, qui sont des composantes actives, représentant le comportement du système. Ce comportement correspond à la dynamique d'évolution du système. Il est caractérisé par les actions effectuées par le système.
2. des **accumulations**, qui sont des composantes passives, représentant la mémoire du système. Cette mémoire caractérise l'état du système. L'état actuel du système est le résultat de l'évolution du système. La mémoire est un ensemble organisé de concepts, correspondant généralement à une vision statique de la réalité.
3. un **environnement**, qui est une composante externe, représentant les limites du système.
4. des **flux**, qui sont des communications entre le système et l'environnement. Les processus communiquent avec l'environnement du système sous forme d'événement ou d'ensemble d'événements.
5. des **canaux**, qui sont des associations entre processus et accumulations. Les processus mettent à contribution la mémoire du système en manipulant le contenu des accumulations.

Ces composantes sont présentées schématiquement à la figure A1.1.

Un système est souvent vu selon le modèle de la "boîte noire", c'est-à-dire qu'un système n'est rien d'autre qu'un processus de transformation de données avec des entrées et des sorties. Cette vue habituelle des systèmes (entrée-processus-sortie), conjuguée avec les deux composantes principales de la méthode (processus et accumulation), a engendré le nom E.P.A.S. pour cette méthode (Entrée-Processus-Accumulation-Sortie).

2.5. Principe de focalisation

La méthode E.P.A.S. utilise le principe de focalisation, qui permet d'observer et de modéliser du général au particulier. Ce principe est rendu possible grâce à des techniques d'abstraction et de changement de perspective.

2.6. Dimension de structure et dimension de fonctionnement

Les systèmes peuvent être étudiés et modélisés suivant deux perspectives différentes et complémentaires :

- une **dimension de structure** qui décrit le système comme un "objet existant". La description y est atemporelle : on ne se préoccupe pas du temps et seules les caractéristiques existentielles ont un intérêt. On y décrit des propriétés "statiques".
- une **dimension de fonctionnement** qui décrit le système comme un objet évoluant dans un contexte. La description y est temporelle : le temps est introduit dans le modèle et le principe de causalité est mis en oeuvre. Les caractéristiques d'évolution et de comportement du système sont modélisées. On y décrit des propriétés "dynamiques".

Nous verrons au point 4. des exemples de techniques appartenant à l'une ou à l'autre de ces dimensions.

3. Cheminement de conception

L'objectif de ce point est de présenter globalement les différentes phases du cheminement de modélisation et de conception de la méthode E.P.A.S.. Celles-ci sont au nombre de quatre : la phase d'**orientation** (qui part de la demande du client), la phase d'**analyse** (qui propose une solution à un niveau conceptuel), la phase de **conception** (qui présente la solution à un niveau fonctionnel) et la phase de **construction** (qui réalise le système qui sera livré).

3.1. Orientation : comprendre le problème

La phase d'orientation permet de poser clairement le problème, de modéliser la réalité concernée (c'est-à-dire le système existant, objet de l'étude), d'établir un diagnostic et de recommander une orientation à prendre en termes de solution(s) au problème analysé. Le modèle de la réalité concerne le "système-objet" auquel est rattaché le problème étudié.

3.2. Analyse : développer une solution à un niveau conceptuel

La phase d'analyse permet d'élaborer une solution à un niveau conceptuel, sans tenir compte des contraintes d'utilisation ou d'implantation du système.

A partir du modèle de la réalité et de la solution envisagée, on élabore la structure conceptuelle des données et les nouveaux plans du système objet. A partir de ces derniers, on élabore les plans du système cible et les maquettes des interfaces. Ensuite, à partir de la structure conceptuelle des données et des plans du système-cible, on élabore les diagrammes de

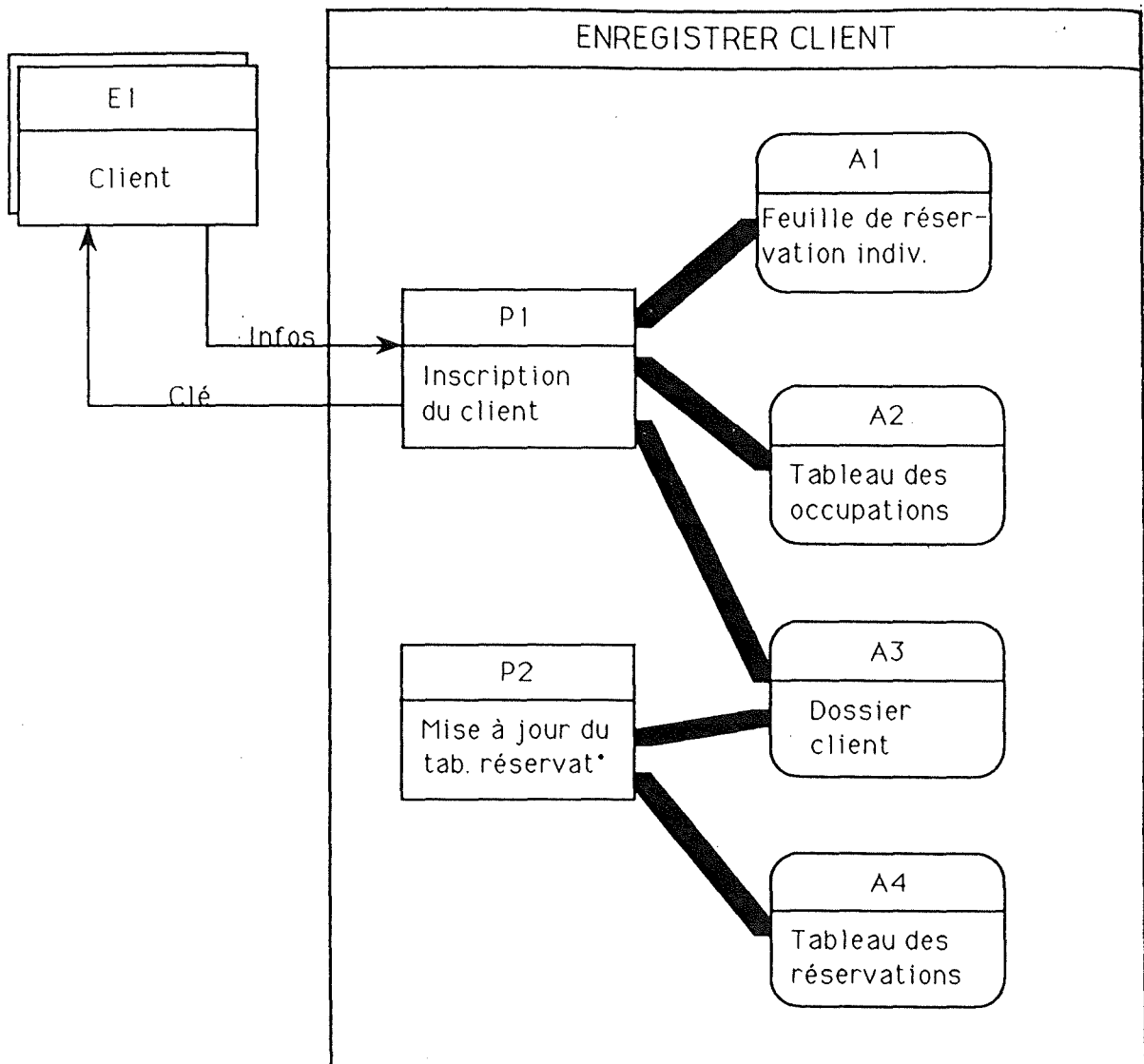


Figure A1.2. : illustration de plan de système

transition. On peut remarquer ici que différentes stratégies de conception interviennent pour orienter l'analyse, la conception et la construction du système étudié.

Les notions de plans de système, de structure conceptuelle des données et de diagrammes de transition seront détaillées au point 4.

3.3. Conception : concevoir la solution à un niveau fonctionnel

La phase de conception consiste à passer de la solution conceptuelle à la solution fonctionnelle, afin de donner des spécifications du système tenant compte des contraintes d'implantation et d'utilisation du système. On exprime ici comment le système fonctionnera.

Ainsi, en fonction de la solution décrite au niveau conceptuel, des décisions de conception et des contraintes de réalisation, les objectifs sont d'obtenir une description fonctionnelle du système-cible (spécification des structures de contrôle du logiciel, description du schéma logique des données, description des vues locales des données et spécification des composantes logicielles) et de planifier les jeux d'essai qui permettront de tester le logiciel lorsqu'il sera réalisé.

3.4. Construction : réaliser la solution et la tester

La phase de construction consiste à réaliser le système à partir des spécifications fonctionnelles.

Ainsi, compte tenu des spécifications de la phase de conception et des stratégies retenues, on procède à la réalisation des modules de programmation, à l'implantation du schéma physique des données, à la création des fichiers de données, à la réalisation des tests des programmes et à l'établissement des procédures d'opération du logiciel.

4. Principales techniques de modélisation

Nous avons introduit au point 3.2. quelques notions et techniques relatives à la modélisation. Le point 4. a pour objectif de définir ces techniques, à savoir les plans de système, la structure conceptuelle des données, les cycles de vie des objets et les diagrammes de transition.

4.1. Plans de système

Les plans permettent de modéliser la stratégie de comportement d'un système. Pour décrire celle-ci, il faut élaborer un modèle dans lequel on peut représenter les buts à atteindre, les activités et les contraintes de comportement ainsi que leurs interactions. Le plan constitue un

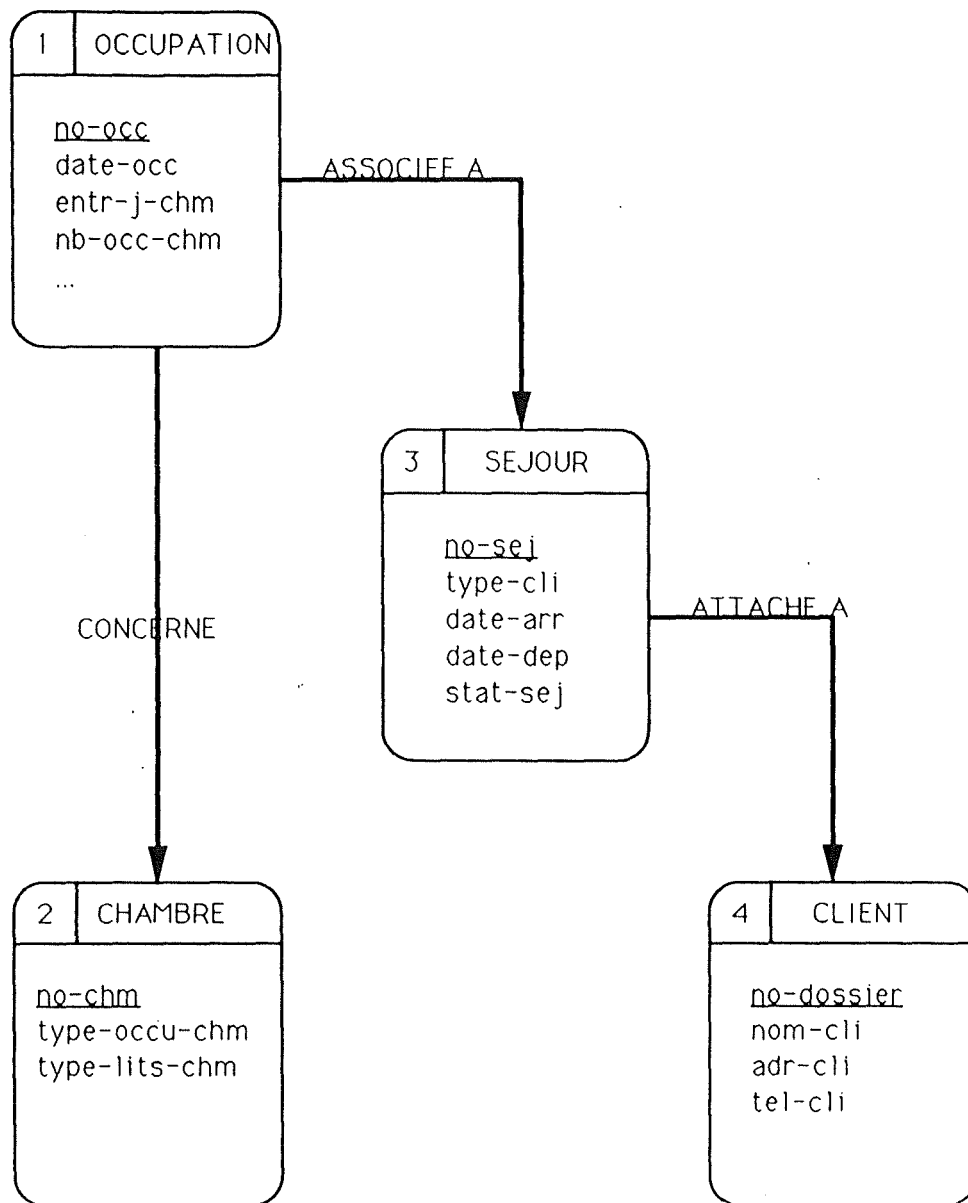


Figure A1.3. : illustration de structure conceptuelle de données

tel modèle. Un but est un état; une activité représente une action qui permet un changement d'état. Un plan décrit donc une méthode de changement d'état.

C'est ici qu'interviennent les concepts de processus, d'accumulation, d'environnement, de flux et de canal, introduits au point 2.4.

La dimension de structure de la méthode E.P.A.S. (cf point 2.6.) apparaît ici car les plans de système sont des représentations atemporelles des caractéristiques stratégiques du système.

Nous présentons à la figure A.1.2. une illustration de plan de système. Cet exemple, repris de [MOUL-88], décrit l'enregistrement d'un client à un hôtel. Pour ce faire, le processus "Inscription du client" (P1) est activé lorsque le client (E1) se présente pour s'enregistrer à l'hôtel. Les informations fournies par le client sont placées dans le dossier client (A3). Ce processus met à jour la feuille de réservation individuelle (A1) et le tableau des occupations (A2). En fonction des informations contenues dans le dossier client (A3), le processus "Mise à jour du tableau des réservations" (P2) met à jour le tableau des réservations (A4) s'il y a lieu; par exemple, si le client s'inscrit pour des journées supplémentaires sans qu'il y ait eu de réservation préalable.

4.2. Structure conceptuelle des données

La structure conceptuelle des données permet de modéliser l'organisation de la mémoire du système. La mémoire correspond à la faculté de conserver et de retrouver des éléments d'information. Elle est associative : les éléments d'information sont structurés afin de faciliter la mémorisation et la recherche des données.

La structure conceptuelle des données est décrite en tenant compte uniquement de la signification des données structurées (caractéristiques sémantiques). On ne se préoccupe pas à ce niveau de modéliser la façon dont la mémoire sera implantée et utilisée. Le paradigme de représentation des connaissances généralement utilisé est celui des réseaux conceptuels.

La structure conceptuelle des données peut être vue comme un ensemble d'objets (entités) reliés par des relations conceptuelles (relations). Chaque objet peut être vu comme la description statique d'un objet réel faisant partie du système-objet ou de son environnement.

Nous présentons un exemple de structure conceptuelle de données à la figure A1.3. Il est inspiré de [MOUL-88] et propose de manière simplifiée un réseau conceptuel présentant les entités "occupation" (1), "chambre" (2), "séjour" (3) et "client" (4) et leurs liaisons pour modéliser l'occupation de chambres lorsque les clients séjournent dans un hôtel.

4.3. Cycles de vie des objets

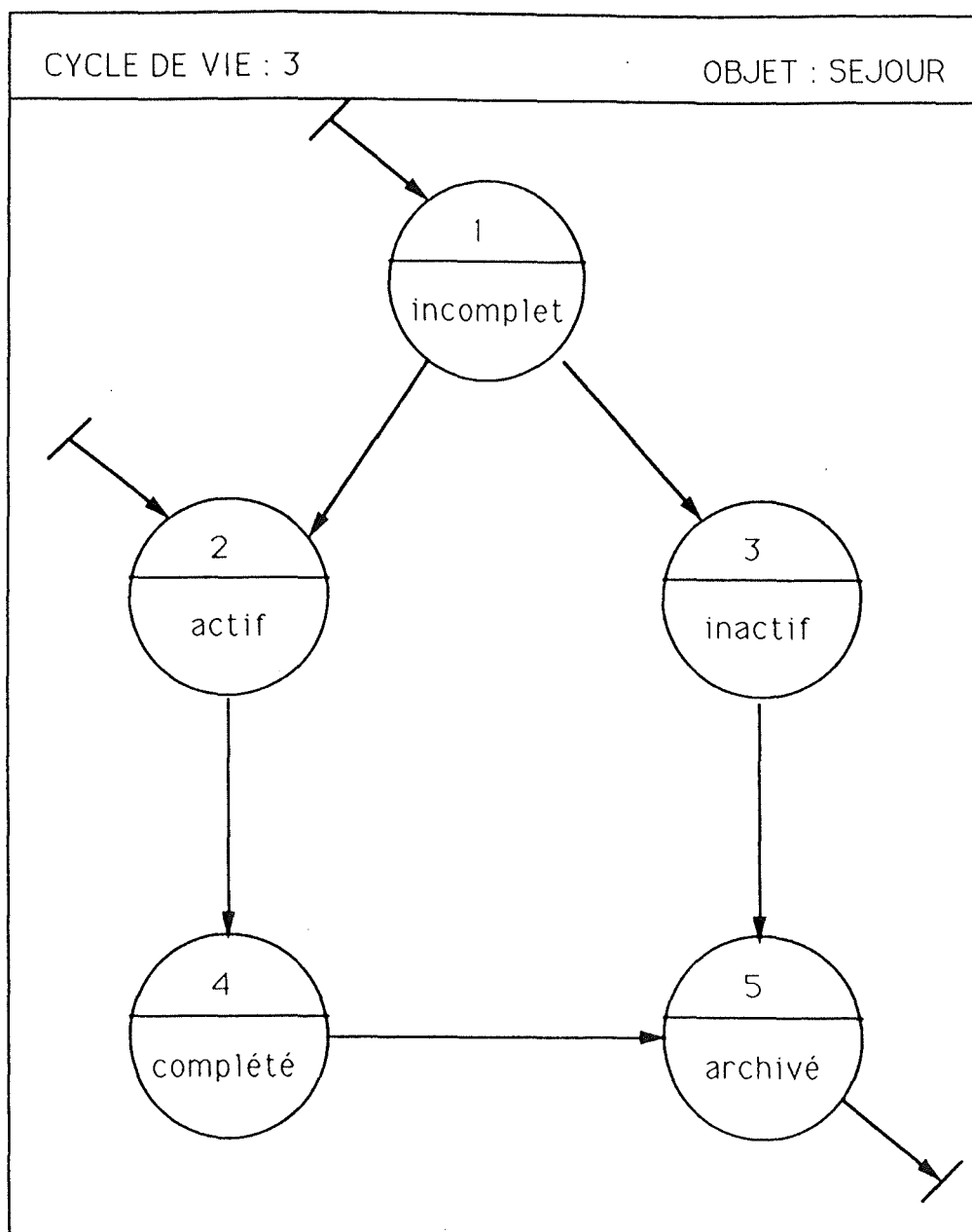


Figure A1.4 : illustration du cycle de vie d'un objet

La mémoire présentée sous la forme d'une structure conceptuelle de données fournit seulement une vision statique de l'organisation des données. Les cycles de vie des objets fournissent une vision dynamique. En effet, au cours de leur évolution, les occurrences d'un objet passent par divers états qu'il peut être nécessaire de connaître à des fins de contrôle ou pour déclencher un comportement particulier du système en réaction à cette évolution. Le cycle de vie d'un objet décrit les divers états et les règles de transition d'un état à un autre que peuvent subir les occurrences de l'objet.

En reprenant l'exemple du point précédent, nous pouvons envisager les différentes valeurs de l'attribut "stat-sej" de l'objet séjour : "incomplet" (1), "actif" (2), "inactif" (3), "complété" (4) ou "archivé" (5). La figure A1.4 décrit le cycle de vie pour cet objet.

4.4. Diagrammes de transition

Lorsque l'on veut décrire le comportement d'un système en tenant compte de la dimension temporelle, on élabore des scénarios de comportement représentés sous la forme de diagrammes de transition. Le scénario permet de fixer de façon déterministe l'ordre de précedence des activités du système : pour chaque activité, on précise les états du système et les événements issus de l'environnement qui doivent précéder la mise en oeuvre de l'activité ainsi que les états et les événements à destination de l'environnement qui doivent la suivre.

Les diagrammes de transition permettent d'intégrer mémoire et comportement au sein des plans du système-cible. Ils représentent des plans détaillés qui expriment la façon dont les processus du système-cible utilisent les objets de la mémoire du système, mettent en oeuvre les interfaces avec l'environnement et utilisent pour leurs propres besoins des accumulations qui sont locales.

Les diagrammes de transition constituent le niveau de plans les plus détaillés du système-cible et définissent au niveau conceptuel les comportements caractéristiques du système.

La dimension de fonctionnement de la méthode E.P.A.S. (cf point 2.6) apparaît clairement ici.

Si nous reprenons le processus "Inscription du client" de la figure A1.2, nous pouvons présenter un exemple de diagramme de transition. Ainsi, dans un système non informatisé, nous pourrions trouver le scénario suivant : un réceptionniste prend une feuille d'inscription et un stylo et note les informations contenues sur la carte d'identité du client. Ce scénario nous donnerait alors la figure A1.5.

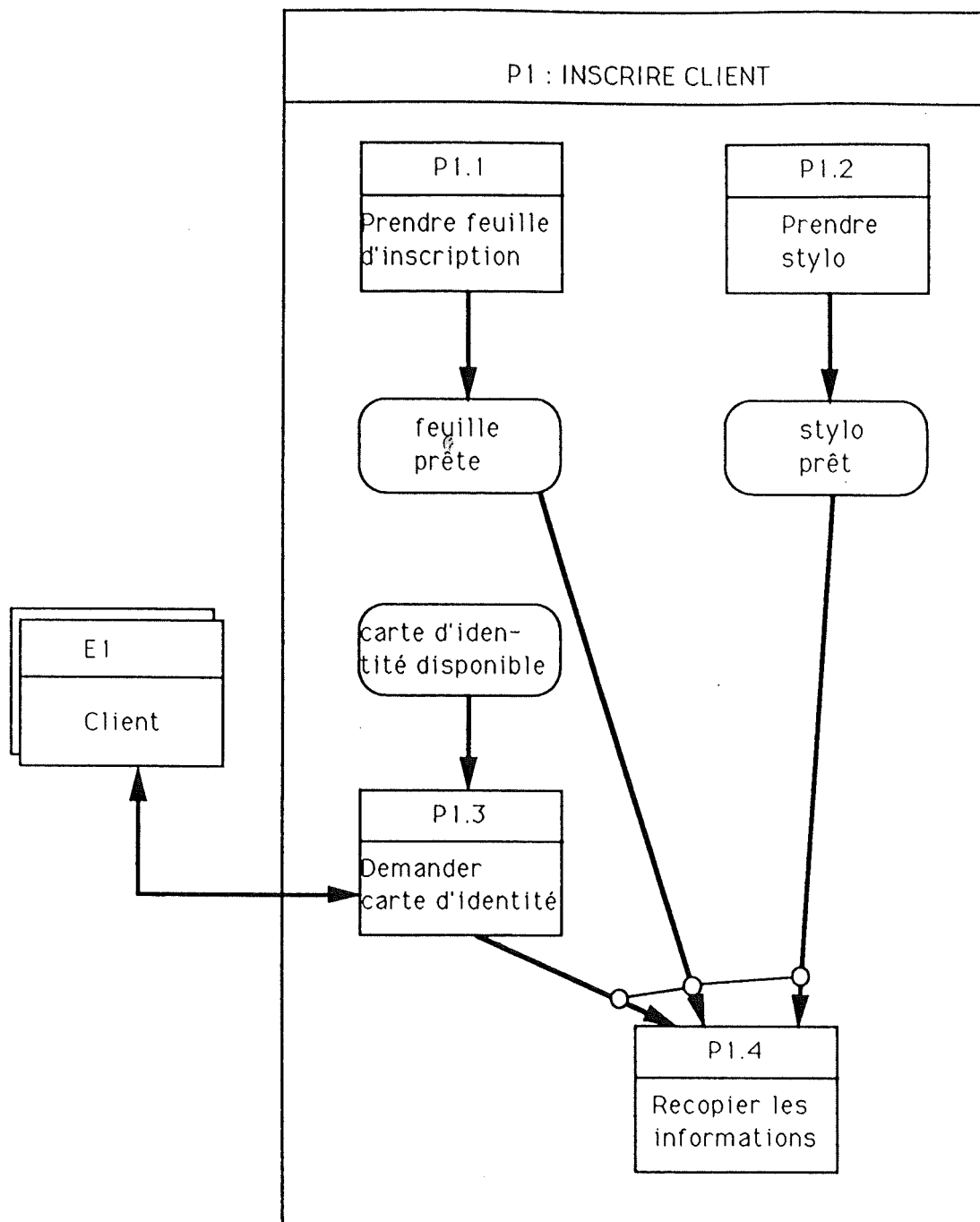


Figure A1.5. : illustration d'un diagramme de transition

5. Remarques

Pour conclure cette présentation de la méthode E.P.A.S., nous abordons encore deux points. Le premier est le fait que la méthode fournit des possibilités en matière de gestion de projet. Le second concerne les structures de contrôle et de communication que la méthode met en oeuvre.

5.1. Gestion de projet

Les méthodologies de conception de système servent généralement de cadre de référence pour les responsables de projet lorsqu'ils élaborent ou adaptent la démarche de développement de systèmes. C'est le cas de la méthode E.P.A.S., qui peut être utilisée pour les activités de gestion de projet (planification, programmation, suivi et contrôle d'un projet). Par exemple, un projet peut être modélisé par des plans de système; la gestion se fait au travers de scénarios de développement, représentés sous la forme de diagrammes de transition.

5.2. Structures de contrôle et de communication

Tout système peut être considéré comme un organisme au sein duquel des organes (sous-systèmes) coopèrent, chaque organe étant lui-même composé de sous-ensembles organisés qui coopèrent. L'harmonisation du comportement global d'un système dépend de la communication qui s'établit entre ses sous-systèmes et de la coordination de leur mise en oeuvre. La structure de contrôle d'un système doit permettre de rendre compte des caractéristiques de coordination des activités d'un scénario de comportement en tenant compte des stratégies de contrôle et de communication retenues.

(Repris de [DEL-JOU], p. 161 à 170)

Annexe 2 : Règlement du jeu de rôle FRIK

FRIK*

© 1987, Tous droits réservés GESTION GROUP-OR



LE MONDE DES AFFAIRES

de 2 à 6 joueurs

LE JEU

Dans quelques instants, vous serez riches de plusieurs millions de dollars que vous aurez à investir dans divers placements afin de les faire fructifier. Pour gagner, vous devrez prendre des risques calculés, négocier avec les autres et faire des échanges. Plus vous investirez, plus vous négocieriez et plus vous échangerez, plus vous aurez de chances de gagner. Dès le début de la partie, foncez et achetez! Les biens ainsi acquis vous serviront de base pour élaborer des stratégies futures. Comme dans la vie, tout est à vendre, tout est négociable. Laissez-vous prendre au jeu.

ENJEU

- Faire fructifier de l'argent en faisant des investissements sûrs ou à risques modérés, ou en spéculant.
- Le gagnant? Celui ou celle qui réussit à acheter les actifs des autres joueurs.

Il faut d'abord opter pour une des deux versions.

Courte durée - On paie immédiatement ses dettes. Emprunts à la Banque FRIK* ou aux autres joueurs interdits.

Longue durée - On peut emprunter de l'argent à la Banque FRIK* ou aux autres joueurs (voir VERSION LONGUE DURÉE).

ACCESSOIRES

- Planche de jeu
- 2 des
- 6 pions
- 3 lingots d'or
- 15 cartes HASARD
- Titres de propriété: 3 BANQUES, 4 SOCIÉTÉS, 5 INDUSTRIES, 6 COMMERCE, 7 IMMEUBLES
- 6 POLICES D'ASSURANCE
- 5 TABLEAUX
- 20 cartes d'ACTIONS de compagnies
- Billets de Banque FRIK*

PRÉPARATIFS

Placez sur la planche:

- les cartes HASARDS, LINGOTS D'OR, TABLEAUX, ACTIONS dans leur rectangle respectif
 - Cartes HASARDS empilées de façon à cacher le texte
 - Cartes TABLEAUX et ACTIONS de façon à toujours voir les oeuvres et les compagnies
 - 3 billets de 1 000 000 \$ dans le rectangle FONDS BOURSIER
 - les pions sur la case DÉPART
- Désignez le Grand Argentier qui:**
- distribue l'argent, les titres de propriété, vend les POLICES D'ASSURANCE, administre la Banque FRIK*, tranche les conflits, etc

PORTFEUILLE DES JOUEURS

	NOMBRE DE JOUEURS				
	Deux	Trois	Quatre	Cinq	Six
NOMBRE DE BILLETS					
1 000 000 \$	8	5	3	3	2
500 000 \$	6	5	5	2	3
100 000 \$	9	4	4	6	3
50 000 \$	8	8	8	8	8
20 000 \$	6	6	6	6	6
10 000 \$	5	5	5	5	5
5 000 \$	6	6	6	6	6

Nota - Cachez votre argent si vous le désirez

AU JEU!

- Déterminez la personne qui commence. Puis procédez dans le sens des aiguilles d'une montre.
- Le Grand Argentier offre à chaque joueur une police d'assurance. Achat facultatif qu'on fait au début, ou plus tard durant la partie.

POLICE D'ASSURANCE

- La police aide à payer les frais encourus à la suite d'une catastrophe.
- La police a trait aux titres de propriété et non aux autres valeurs.
- Vous êtes victime d'une catastrophe et n'êtes pas assuré? Payez les frais. Vous pouvez ensuite acheter une police.
- Achat - Le Grand Argentier prend les 6 polices (quel que soit le nombre de joueurs) les mélange et fait piger au hasard ceux qui en désirent une, en commençant par le joueur devant débiter la partie.
- Police avec franchise - Lors d'une catastrophe vous payez seulement la franchise.

- Police sans franchise - Lors d'une catastrophe, vous n'avez rien à déboursier.
- Vous ne pouvez posséder plus d'une police.
- La police est en vigueur la partie durant, à moins qu'un HASARD ne vous oblige à en payer le renouvellement.
- Une police s'échange contre d'autres valeurs.
- Paiement relatif aux titres en copropriété - À la suite d'une catastrophe, tous les copropriétaires doivent assumer les frais en fonction du pourcentage détenu et de la couverture d'assurance (s'il y a lieu).

Exemple - Trois joueurs possèdent un titre (33 1/3 %) d'une BANQUE. Un des joueurs pige le HASARD qui porte sur le saccage dans une BANQUE.

Joueur A	Joueur B	Joueur C
Police : sans franchise	sans franchise	sans franchise
Frais : aucuns	aucuns	aucuns
Police : sans franchise	avec franchise	sans franchise
Frais : aucuns	33 1/3 % de la franchise	aucuns
Police : sans franchise	avec franchise	non assuré
Frais : aucuns	33 1/3 % de la franchise	33 1/3 % du montant des dommages

RÈGLEMENT

Vous allez investir dans trois types de placements...

1. PLACEMENTS SÛRS

- Séries de titres de propriété variés (SOCIÉTÉS, INDUSTRIES, COMMERCE, IMMEUBLES)
- Séries de titres de BANQUES

2. PLACEMENTS À RISQUES MODÉRÉS

- ACTIONS de compagnies
- TABLEAUX
- PRÊT

3. SPÉCULATION

- LINGOTS D'OR

1 PLACEMENTS SÛRS

Pour vous assurer des revenus réguliers, vous voudrez acquérir des séries de titres de propriété variés et de banques. Vous aurez ainsi une plus grande marge de manoeuvre pour faire d'autres placements (TABLEAUX, ACTIONS, PRÊTS et OR)

■ Séries de titres de propriété variés (SOCIÉTÉS, INDUSTRIES, COMMERCE, IMMEUBLES)

Série de base qui double le revenu de chaque titre :

1 SOCIÉTÉ, 1 INDUSTRIE, 1 COMMERCE, 1 IMMEUBLE

Pour quadrupler le revenu de chaque titre, achetez un deuxième COMMERCE et un deuxième IMMEUBLE pour les intégrer dans votre série de base.

Pour sextupler le revenu de chaque titre, achetez un troisième COMMERCE et un troisième IMMEUBLE pour les intégrer dans votre série qui quadruplant

- Évitez d'acheter n'importe quel titre, à n'importe quel moment. Planifiez vos acquisitions : formez une ou des séries.

Nota - Quoi faire avec une seconde SOCIÉTÉ et ou INDUSTRIE? Formez une autre série ou échangez-les contre d'autres titres.

- Un titre qui ne fait pas partie d'une série rapporte un revenu ordinaire (indiqué sur le titre).
- Comment encaisser?

Exemple 1 - Si un pion s'arrête sur votre COMMERCE (propriété à 100 %), dans une série qui double, réclamez le double du revenu ordinaire.

Exemple 2 - Vous possédez les deux parts (70 % et 30 %) du même COMMERCE dans une série qui double? Réclamez le total des revenus des titres, multiplié par deux.

- Une série est complète quand vous possédez CHACUNE des propriétés à 100 %. Certains titres ont deux parts. faites des offres d'achat ou d'échange aux joueurs qui détiennent ce qui vous manque.

- Séparez la série complétée d'avec les titres hors série.

- Faites des substitutions pour rentabiliser davantage une série. Quand? Avant qu'un joueur lance les dés, à n'importe quel moment durant la partie. Au vu et au su des joueurs.

Exemple - Vous venez d'acheter un IMMEUBLE qui rapporte plus que l'ancien IMMEUBLE. Intégrez le nouvel IMMEUBLE à votre série, et retirez l'ancien.

- Vous pouvez aussi modifier des séries.

Exemple - Vous possédez 2 SOCIÉTÉS, 2 INDUSTRIES, 3 COMMERCE et 3 IMMEUBLES.

Le résultat peut s'exprimer comme suit

Série qui sextuple	Hors série
1 SOCIÉTÉ	1 SOCIÉTÉ
1 INDUSTRIE	1 INDUSTRIE
3 COMMERCE	
3 IMMEUBLES	

Ou comme ceci:

Série qui quadruple	Série qui double
1 SOCIÉTÉ	1 SOCIÉTÉ
1 INDUSTRIE	1 INDUSTRIE
2 COMMERCE	1 COMMERCE
2 IMMEUBLES	1 IMMEUBLE

ACHAT DES TITRES DE PROPRIÉTÉ VARIÉ ET DE TITRES DE BANQUES

- Copropriété interdite, sauf en ce qui a trait à titres déjà prévus à cette fin.

Vous achetez?

- Payez le prix indiqué sur le titre.
- Payez aussi les frais de courtage. Dépensez l'argent dans le rectangle COURTAGE. Un pion encaisse le montant dans ce rectangle quand le pion s'arrête sur COURTAGE.

- Le Grand Argentier vous remet le titre.
- Propriété à plus d'un titre - achat d'un titre à la fois seulement, lors d'un même tour.

Vous n'achetez pas?

- Le Grand Argentier offre le titre aux enchères.
- Prix de départ - le prix du titre.
- Le joueur qui fait la meilleure offre devient propriétaire.

- Frais de courtage - 10 % de la mise.
- Propriété composée de plus d'un titre - le + important est mis en vente.

- Le joueur qui s'arrête sur votre propriété paie le montant du REVENU inscrit sur le titre. Si votre propriété comprend deux titres, que vous possédez, il paie le total des revenus des deux titres.

- Votre pion est sur une propriété détenue par deux joueurs? Payez à chacun l'argent dû.
- Votre pion est sur une propriété à deux : dont un invendu? Vous pouvez l'acheter après avoir payé votre dette au premier propriétaire.
- Deux joueurs qui possèdent une même propriété n'ont rien à déboursier lorsque leur pion s'arrête sur cette propriété.

■ TITRES DE BANQUES

- Les trois BANQUES forment un groupe de propriétés distinct des séries de titres de propriété variés. Chaque BANQUE est différente des deux autres et chacune se divise en trois parts de 33 1/3 %.

- Si un pion s'arrête sur un titre de BANQUE qui vous appartient, réclamez le revenu ordinaire du titre. Vous avez deux parts de la même BANQUE? Réclamez le total des deux revenus, multiplié par deux. Trois parts? Le total des trois revenus, multiplié par trois.

- Au départ, exercez votre sens de la décision. Tenter d'acquérir à la fois des titres de BANQUE et de faire des séries de titres de propriété n'est pas prudent. Pourquoi? Vous risquez de manquer d'argent (votre portefeuille n'est pas une banque). Bref, privilégiez une option.

2. PLACEMENTS À RISQUES MODÉRÉS

But : faire des gains importants, en assumant toutefois des risques plus élevés. Certains HASARDS peuvent affecter le rendement de vos placements

RÈGLES GÉNÉRALES (ACTIONS, TABLEAUX, PRÊT)

- Copropriété interdite
- Placez vos cartes à la vue des joueurs
- Plutôt que d'investir, vous pouvez vendre votre privilège ou l'échanger contre un bien. Le prix est fonction de l'offre et de la demande
- Si votre pion s'arrête sur GALERIE D'ART ou BOURSE, achetez un TABLEAU ou des ACTIONS ou revendez un TABLEAU ou des ACTIONS que vous avez en main (encore, ne faites aucune transaction)
- (Vous voulez revendre immédiatement à la Banque FRIK* ou à la BOURSE? Non. À un joueur? Oui.)

■ PRÊT

Somme d'argent que vous avancez à une personne fictive (jamais aux autres joueurs)

- Si votre pion s'arrête sur PRÊT, vous pouvez prêter jusqu'à 1 000 000 \$ Intérêts : 50 %, payés par la Banque FRIK*
- La somme déposée reste dans le rectangle pour la durée du PRÊT

COMMENT PRÊTER

- Déposez les billets (1 000 000 \$ ou moins) dans un rectangle PRÊT

COMMENT ENCAISSER

- Pour encaisser, votre pion doit effectuer un tour complet et s'arrêter là où vous avez fait le PRÊT (vous pouvez alors prêter à nouveau) ou dépasser la case
- Après un tour de jeu effectué par votre pion, le PRÊT est terminé. Vous reprenez la somme prêtée et réclamez les intérêts à la Banque FRIK*. Il est interdit de laisser une somme dans le rectangle plus d'un tour et d'en retirer des intérêts à chaque tour durant toute la partie
- Vous pouvez prêter chaque fois que vous tombez sur PRÊT. Un prêt, un rectangle (jamais deux prêts dans un même rectangle)
- Vous avez oublié de réclamer votre argent? Attendez que votre pion ait complété un second tour

■ TABLEAUX

ACHAT

- Si votre pion s'arrête sur GALERIE D'ART, vous pouvez acheter un tableau (un seul à la fois, celui de votre choix)
- Lequel désirez-vous? Déposez l'argent à la Banque FRIK*. Mettez le TABLEAU à la vue des joueurs

VENTE

- Vous pouvez vendre un ou plusieurs TABLEAUX à la fois. Il suffit que votre pion s'arrête de nouveau sur GALERIE D'ART (Rappel : vente et échange entre joueurs permis en tout temps)
- Un TABLEAU revendu est retiré de la circulation, il ne peut plus être racheté pour le reste de la partie
- La Banque FRIK* verse le montant de la transaction

■ ACTIONS

- Lorsque votre pion s'arrête sur BOURSE, vous pouvez acheter des ACTIONS de votre choix, de la même entreprise ou non
- Chaque carte d'ACTIONS vous donne droit à 5 % du FONDS BOURSIER
- Prix d'une carte : 50 000 \$
- Achat limité : 80 % de toutes les actions
- Personne ne peut détenir 100 % des actions
- Vous achetez? Combien d'actions voulez-vous? Déposez l'argent dans le FONDS BOURSIER, et partez à la Banque. Placez vos actions à la vue des joueurs
- Pour vendre vos actions à la BOURSE, votre pion doit s'arrêter à nouveau sur BOURSE
- Exemple – Vous avez 60 % des actions? Vous retirez 60 % du FONDS BOURSIER (Rappel : vente et échange entre joueurs permis en tout temps)
- Chaque fois qu'un pion franchit le DÉPART, le Grand Argentier dépose 100 000 \$ de la Banque FRIK* dans le FONDS BOURSIER. C'est la valeur de vos actions qui augmente...

3. SPÉCULATION

■ LINGOTS D'OR

- Pour spéculer sur la valeur de l'or, votre pion doit s'arrêter sur LINGOTS D'OR
- On n'achète qu'un seul lingot à la fois. L'argent va à la Banque FRIK*
- On peut acheter de l'or et le revendre soit lors d'un même tour, soit plus tard durant la partie. Pour revendre, votre pion n'a pas à s'arrêter de nouveau sur la case LINGOTS D'OR
- Vous pouvez revendre un ou plusieurs lingots à la fois. Combien pour un lingot? Lancez les dés et multipliez par 50 000 \$. C'est la Banque FRIK* qui paye. Combien pour 2 ou 3 lingots? Un coup de dés par lingot, multiplié par 50 000 \$, ou, un seul coup de dés, multiplié par 50 000 \$, multiplié par le nombre de lingots. Les lingots revendus retournent dans le rectangle LINGOTS D'OR

- Vous voulez vendre ou échanger votre or contre d'autres valeurs? Le prix est fonction de l'offre et de la demande. Vendre ou échanger votre privilège d'acheter contre d'autres biens? Oui

CARTES HASARD

- Certaines sont négociables contre d'autres valeurs
- Pourquoi ne pas lire les cartes avant la partie pour voir venir les coûts?
- HASARD a toujours priorité : répondez à l'obligation, faites ensuite vos autres transactions

NÉGOCIATIONS

Vous voulez faire des propositions d'achat ou d'échange? Attendez votre tour. Lorsque la négociation est terminée, un autre joueur peut vous relancer

PRINCIPE DE L'OFFRE ET DE LA DEMANDE

Malgré le prix payé, la valeur marchande d'un bien pourra varier, selon que vous serez en position d'offrir ou faible

DOUBLET

Jouez autant de fois que vous avez un doublet

À COURT D'ARGENT?

1. MANQUE D'ARGENT

Il y a plusieurs façons de rembourser votre créancier. Les règles suivantes demeurent toutefois obligatoires.

- Le paiement d'une dette ne peut être étalé ou reporté dans le temps. Payez immédiatement (version courte durée).
- Remboursez le montant exact. Aucune réduction n'est permise.
- Le créancier peut exiger de l'argent comptant, mais les moyens à prendre pour réunir la somme ne sont pas de son ressort.

EXEMPLES DE PAIEMENT

A Vous devez 50 000 \$. Offrez au créancier un ou des titres dont le prix payé équivaut à la somme due. Ou offrez ACTIONS, TABLEUX et autres valeurs.

B Si le créancier refuse (exige du comptant), offrez vos valeurs aux autres joueurs. Cet argent servira à payer la dette. Par contre, les joueurs peuvent, eux aussi, refuser d'acheter.

C Dernier recours : vendez des titres de propriété à la Banque FRIK* qui vous donnera la moitié du prix payé. Ces titres seront aussitôt revendus aux enchères.

- Lorsque vous manquez d'argent, vous pouvez vendre aux joueurs ACTIONS et TABLEUX au prix de l'offre et de la demande. Il est interdit de revendre directement vos ACTIONS à la BOURSE et vos TABLEUX à la GALERIE D'ART sans suivre la procédure normale. Vous pouvez céder un PRÊT, mais l'argent demeure dans le rectangle. Avis au créancier voir FAILLITE, dernier paragraphe.

2. FAILLITE

Lorsque vous n'êtes plus en mesure de satisfaire à vos obligations :

- Remettez tout votre avoir au créancier (argent, titres de propriété, tableaux, etc.) ACTIONS et TABLEUX doivent rester en circulation. Un créancier ne peut donc pas les refuser.
- Ou vendez vos titres à moitié prix à la Banque FRIK* et remettez l'argent, ainsi que vos ACTIONS et TABLEUX, au créancier. La Banque vendra ces titres aux enchères. Si votre créancier possède déjà 70 % des ACTIONS de la BOURSE, il ne peut accepter plus de 10 % d'ACTIONS (Rappel : on ne peut détenir plus de 80 % des ACTIONS). S'il n'y a pas d'acheteur, cédez alors les actions qui restent à la BOURSE. S'il y a mésestante quant à la valeur des ACTIONS et des TABLEUX, le prix d'achat l'emporte sur le prix potentiel.
- S'il y a plus d'un créancier (dans le cas d'une copropriété) et qu'il y ait mésestante quant au partage, vendez à moitié prix vos titres à la Banque FRIK* et remettez à chacun la somme équivalant au pourcentage détenu par chaque créancier.
- Exemple : Si les copropriétaires d'un titre vous ont mis en faillite, remettez-leur l'argent qu'il vous reste selon le pourcentage que chacun détient dans ce titre. S'il y a mésestante quant au partage des ACTIONS et des TABLEUX, le Grand Argentier les fait tirer au sort, séparément.
- Pour ce qui est d'un lingot d'or, vous avez le choix : donnez-le au créancier, ou lancez les dés (pour fixer le prix), encaissez et remettez l'argent au créancier.
- En ce qui concerne un PRÊT, laissez l'argent dans le rectangle. Cédez l'argent au créancier qui profitera des intérêts. Si vous avez prêté 1 000 000 \$ et si vous devez 800 000 \$, le créancier vous remet la différence. Pour récupérer l'argent du PRÊT, c'est maintenant le créancier qui doit s'arrêter sur la case PRÊT ou la dépasser (peu importe où se trouve son pion).

3. VENTE AUX ENCHÈRES

- Le Grand Argentier préside la criée.
- Montant de départ : la moitié du prix d'achat (Rappel : ne pas confondre avec ACHAT LE TITRE DE PROPRIÉTÉ, troisième para.).
- Les offres se font à tour de rôle.
- On revend les propriétés d'abord en bloc (les titres). Si personne n'est preneur, on lève à l'unité. Dans les deux cas, on doit payer 10 % de COURTAGE.
- Les titres offerts aux enchères et non revendus restent en vente en tout temps. Le prix de vente demeure fixé à la moitié du prix réel. Un joueur attend son tour avant de faire une offre ; il n'est pas nécessaire de s'arrêter sur le titre coté.
- Le Grand Argentier sépare les titres non revendus de ceux qui le sont.

VERSION LONGUE DURÉE

- Oui, vous pouvez emprunter à la Banque FRIK*.
- L'emprunteur offre en garantie un ou des titres de propriété (jamais d'autres valeurs).
- Chaque titre permet d'emprunter jusqu'à 50 % de son prix d'achat.
- Une propriété en garantie continue de rapporter un revenu.
- Temps alloué pour rembourser : 3 tours depuis la case où se trouve le pion au moment de l'emprunt.
- Le Grand Argentier note le prénom de l'emprunteur, la case où il est, la propriété en garantie, le montant de l'emprunt et le nombre de tours complétés.

- Chaque emprunt comporte un intérêt de 10 %.
- Emprunt remboursable en totalité (en un seul paiement) lorsque l'emprunteur a complété 3 tours.
- Un joueur ne peut rembourser? La Banque FRIK* saisit le titre en garantie et le revend aux enchères.
- Les prêts d'argent entre joueurs sont permis à eux d'en établir les modalités.
- On utilise les cases PRÊT de la même manière que dans la version courte durée. Puisqu'il s'agit d'un PRÊT à une personne fictive, cet élément du jeu est indépendant des emprunts faits à la Banque FRIK* et aux autres joueurs.

À vous de jouer

Pour tout renseignement concernant le jeu et la société FRIK*, écrivez à :

Productions FRIK International, Inc.
Case postale 7125
Charlesbourg (Québec)
Canada
G1G 5E1

623 7882

* Marque de commerce de GESTION GÉOCLIP CANADA

Annexe 3 :

Spécification du jeu de rôle FRIK

1° DESCRIPTION DE L'IDEE ORIGINALE OU D'UN SYNOPSIS POUR LE SCENARIO.

Le jeu automatisé se déroulera de la façon suivante :

- Le jeu est ouvert
- Les différents joueurs recevront leur argent en début de partie
- Les différents joueurs se verront proposer une PA
- Tant qu'il restera plus d'un joueur, la séquence suivante sera exécutée :
 - Un joueur lance les dés et avance du nombre de cases voulu en déclenchant éventuellement certains comportements de cases en passant par-dessus (case de prêt et case départ).
 - Une série de comportements due au type de la case sur laquelle on s'est arrêté est déclenchée.
 - Les joueurs tombés en faillite sont éliminés
 - Les joueurs peuvent négocier entre-eux
- Le jeu est fermé

2° IDENTIFICATION DES ACTEURS ET DESCRIPTION DE LEURS PRINCIPALES CARACTERISTIQUES.

1) Relevé des différents acteurs.

Les acteurs présents dans le jeu FRIK sont:

- les joueurs
- le Grand Argentier
- les cases de Hasard
- la case de Courtage
- les cases de Prêt
- les cases de Galerie d'Art
- les cases de Bourses
- la case de Lingot d'Or
- les cases représentant des titres de propriétés, à savoir les cases de Société, d'Industrie, de Commerce, d'Immeuble ou de Banque

2) Identification des comportements des acteurs.

A. Les joueurs.

Le joueur est un acteur mobile qui se déplace sur les cases du plateau.

Il peut: -acheter une police d'assurance, s'il le désire

-acheter un titre de propriété. Pour ce faire, il doit payer le prix et le courtage du titre. Si plusieurs titres sont disponibles, le plus important lui est proposé.

-encaisser la somme d'argent contenue dans la case courtage

-percevoir un revenu de titre de propriété lorsqu'un joueur s'arrête sur une propriété qui lui appartient

-acheter un titre de banque (payer le prix et le courtage du titre)

-percevoir le revenu du titre de banque lorsqu'un joueur s'arrête sur la case de sa banque

-acheter des tableaux (un à la fois, choisir, puis payer)

-prêter de l'argent lorsqu'il tombe sur la case de Prêt

-acheter des actions à concurrence de 80 % des actions totales disponibles (dans ce cas, la somme due va dans la case du Fond Boursier)

-revendre un ou des tableaux au Grand Argentier lorsqu'il tombe à nouveau sur la case Galerie d'Art

-réclamer un prêt effectué à une personne fictive et ceci juste après le coup qui a permis de s'arrêter ou de passer à nouveau au dessus de la case Prêt pour laquelle ce prêt a été accordé

-revendre des actions au Grand Argentier lorsqu'on tombe sur une case Bourse

- acheter des lingots d'or (un seul à la fois) lorsqu'on tombe sur la case Lingot d'Or
- revendre un ou des lingots d'or au Grand Argentier à n'importe quel coup du jeu. La vente se fait au prix de 50000\$ multiplié par les points obtenus lors d'un lancé de dés
- faire une offre d'achat sur des biens et/ou privilèges d'un autre joueur
- faire une offre de vente sur des biens et/ou privilèges à un autre joueur
- faire une offre d'échange de biens et/ou de privilèges avec des biens et/ou des privilèges d'un autre joueur
- exiger de l'argent comptant lorsqu'un autre joueur lui doit un revenu pour un passage sur une de ses propriétés
- faire faillite lorsqu'on ne peut plus payer une dette et qu'aucun prêt ne peut plus être accordé
- faire une offre d'achat dans le cas d'une vente aux enchères
- emprunter de l'argent au Grand Argentier. Il doit déposer un ou des titres de propriété en garantie (titres d'une somme au moins égale à la moitié du montant emprunté).
- rembourser un emprunt au Grand Argentier. Le remboursement se fait impérativement 3 tours après le prêt, sinon le Grand Argentier saisit les titres en garantie et les vend aux enchères
- emprunter de l'argent à un autre joueur
- rembourser un emprunt contracté auprès d'un autre joueur
- placer un titre de propriété dans une série
- retirer un titre de propriété d'une série

B. Le Grand Argentier.

Le Grand Argentier est un acteur invisible. C'est lui qui gère la banque FRIK. Son rôle est de :

- proposer une police d'assurance à chaque joueur au début de la partie
- proposer les titres de propriétés invendus lorsqu'un joueur tombe sur une propriété non encore totalement vendue
- proposer des titres de propriétés aux enchères lorsque le joueur qui a accédé à la case correspondante ne désire pas les acheter
- payer aux joueurs les intérêts des prêts qu'ils ont consentis à des personnes fictives
- proposer des actions aux joueurs s'arrêtant sur une des cases Bourse
- proposer des tableaux (un à la fois) aux joueurs s'arrêtant sur une des cases Galerie d'Art
- déposer 100000\$ dans le Fond Boursier à chaque passage d'un joueur par la case départ
- proposer des lingots d'or (un à la fois) aux joueurs qui tombent sur la case Lingots d'Or
- présider les ventes aux enchères
- prêter de l'argent aux joueurs (en exigeant des titres de propriétés pour au moins le double du montant du prêt en garantie)
- exiger le remboursement d'un prêt auprès d'un joueur après 3 tours calculés depuis la case où se trouvait celui-ci lors de l'emprunt

C. Case de Hasard.

Les cases de hasard sont des acteurs visibles et fixes. Leur rôle est de donner une carte de hasard au joueur qui est tombé sur une de ces cases. S'il s'agit d'une carte catastrophe, le joueur doit payer la somme indiquée, à moins qu'il possède une police d'assurance portant sur l'objet sinistré. Dans ce cas, et dans ce cas uniquement, il ne doit payer que la franchise, si elle existe.

S'il s'agit d'une carte de privilège, le joueur peut soit en profiter, soit la négocier avec d'autres joueurs.

D. Case de Courtage.

La case de Courtage est un acteur visible et fixe. Son rôle est de recueillir l'argent de courtage lors de ventes de titres et de le redistribuer aux joueurs qui tombent sur la case.

E. Les cases de Prêt.

Les cases de prêt sont des acteurs visibles et fixes. Leur rôle est de recevoir en prêt des joueurs (un seul à la fois) un maximum de 1000000\$.

Le remboursement du prêt doit être demandé par le joueur lui-même lors d'un coup qui lui permet de passer à nouveau au-dessus ou sur la case. La somme remboursée est augmentée de 50% d'intérêts. Le prêt pouvant être cédé à un autre joueur, la personne pouvant réclamer le montant et les intérêts peut changer.

F. Les cases Galerie d'Art.

Les cases galerie d'art sont des acteurs visibles et fixes. Leur rôle est de proposer au joueur qui tombe sur la case d'acheter un tableau au choix (s'il en reste) au prix mentionné sur la carte et/ou de revendre des tableaux que le joueur possédait déjà auparavant au prix également écrit sur la carte.

G. Les cases Bourses.

Les cases bourses sont des acteurs visibles et fixes. Leur rôle est de proposer aux joueurs qui tombent sur une de ces cases d'acheter des actions, à concurrence de 80% maximum, y inclus les actions déjà achetées auparavant. Elle proposent également aux joueurs de revendre tout ou partie de ses actions acquises auparavant au prix de l'action (à savoir 50000\$)

3° SPECIFICATION DU SCENARIO EN DEFINISSANT LES JALONS ET LA DECOUPE EN PARTIES DU SCENARIO.

1) Définition des jalons.

- J1 : Le jeu n'est pas commencé.
- J2 : Le jeu est commencé et chaque joueur a reçu une somme d'argent.
- J3 : Le MJ a proposé des polices d'assurance aux joueurs.
- J4 : Le joueur courant a lancé des dés.
- J5 : Le joueur est tombé sur une case (et a éventuellement effectué ce qu'il pouvait ou devait faire).
- J6 : Le MJ a éventuellement éliminé un joueur en faillite.
- J7 : Les joueurs ont éventuellement négocié avec les autres joueurs et/ou le MJ.
- J8 : Les joueurs ont éventuellement manipulé leur(s) série(s).
- J9 : Il ne reste qu'un joueur.
- J10 : Le jeu est terminé.

Les jalons J4 à J8 surviennent un nombre quelconque de fois.

2) Découpe en partie du scénario.

- P10 (J1 à J2) : Le MJ ouvre le jeu et distribue l'argent selon le nombre de joueurs.
- P20 (J2 à J3) : Le MJ propose des polices d'assurance aux joueurs.
- P30 (J3 à J4) : Le MJ désigne le joueur courant et celui-ci lance les dés.
- P40 (J4 à J5) : Le joueur avance d'autant de cases que de points obtenus au lancé de dés. La case déclenche les comportements qui lui sont associés.
- P50 (J5 à J6) : Le MJ élimine éventuellement les personnes qui sont en faillite.
- P60 (J6 à J7) : Les joueurs négocient éventuellement avec les autres joueurs.
- P70 (J7 à J8) : Les joueurs manipulent éventuellement leur(s) série(s).
- P80 (J8 à J9) : Le MJ constate qu'il ne reste qu'un joueur.
- P90 (J9 à J10) : Le MJ ferme le jeu.

4° REVISION DE LA DESCRIPTION DES ACTEURS ET DES PARTIES DU SCENARIO ET ETABLISSEMENT DU RESEAU DE TRANSITIONS ENTRE LES JALONS DU SCENARIO.

1) Application de la méthode telle quelle.

A. Acteurs et caractéristiques des acteurs supplémentaires.

* *Le jeu* : acteur fixe et visible. Il est le cadre de l'animation graphique. Il n'a aucun comportement et est caractérisé par son état "ouvert" ou "fermé" et par le nombre de joueurs.

* *Le joueur* : il est caractérisé par 3 états : "en attente" lorsqu'il n'est pas le joueur courant, "courant" lorsqu'il est le joueur courant, "en faillite" lorsqu'il est éliminé pour faillite.

* *Les cartes de police d'assurance* : elles sont des acteurs visibles. Elles sont caractérisées par l'état "vendue" ou "non vendue".

* *Les dés* : ce sont des acteurs fixes et visibles. Il sont caractérisés par 7 états : "en lancement", "1", "2", "3", "4", "5" ou "6".

* *Les séries* : ce sont des acteurs (in)visibles. Elles sont caractérisées par 3 états : "disponibles" si les joueurs ont le droit de les modifier, "en cours" si les joueurs sont en train de les modifier et "complétée" quand plus aucun des joueurs ne désire plus modifier ses séries. A partir de ce moment, aucun joueur ne peut plus toucher à une de ses séries.

B. Jalons supplémentaires.

P10 : J1 : Le jeu n'est pas commencé.

J1.1 : Le nombre de joueur est connu.

J1.2 : Le MJ a distribué l'argent aux joueurs.

J.2 : Le jeu est commencé et chaque joueur a reçu une somme d'argent.

P20 : J2 : Le jeu est commencé et chaque joueur a reçu une somme d'argent.

J2.1 : Le MJ a proposé à un joueur d'acheter une police d'assurance (PA).

J2.2 : Le joueur a répondu à l'offre du MJ d'acheter une PA.

J2.3 : Le joueur a éventuellement payé et reçu sa PA.

J3 : Le MJ a proposé des PA à tous les joueurs.

P30 : J3 : Le MJ a proposé des PA à tous les joueurs.

J3.1 : Le MJ a désigné le joueur courant.

J4 : Le joueur a lancé les dés.

P40 : J4 : Le joueur a lancé les dés.

J4.1 : Le joueur a avancé du nombre de cases désigné par les dés en déclenchant des comportements éventuels de cases en passant par dessus.

J5 : Le joueur est tombé sur une case et a éventuellement effectué ce qu'il pouvait ou devait faire.

P60 : J6 : Le MJ a éventuellement éliminé un joueur en faillite.

J6.1 : Un joueur a éventuellement fait une offre de vente, d'achat ou d'échange à un autre joueur.

J6.2 : L'autre joueur a soit accepté, soit refusé l'offre ou le joueur n'a pas fait d'offre.

J6.3 : Plus aucun joueur n'a fait de proposition de vente, d'achat ou d'échange.

J7 : Les joueurs ont éventuellement négocié.

P70 : J7 : Les joueurs ont éventuellement négocié.

J7.1 : Un joueur a éventuellement retiré une ou des cartes d'une ou de plusieurs de ses séries.

J7.2 : Un joueur a éventuellement placé un ou des titres de propriété dans une série.

J7.3 : Plus aucun joueur n'a désiré faire de manipulations

J8 : Les joueurs ont éventuellement manipulé leurs séries.

C. Découpe en séquence.

- S10 (J1 à J1.1) : Le MJ demande le nombre de joueurs.
S20 (J1.1 à J1.2) : Le MJ distribue l'argent aux joueurs.
S30 (J1.2 à J2) : Le MJ ouvre le jeu.
S40 (J2 à J2.1) : Le MJ propose à un joueur d'acheter une PA.
S50 (J2.1 à J2.2) : Le joueur répond à l'offre du MJ d'acheter une PA.
S60 (J2.2 à J2.3) : Eventuellement, le joueur paye et reçoit sa PA.
S70 (J2.3 à J2) : Si le MJ n'a pas encore proposé une PA à tout le monde, il la propose au joueur suivant.
S80 (J2.3 à J3) : Cas contraire à la séquence 70.
S90 (J3 à J3.1) : Le MJ désigne le joueur courant.
S100 (J3.1 à J4) : Le joueur courant lance les dés.
S110 (J4 à J4.1) : Le joueur avance d'autant de cases que le nombre obtenu au lancé de dés et déclenche le comportement éventuel d'une case en passant par dessus.
S120 (J4.1 à J4.2) : La case déclenche les comportements et le joueur exécute éventuellement ce qu'il peut ou doit faire.
S140 (J5 à J6) : Le MJ élimine éventuellement les joueurs en faillite.
S150 (J6 à J6.1) : Eventuellement, un joueur fait une offre de vente, d'achat ou d'échange d'un bien ou privilège à un autre joueur.
S160 (J6.1 à J6.2) : L'autre joueur refuse ou accepte l'offre.
S162 (J6.2 à J6) : Un joueur fait à nouveau une offre.
S163 (J6.2 à J6.3) : Plus aucun joueur ne fait d'offre.
S164 (J6.3 à J7) : Fin des négociations.
S170 (J6 à J6.2) : Pas d'offre d'achat, de vente ou d'échange.
S180 (J7 à J7.1) : Un joueur retire un ou des titres d'une ou de plusieurs de ses séries.
S185 (J7.1 à J7) : Un joueur désire manipuler une de ses séries.
S190 (J7.1 à J7.2) : Un joueur place un ou des titres dans une série.
S200 (J7.1 à J7.3) : Plus aucun joueur ne désire faire de manipulation.
S210 (J7.2 à J7.3) : Plus aucun joueur ne désire faire de manipulation.
S220 (J7 à J7.2) : Un joueur place un ou des titres dans une série (sans en retirer).
S225 (J7.2 à J7) : Un joueur désire manipuler une de ses séries.
S228 (J7.3 à J8) : Fin des manipulations de séries.
S230 (J8 à J9) : Le MJ constate qu'il ne reste qu'un joueur.
S240 (J9 à J10) : Le MJ ferme le jeu.

D. Acteurs et caractéristiques des acteurs supplémentaires.

- * *Les cases de hasard* : elles doivent gérer l'ensemble des cartes de hasard.
- * *Les cartes de hasard* : ce sont des acteurs visibles. Elles sont caractérisées par l'état "catastrophe" ou "privilège".
- * *La case de courtage* : elle est caractérisée par le montant qu'elle contient.
- * *Les cases de prêt* : elles sont caractérisées par les identifiants des joueurs qui ont prêté (ou à qui le prêt a été cédé) et par l'état "autorisation de réclamer" si le joueur vient juste de s'arrêter ou de passer au-dessus de la case de prêt et "interdiction de réclamer" dans le cas contraire.
- * *Les cases galerie d'art* : elles doivent gérer l'ensemble des cartes tableaux invendues.
- * *Les cartes de tableaux* : ce sont des acteurs visibles caractérisés par l'état "invendue", "vendue" ou "retirée du jeu".
- * *Les cases de bourses* : elles doivent gérer l'ensemble des actions.
- * *Les cartes d'actions* : ce sont des acteurs visibles. Elles sont caractérisées par l'état "vendue" ou "invendue".
- * *La case fonds boursier* : c'est un acteur visible et fixe. Elle est caractérisée par le montant qu'elle contient.
- * *La case lingot d'or* : elle doit gérer l'ensemble des lingots d'or.
- * *Les lingots d'or* : ce sont des acteurs visibles. Ils sont caractérisés par l'état "vendu" ou "invendu".
- * *Les cases de titres de propriétés* : elles gèrent l'ensemble des cartes de propriété se rapportant à leur case. Chaque case est caractérisée par l'état "libre" si aucune partie de la

propriété n'est déjà vendue, "semi-libre" si une partie (non entière) de la propriété est libre et "occupée" lorsque l'entière des titres de la propriété est vendue.

* *Les cartes de titres de propriétés* : ce sont des acteurs visibles caractérisés par un pourcentage du titre de propriété qu'ils représentent et par l'état "vendu" ou "invendu" ainsi que par l'appartenance à une série qui double, quadruple ou sextuple le revenu du titre.

* *Les cartes de banques* : ce sont des acteurs visibles caractérisés par l'état "vendu" ou "invendu" ainsi que par l'appartenance de plusieurs titres de la même banque à un même joueur (ce qui permet de doubler ou de tripler la somme des revenus des titres).

E. Jalons supplémentaires (suite).

- S110 : J4 : Le joueur a lancé les dés.
J4.0.1 : Le joueur a avancé d'une case.
J4.0.2 : Si la case est une case de prêt et que le joueur a prêté de l'argent sur cette case lors d'un tour précédent, la case est passée à l'état "autorisation de réclamer" pour ce joueur.
J4.0.3 : Si la case est la case de départ, le MJ a déposé 100000\$ dans le fonds boursier et le joueur a reçu 5% du fonds boursier par carte action qu'il possède.
J4.1 : Le joueur a avancé du nombre de cases désigné par les dés en déclenchant des comportements éventuels de cases en passant par-dessus.
- S120 : J4.1 : Le joueur a avancé du nombre de cases désigné par les dés en déclenchant des comportements éventuels de cases en passant par-dessus.
J4.1.1 : S'il s'agit d'une case de hasard, le MJ a donné une carte.
J4.1.2 : Si la carte de hasard est une carte "catastrophe", le joueur a payé la somme.
J4.1.3 : Si la carte de hasard est une carte "privilege", le joueur en a éventuellement profité.
J4.1.4 : Si la case est la case de courtage, le joueur a reçu la somme contenue dans cette case.
J4.1.5 : Si la case est une case de prêt, le joueur a éventuellement réclaté son argent et ses intérêts.
J4.1.6 : Si la case est une case de prêt, le joueur a éventuellement prêté de l'argent.
J4.1.7 : Si la case est une case galerie d'art, le joueur a éventuellement vendu un ou des tableaux.
J4.1.8 : Si la case est une case galerie d'art, le joueur a éventuellement acheté un tableau.
J4.1.9 : Si la case est une case bourse, le joueur a éventuellement vendu des actions.
J4.1.10 : Si la case est une case bourse, le joueur a éventuellement acheté des actions.
J4.1.11 : Si la case est une case lingot d'or, le joueur a éventuellement acheté un lingot d'or.
J4.1.12 : Si la case est une case avec titres de propriété dont l'état est "semi-libre", le joueur a éventuellement payé le revenu du titre à un autre joueur (si aucune partie de la propriété ne lui appartient).
J4.1.13 : Si la case est une case avec titres de propriété dont l'état est "occupée", le joueur a éventuellement payé le revenu du titre au(x) joueur(s).
J4.1.14 : Si la case est une case avec titres de propriété dont l'état est "semi-libre" ou "libre", le MJ a proposé au joueur d'en acheter une partie.
J4.1.15 : Si la case est une case avec titres de propriété dont l'état est "semi-libre" ou "libre", le joueur a éventuellement acheté le titre.
J4.1.16 : Si la case est une case avec titres de propriété dont l'état est "semi-libre" ou "libre", le MJ propose éventuellement le titre aux enchères.
J4.1.17 : Le titre de propriété a été éventuellement vendu aux enchères.
J5 : Le joueur est tombé sur une case et a éventuellement effectué ce qu'il pouvait ou devait faire.

- S150 : J6 : Le MJ a éventuellement éliminé un joueur en faillite.
 J6.0.1 : Un joueur a éventuellement fait une offre de vente d'un bien ou privilège qu'il détient.
 J6.0.2 : Un joueur a éventuellement fait une offre d'achat d'un bien ou privilège à un autre joueur.
 J6.0.3 : Un joueur a éventuellement fait une offre d'échange d'un bien ou privilège à un autre joueur.
 J6.1 : Un joueur a éventuellement fait une offre de vente, d'achat ou d'échange à un autre joueur.

F. Découpe en tableaux.

- T10 (J1 à J1.1) : Le MJ demande le nombre de joueurs.
 T20 (J1.1 à J1.2) : Le MJ distribue l'argent aux joueurs.
 T30 (J1.2 à J2) : Le MJ ouvre le jeu.
 T40 (J2 à J2.1) : Le MJ propose à un joueur d'acheter une police d'assurance (PA).
 T50 (J2.1 à J2.2) : Le joueur répond à l'offre du MJ d'acheter une PA.
 T60 (J2.2 à J2.3) : Eventuellement, le joueur paye et reçoit sa PA.
 T70 (J2.3 à J2) : Si le MJ n'a pas encore proposé à tout le monde une PA, il la propose au joueur suivant.
 T80 (J2.3 à J3) : Fin de proposition de PA.
 T90 (J3 à J3.1) : Le MJ désigne le joueur courant.
 T100 (J3.1 à J4) : Le joueur courant lance les dés.
 T110 (J4 à J4.0.1) : Le joueur avance d'une case.
 T115 (J4.0.1 à J4.0.1) : Si la case n'est pas une case de prêt ou de départ et que le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
 T120 (J4.0.1 à J4.0.2) : Si la case est une case de prêt et que le joueur a prêté sur cette case, l'état de la case passe à "autorisation de réclamer" pour ce joueur.
 T130 (J4.0.1 à J4.0.3) : Si la case est la case de départ, le MJ dépose 100000\$ dans le fonds boursier et le joueur reçoit 5% du fonds boursier par carte action qu'il possède.
 T140 (J4.0.2 à J4.0.1) : Si le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
 T150 (J4.0.3 à J4.0.1) : Si le joueur n'a pas encore avancé du nombre de cases voulu, il avance de nouveau d'une case.
 T155 (J4.0.1 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
 T160 (J4.0.2 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
 T170 (J4.0.3 à J4.1) : Terminaison du déclenchement des comportements de cases en passant par dessus.
 T180 (J4.1 à J4.1.1) : S'il s'agit d'une case de hasard, donner une carte.
 T190 (J4.1.1 à J4.1.2) : S'il s'agit d'une carte de hasard "catastrophe", le joueur paye la somme, ainsi que les copropriétaires.
 T200 (J4.1.1 à J4.1.3) : S'il s'agit d'une carte de hasard "privilège", le joueur en profite ou la garde.
 T210 (J4.1 à J4.1.4) : S'il s'agit d'une case de courtage, le joueur empoche la somme contenue dans la case.
 T220 (J4.1 à J4.1.5) : S'il s'agit d'une case de prêt et que le joueur a prêté, il réclame éventuellement son argent et ses intérêts.
 T230 (J4.1.5 à J4.1.6) : S'il s'agit d'une case de prêt, le joueur prête éventuellement de l'argent.
 T240 (J4.1 à J4.1.6) : S'il s'agit d'une case de prêt, le joueur prête éventuellement de l'argent (sans avoir réclamé auparavant).
 T250 (J4.1 à J4.1.7) : S'il s'agit d'une case galerie d'art, le joueur vend éventuellement un ou des tableaux.

- T260 (J4.1.7 à J4.1.8) : S'il s'agit d'une case galerie d'art, le joueur achète éventuellement un tableau.
- T270 (J4.1 à J4.1.8) : S'il s'agit d'une case galerie d'art, le joueur achète un tableau (sans en vendre car il n'en a pas).
- T280 (J4.1 à J4.1.9) : S'il s'agit d'une case bourse, le joueur revend éventuellement des actions.
- T290 (J4.1.9 à J4.1.10) : S'il s'agit d'une case bourse, le joueur achète éventuellement des actions.
- T300 (J4.1 à J4.1.10) : S'il s'agit d'une case bourse, le joueur achète des actions (sans en vendre car il n'en a pas).
- T310 (J4.1 à J4.1.11) : S'il s'agit d'une case lingot d'or, le joueur achète éventuellement un lingot d'or.
- T320 (J4.1 à J4.1.12) : S'il s'agit d'une case de propriété avec état = "semi-libre", le joueur paye le revenu du titre à un autre joueur si aucune partie ne lui appartient.
- T330 (J4.1 à J4.1.13) : S'il s'agit d'une case de propriété avec état = "occupée", le joueur paye le revenu du titre à un autre joueur si aucune partie ne lui appartient.
- T335 (J4.1.13 à J5) : Fin si la case est "occupée".
- T340 (J4.1 à J4.1.14) : S'il s'agit d'une case de propriété avec état = "libre", le MJ propose au joueur d'acheter la partie la plus importante.
- T350 (J4.1.12 à J4.1.14) : S'il s'agit d'une case de propriété avec état = "semi-libre", le MJ propose au joueur d'acheter la partie la plus importante qui reste.
- T360 (J4.1.14 à J4.1.15) : S'il s'agit d'une case de propriété avec état = "libre" ou "semi-libre", le joueur achète le titre.
- T365 (J4.1.15 à J5) : Fin de l'achat du titre par le joueur.
- T370 (J4.1.14 à J4.1.16) : Cas contraire à 360, le MJ propose le titre aux enchères.
- T375 (J4.1.16 à J5) : Le titre n'intéresse personne et n'est pas vendu.
- T380 (J4.1.16 à J4.1.17) : Le titre est vendu aux enchères.
- T390 (J4.1.16 à J5) : Le titre n'est pas vendu aux enchères.
- T395 (J4.1.17 à J5) : Fin de la vente aux enchères.
- T400 (J4.1.2 à J5) : Fin du comportement de hasard "catastrophe".
- T410 (J4.1.3 à J5) : Fin du comportement de hasard "privilège".
- T420 (J4.1.4 à J5) : Fin du comportement de courtage.
- T440 (J4.1.6 à J5) : Fin d'une réclamation de prêt éventuelle suivie d'un prêt.
- T460 (J4.1.8 à J5) : Fin d'une vente éventuelle d'un tableau suivie d'un achat.
- T480 (J4.1.10 à J5) : Fin d'un achat éventuel d'actions suivie d'une vente d'actions.
- T490 (J4.1.11 à J5) : Fin d'un achat de lingot d'or.
- T500 (J5 à J6) : Le MJ élimine éventuellement les personnes en faillite.
- T502 (J6 à J6.0.1) : Eventuellement, le joueur fait une offre de vente d'un bien ou privilège qu'il détient.
- T504 (J6 à J6.0.2) : Eventuellement, le joueur fait une offre d'achat d'un bien ou privilège à un autre joueur.
- T506 (J6 à J6.0.3) : Eventuellement, le joueur fait une offre d'échange d'un bien ou privilège.
- T510 (J6.0.1 à J6.1) : Fin d'offre de vente.
- T512 (J6.0.2 à J6.1) : Fin d'offre d'achat.
- T515 (J6.0.3 à J6.1) : Fin d'offre d'échange.
- T520 (J6.1 à J6.2) : Eventuellement, l'autre joueur accepte ou refuse l'offre.
- T525 (J6.2 à J6) : Un joueur fait à nouveau une offre.
- T530 (J6 à J6.2) : Le joueur ne fait pas d'offre de vente, d'achat ou d'échange.
- T535 (J6.2 à J6.3) : Plus aucun joueur ne fait d'offre.
- T540 (J6.3 à J7) : Fin des négociations.
- T550 (J7 à J7.1) : Un joueur retire un ou des titres d'une ou de plusieurs de ses séries.
- T555 (J7.1 à J7) : Un joueur désire manipuler une de ses séries.
- T560 (J7.1 à J7.2) : Un joueur place un ou des titres de propriétés dans une série.
- T565 (J7.2 à J7.3) : Plus aucun joueur ne désire faire de manipulation.
- T570 (J7.1 à J7.3) : Plus aucun joueur ne désire faire de manipulation.
- T580 (J7.3 à J8) : Un joueur termine sa manipulation.

T590 (J7 à J7.2) : Un joueur place un ou des titres dans une série.
 T593 (J7.2 à J7) : Un joueur désire manipuler une de ses séries.
 T595 (J7 à J8) : Aucun joueur ne manipule de séries.
 T600 (J8 à J9) : Le MJ constate qu'il ne reste qu'un joueur.
 T605 (J8 à J4) : Il reste plus d'un joueur et le joueur suivant lance les dés.
 T610 (J9 à J10) : Le MJ ferme le jeu.

On trouvera en fin de l'annexe le réseau de transition tel qu'il apparaît à la fin de cette étape. On remarquera que des jalons étoiles *1 et *2 ont été introduits, ceux-ci correspondant respectivement au remboursement d'un prêt et à la vente d'un lingot d'or. Le développement de ces mini-scénarios se trouve ci-dessous et leurs réseaux de transition en fin de l'annexe.

*1 : (Remboursement d'un prêt)

J*100 : Le joueur courant a demandé le remboursement de son prêt.
 J*101 : Le MJ a remboursé le prêt.
 J*102 : Le MJ a débouté la demande.
 J*103 : Le joueur courant a éventuellement remboursé la demande.

T*110 (J*100 à J*101) : Si la case pour laquelle le prêt est réclamé est à l'état "autorisation de réclamer" pour ce joueur, le MJ rembourse le prêt et les intérêts.
 T*120 (J*100 à J*102) : Si la case pour laquelle le prêt est réclamé est à l'état "interdiction de réclamer" pour ce joueur, le MJ déboute sa demande.
 T*130 (J*101 à J*103) : Fin du remboursement.
 T*140 (J*102 à J*103) : Fin du déboutement de remboursement.

*2 : (vente de lingot d'or).

J*200 : Un joueur a demandé la vente de son or.
 J*201 : Le joueur a lancé les dés.
 J*202 : Le MJ a payé le joueur selon les points obtenus au lancé de dés.
 T*210 (J*200 à J*201) : Le joueur lance les dés.
 T*220 (J*201 à J*202) : Le MJ paie le joueur.

5° SPECIFICATION DES ACTEURS : FORMALISATION EN TERME D'ATTRIBUTS, D'UNE PART, ET CYCLE DE VIE ET REGLES DE COUPLAGE, D'AUTRE PART.

A. Liste des attributs d'instances des acteurs.

JOUEUR : *Num-joueur* : numéro du joueur { J1, J2, J3, J4, J5, J6 }. Il s'agit en fait d'un identifiant de JOUEUR.
Etat-joueur : état dans lequel se trouve le joueur { "en attente", "courant", "en faillite", "en négociation", "créé", "prêt à jouer", "offre de vente", "offre d'achat", "offre d'échange", "plus d'offre", "en manipulation", "plus de manipulation" }.
Montant-joueur : somme d'argent que détient le joueur.
Nom-joueur : nom du joueur.
Num-carte-pro-joueur : liste des numéros de cartes propriété que détient le joueur.
Num-carte-bq-joueur : liste des numéros de cartes de banque que détient le joueur.
Num-série-joueur : liste des numéros de séries que détient le joueur.
Poss-act-num : liste des actions que possède le joueur.
Poss-tab-num : liste des tableaux que possède le joueur.
Poss-prêt-num : liste des prêts que possède le joueur.
Poss-lingot-num : liste des lingots que possède le joueur.

Pass-PA-num : numéro de la PA que possède le joueur. S'il n'en possède pas, la valeur de *Pass-PA-num* est "0".

Pass-has-num : liste des cartes de hasard que possède le joueur.

Position-joueur : position courante du joueur sur le plateau de jeu. Elle vaut "0" si le jeu est fermé.

Réponse-joueur : réponse du joueur {"oui", "non"}

Réponse-joueur-offre : réponse du joueur à l'offre d'un autre joueur

GA : *Montant-FRIK* : somme d'argent que détient la banque FRIK.

Etat-GA : état du GA {"en négociation", "aux enchères", "normal", "en proposition"}.

Joueur-désigné : joueur désigné par le GA en ce moment.

CASE : *Id-case* : identifiant du générique CASE.

Succ-case : identifiant de la case succédant à cette case.

Num-joueur-case : liste des numéros de joueurs présents sur la case.

Etat-case : état de la case {"normal", "déclenché"}.

CASE-PROPRIETE : (spécialisation de CASE)

Type-propriété : type de la propriété {"société", "industrie", "commerce", "immeuble", "banque"}.

Etat-propriété : état de la propriété {"libre", "semi-libre", "occupée"}.

Nbre-carte-pro : nombre de cartes de titre de propriété que possède la case.

Num-carte-pro : liste des numéros de ces cartes.

CASE-BOURSE : (spécialisation de CASE).

CASE-ART : (spécialisation de CASE).

CASE-PRET : (spécialisation de CASE)

Limite-prêt : somme maximale que le joueur peut prêter.

Gère-prêt-num : liste des prêts que gère la case.

CASE-COURTAGE : (spécialisation de CASE)

Montant-cour : somme contenue dans la case courtage.

CASE-OR : (spécialisation de CASE)

CASE-DEPART : (spécialisation de CASE)

CASE-HASARD : (spécialisation de CASE)

CARTE : *Id-carte* : identifiant du générique CARTE.

CARTE-PROPRIETE : (spécialisation de CARTE)

Pourcent-pro : pourcentage que représente le titre.

Prix-pro : prix du titre.

Courtage-pro : courtage à payer lors de l'achat du titre.

Revenu-pro : revenu du titre.

Nom-pro : nom de la propriété.

Num-propriété-pro : numéro de la case propriété à laquelle appartient la carte.

Num-joueur-pro : numéro du joueur à qui appartient le titre. Si la carte n'est pas vendue, la valeur de *Num-joueur-pro* vaut "0".

Etat-carte-pro : état de la carte propriété {"vendue", "invendue"}.

CARTE-BANQUE : (spécialisation de CARTE)

Prix-bq : prix du titre de banque.

Courtage-bq : courtage à payer lors de l'achat du titre.

Revenu-bq : revenu du titre.

Nom-bq : nom de la banque.

Num-propriété-bq : numéro de la case de propriété à laquelle appartient la carte.

Num-joueur-bq : numéro du joueur à qui appartient le titre. Si la carte n'est pas vendue, la valeur de *Num-joueur-bq* vaut "0".

Etat-carte-bq : état de la carte de banque { "vendue", "invendue" }.

Coeff-carte-bq : coefficient de multiplication du revenu.

ACTION : (spécialisation de CARTE)

Prix-act : prix de l'action.

Etat-act : état de l'action { "vendue", "invendue" }.

Num-joueur-act : numéro du joueur à qui appartient le titre. Si la carte n'est pas vendue, la valeur de *Num-joueur-act* vaut "0".

TABLEAU : (spécialisation de CARTE)

Achat-tab : valeur d'achat du tableau.

Revente-tab : valeur de revente du tableau.

Auteur-tab : nom de l'auteur du tableau.

Titre-tab : titre du tableau.

Etat-tab : état de la carte { "vendue", "invendue", "retirée du jeu" }.

Num-joueur-tab : numéro du joueur à qui appartient le tableau. Si la carte n'est pas vendue ou est retirée du jeu, la valeur de *Num-joueur-tab* vaut "0".

PA : (spécialisation de CARTE)

Type-PA : type de la PA { "avec franchise", "sans franchise" }.

Etat-PA : état de la PA { "vendue", "invendue" }.

Prix-PA : prix de la PA.

Franchise-PA : montant de la franchise.

Num-joueur-PA : numéro du joueur à qui appartient la PA. Si la carte n'est pas vendue, la valeur de *Num-joueur-PA* vaut "0".

HASARD : (spécialisation de CARTE)

Texte-has : texte de la carte hasard.

Type-has : type de la carte { "catastrophe", "privilège" }.

Etat-has : état de la carte hasard : { "payée", "non payée", "non distribuée" } dans le cas d'une carte "catastrophe" et { "profitée", "non profitée", "non distribuée" } dans le cas d'une carte "privilège".

Montant-has : montant réclamé dans le cas d'une catastrophe.

Num-joueur-has : numéro du joueur à qui appartient la carte. Si la carte n'est pas distribuée, la valeur de *Num-joueur-has* vaut "0".

PRET : *Num-prêt* : numéro identifiant du prêt.

Montant-prêt : somme que le joueur a prêté.

Etat-prêt : état du prêt { "autorisation de réclamer", "interdiction de réclamer" }.

Géré-par-case-prêt-num : numéro de la case prêt où le prêt a été effectué.

Num-joueur-prêt : numéro du joueur qui a prêté.

LINGOT : *Etat-lingot* : état du lingot { "vendu", "invendu" }.

Num-lingot : numéro identifiant du lingot d'or.

Num-joueur-lingot : numéro du joueur à qui appartient le lingot. S'il n'est pas vendu, la valeur de *Num-joueur-lingot* vaut "0".

SERIE : *Num-série* : numéro identifiant de série.

Num-carte-pro-série : liste des numéros de cartes qui sont dans la série.

Coeff-série : coefficient par lequel il faut multiplier le revenu des titres appartenant à la série { 1, 2, 4, 6 }. Le coefficient vaut 1 dans les cas où la série n'est pas valide.

Etat-série : état de la série { "disponible", "en cours", "complétée" }.

DE : *Id-dé* : identifiant du dé.

Etat-dé : état du dé { "en lancement", "1", "2", "3", "4", "5", "6" }.

JEU : *Etat-jeu* : état du jeu {"ouvert", "fermé"}.

Nbre-avancé : nombre de case dont le joueur courant a déjà avancé.

Somme-dés : nombre de case dont le joueur courant doit avancer.

Nbre-joueur : nombre de joueur courant dans le jeu.

FONDS-BOURSIER : *Montant-fonds* : montant contenu dans le fonds boursier.

NEGOCIATION : *Etat-négociation* : état de la négociation {"pas commencée", "en cours", "terminée"}.

ENCHERES : *Etat-enchères* : état des enchères {"autorisée", "en cours", "terminée"}.

Invendus : liste des invendus

B. Liste des attributs de classe ou de système des acteurs.

JOUEUR :

Vitesse[joueur] = v, où $v > 0$

Nombre[joueur] = 6.

Repr[joueur] = { cercle dans lequel est inscrit le numéro du joueur }.

Fct[joueur] = "partie prenante au jeu".

Compt[joueur] = {

(J1, participer-ouverture-jeu, J2)

(J1.1, **recevoir-argent-début**, J1.2)

(J2, participer-proposition-PA, J3)

(J2.1, **répondre-offre-PA**, J2.2)

(J2.2, **payer-recevoir-PA**, J2.3)

(J3, participer-lancement-dés, J4)

(J3, **devenir-joueur-courant**, J3.1)

(J3.1, **lancer-dés**, J4)

(J4, participer-avancement-case, J5)

(J4, avancer-case-déclenchement-par-dessus, J4.1)

(J4, **avancer-1-case**, J4.0.1)

(J4.0.1, **avancer-1-case-à-nouveau**, J4.0.1)

(J4.0.1, **déclencher-état-autorisation-réclamer**, J4.0.2)

(J4.0.1, **participer-passage-départ**, J4.0.3)

(J4.0.2, **avancer-1-case-à-nouveau**, J4.0.1)

(J4.0.3, **avancer-1-case-à-nouveau**, J4.0.1)

(J4.1, faire-ce-que-peut-doit-faire-case, J5)

(J4.1.1, **payer-somme-catastrophe**, J4.1.2)

(J4.1.1, **profiter-privilege**, J4.1.3)

(J4.1, **empocher-argent-courtage**, J4.1.4)

(J4.1, **réclamer-argent-prêt**, J4.1.5)

(J4.1.5, **prêter-argent**, J4.1.6)

(J4.1, **prêter-argent**, J4.1.6)

(J4.1, **vendre-tableaux**, J4.1.7)

(J4.1.7, **acheter-tableau**, J4.1.8)

(J4.1, **acheter-tableau-sans-vendre**, J4.1.8)

(J4.1, **vendre-actions**, J4.1.9)

(J4.1.9, **acheter-actions**, J4.1.10)

(J4.1, **acheter-actions-sans-vendre**, J4.1.10)

(J4.1, **acheter-lingot**, J4.1.11)

(J4.1, **payer-revenu-titre-semi-libre**, J4.1.12)

(J4.1, **payer-revenu-titre-occupée**, J4.1.13)

(J4.1.14, **acheter-titre-prop-libre-semi-libre**, J4.1.15)

(J4.1.16, **participer-vente-aux-enchères**, J4.1.17)

(J5, **être-éliminé**, J60)

(J6, participer-négociation, J7)

(J6, faire-offre-vente-achat-échange, J6.1)

(J6, **faire-offre-vente**, J6.0.1)

(J6, **faire-offre-achat**, J6.0.2)
 (J6, **faire-offre-échange**, J6.0.3)
 (J6.1, **refuser-accepter-offre**, J6.2)
 (J6.2, **faire-nouvelle-offre**, J6)
 (J6.2, **ne-plus-faire-offre**, J6.3)
 (J7, **manipuler-séries**, J8)
 (J7, **retirer-titres-séries**, J7.1)
 (J7.1, **désirer-manipuler-séries**, J7)
 (J7.1, **placer-titres-séries**, J7.2)
 (J7.1, **ne-plus-désirer-manipuler**, J7.3)
 (J7.2, **ne-plus-désirer-manipuler**, J7.3)
 (J7, **placer-titres-séries-sans-retirer**, J7.2)
 (J7.2, **désirer-manipuler-séries**, J7)
 (J8, **participer-tour-suivant**, J4) }.

Compt[joueur]* = { (*1, **demander-remboursement-prêt**, J*100)
 (J*100, **recevoir-argent-intérêts**, J*101)

 (*2, **demander-vente-or**, J*200)
 (J*200, **lancer-dés**, J*201)
 (J*201, **recevoir-argent-or**, J*202) }.

GA :

Vitesse[GA] = 0.
Nombre[GA] = 1.
Repr[GA] = "Il sera représenté par le dialogue avec les joueurs".
Fct[GA] = "Gérer l'ensemble du jeu".
Compt[GA] = {
 (J1, **commencer-jeu**, J2)
 (J1, **demander-nombre-joueurs**, J1.1)
 (J1.1, **distribuer-argent-joueurs**, J1.2)
 (J1.2, **ouvrir-jeu**, J2)
 (J2, **proposer-PA**, J3)
 (J2, **proposer-PA-joueur-courant**, J2.1)
 (J2.3, **proposer-PA-autres**, J2)
 (J3, **participer-lancé-dés**, J4)
 (J3, **désigner-joueur-courant**, J3.1)
 (J4, **participer-avancement**, J5)
 (J4, **participer-comportement-par-dessus**, J4.1)
 (J4.0.1, **déposer-distribuer-argent-boursier**, J4.0.3)
 (J4.1, **participer-déclenchement-case**, J5)
 (J4.1, **rembourser-prêts-intérêts**, J4.1.5)
 (J4.1, **proposer-achat-titre-important**, J4.1.14)
 (J4.1.14, **proposer-titre-aux-enchères**, J4.1.16)
 (J4.1.16, **gérer-vente-aux-enchères**, J4.1.17)
 (J5, **éliminer-personnes-faillite**, J6)
 (J6, **gérer-négociations**, J7)
 (J8, **constater-fin-du-jeu**, J9)
 (J9, **fermer-le-jeu**, J10) }

Compt[GA]* = {
 (J*100, **rembourser-prêt-intérêts-à-joueur**, J*101)
 (J*100, **débouter-demande-remboursement**, J*102)

 (J*201, **payer-or-à-joueur**, J*202) }.

CASE :

Vitesse[case] = 0.
Nombre[case] = 40.

Repr[case] = "rectangle".
Fct[case] = "lieu d'arrêt des joueurs".
Compt[case] = .

CASE-PROPRIETE :

Vitesse[case-proprété] = 0.
Nombre[case-proprété] = 25.
Repr[case-proprété] = "rectangle avec PROPRIETE inscrit dedans".
Fct[case-proprété] = "Gérer l'ensemble des cartes de propriétés se rapportant à leur case".
Compt[case-proprété] = {
 (J4, participer-avancement, J5)
 (J4.1, participer-déclenchement-case, J5)
 (J4.1, **déclencher-paiement-semi-libre**, J4.1.12)
 (J4.1, **déclencher-paiement-occupée**, J4.1.13)
 (J4.1, **déclencher-propos-achat-libre**, J4.1.14)
 (J4.1.12, **déclencher-propos-achat-semi-libre**, J4.1.14)
 (J4.1.14, **déclencher-vente-aux-enchères**, J4.1.16) }.

CASE-BOURSE :

Vitesse[case-bourse] = 0.
Nombre[case-bourse] = 3.
Repr[case-bourse] = "rectangle avec BOURSE inscrit dedans".
Fct[case-bourse] = "proposer des actions aux joueurs et gérer les cartes d'actions".
Compt[case-bourse] = {
 (J4, participer-avancement, J5)
 (J4.1, participer-déclenchement-case, J5)
 (J4.1, **déclencher-propos-vente-actions**, J4.1.9)
 (J4.1.9, **déclencher-propos-achat-actions**, J4.1.10)
 (J4.1, **déclencher-propos-achat-actions-sans-vendre**, J4.1.10) }.

CASE-ART :

Vitesse[case-art] = 0.
Nombre[case-art] = 3.
Repr[case-art] = "rectangle avec ART inscrit dedans".
Fct[case-art] = "gérer les cartes tableaux".
Compt[case-art] = {
 (J4, participer-avancement, J5)
 (J4.1, participer-déclenchement-case, J5)
 (J4.1, **gérer-vente-par-joueur-tableaux**, J4.1.7)
 (J4.1.7, **gérer-achat-par-joueur-tableau**, J4.1.8)
 (J4.1, **gérer-achat-par-joueur-tableau-sans-vente**, J4.1.8) }.

CASE-PRET :

Vitesse[case-prêt] = 0.
Nombre[case-prêt] = 2.
Repr[case-prêt] = "rectangle avec PRET inscrit dedans".
Fct[case-prêt] = "recevoir et gérer les prêts des joueurs".
Compt[case-prêt] = {
 (J4, participer-avancement, J5)
 (J4, participer-comportement-par-dessus, J4.1)
 (J4.0.1, **changer-ou-non-état-case-prêt**, J4.0.2)
 (J4.1, participer-déclenchement-case, J5)
 (J4.1.5, **recevoir-argent-en-prêt**, J4.1.6)
 (J4.1, **recevoir-argent-en-prêt**, J4.1.6) }.

CASE-COURTAGE :

Vitesse[case-courtage] = 0.
Nombre[case-courtage] = 1.
Repr[case-courtage] = "rectangle avec COURTAGE inscrit dedans".

Fct[case-courtage] = "recueillir l'argent de courtage et le distribuer".

Compt[case-courtage] = {

(J4, participer-avancement, J5)

(J4.1, participer-déclenchement-case, J5)

(J4.1, **donner-argent-courtage**, J4.1.4) }.

CASE-OR :

Vitesse[case-or] = 0.

Nombre[case-or] = 1.

Repr[case-or] = "rectangle avec OR inscrit dedans".

Fct[case-or] = "proposer aux joueurs d'acheter un lingot d'or et gérer l'ensemble des lingots d'or.

Compt[case-or] = {

(J4, participer-avancement, J5)

(J4.1, participer-déclenchement-case, J5)

(J4.1, **lancer-propos-achat-lingot**, J4.1.11) }.

CASE-DEPART :

Vitesse[case-départ] = 0.

Nombre[case-départ] = 1.

Repr[case-départ] = "rectangle avec DEPART inscrit dedans".

Fct[case-départ] = "départ du jeu et déclenche le comportement du GA en ce qui concerne l'alimentation du fonds boursier et la distribution des revenus des actions."

Compt[case-départ] = {

(J4, participer-avancement, J5)

(J4, participer-comportement-par-dessus, J4.1)

(J4.0.1, **déclencher-dépôt-distribution-argent-fonds-boursier**, J4.0.3) }.

CASE-HASARD :

Vitesse[case-hasard] = 0.

Nombre[case-hasard] = 4.

Repr[case-hasard] = "rectangle avec HASARD inscrit dedans".

Fct[case-hasard] = "gérer l'ensemble des cartes de hasard".

Compt[case-hasard] = {

(J4, participer-avancement, J5)

(J4.1, participer-déclenchement-case, J5)

(J4.1, **donner-carte-hasard**, J4.1.1) }.

CARTE :

Vitesse[carte] = 0.

Nombre[carte] = 84.

Repr[carte] = "rectangle".

Fct[carte] = "représenter différents objets".

Compt[carte] = .

CARTE-PROPRIETE :

Vitesse[carte-propriété] = 0.

Nombre[carte-propriété] = 29.

Repr[carte-propriété] = "rectangle avec CARTE-PRO inscrit dedans".

Fct[carte-propriété] = "représenter les titres de propriétés des joueurs".

Compt[carte-propriété] = .

CARTE-BANQUE :

Vitesse[carte-banque] = 0.

Nombre[carte-banque] = 9.

Repr[carte-banque] = "rectangle avec CARTE-BANQUE inscrit dedans".

Fct[carte-banque] = "représenter les titres de banques des joueurs".

Compt[carte-banque] =.

ACTION :

Vitesse[action] = 0.

Nombre[action] = 20.

Repr[action] = "rectangle avec ACTION inscrit dedans".

Fct[action] = "représenter un placement d'argent en actions".

Compt[action] =.

TABLEAU :

Vitesse[tableau] = 0.

Nombre[tableau] = 5.

Repr[tableau] = "rectangle avec TABLEAU inscrit dedans".

Fct[tableau] = "indiquer un placement d'argent dans un tableau".

Compt[tableau] =.

PA :

Vitesse[PA] = 0.

Nombre[PA] = 6.

Repr[PA] = "rectangle avec PA inscrit dedans".

Fct[PA] = "assurer le propriétaire contre certains coûts de catastrophe".

Compt[PA] =.

HASARD :

Vitesse[hasard] = 0.

Nombre[hasard] = 15.

Repr[hasard] = "rectangle avec CARTE-HASARD inscrit dedans".

Fct[hasard] = "donner un privilège ou une catastrophe aux joueurs".

Compt[hasard] =.

PRET :

Vitesse[prêt] = 0.

Nombre[prêt] = p, où $p \geq 0$.

Repr[prêt] =.

Fct[prêt] = "représenter un prêt qu'un joueur a fait".

Compt[prêt] =.

LINGOT :

Vitesse[lingot] = 0.

Nombre[lingot] = 3.

Repr[lingot] = "trapèze".

Fct[lingot] = "représenter un placement d'argent en or".

Compt[lingot] =.

SERIE :

Vitesse[série] = 0.

Nombre[série] = s, où $s \geq 0$.

Repr[série] =.

Fct[série] = "rassembler différents titres afin d'augmenter leurs revenus".

Compt[série] =.

DE :

Vitesse[dé] = 0.

Nombre[dé] = 2.

Repr[dé] = "carré avec un certain nombre de points (1 à 6)".
Fct[dé] = "permet de savoir de combien de cases avancer".
Compt[dé] = .

JEU :

Vitesse[jeu] = 0.
Nombre[jeu] = 1.
Repr[jeu] = "grand cercle-plateau de jeu".
Fct[jeu] = "cadre du jeu".
Compt[jeu] = .

FONDS-BOURSIER :

Vitesse[fonds-boursier] = 0.
Nombre[fonds-boursier] = 1.
Repr[fonds-boursier] = "rectangle avec un montant inscrit dedans".
Fct[fonds-boursier] = "indique la somme d'argent liée aux actions".
Compt[fonds-boursier] = .

NEGOCIATION :

Vitesse[négociation] = 0.
Nombre[négociation] = 1.
Repr[négociation] = .
Fct[négociation] = "voir où en est la négociation".
Compt[négociation] = .

ENCHERES :

Vitesse[enchères] = 0.
Nombre[enchères] = 1.
Repr[enchères] = .
Fct[enchères] = "voir où en est la vente aux enchères".
Compt[enchères] = .

C. Cycle de vie des acteurs.

On trouvera en fin de l'annexe les cycles de vie des différents acteurs.

D. Règles de couplage.

Les seules règles de couplage relevées dans le système sont :

Si Etat-jeu = "fermé" alors

Etat-GA = "normal" ET Etat-propriété = "libre" ET Etat-carte-pro = "invendue"
ET Etat-carte-bq = "invendue" ET Etat-act = "invendue" ET Etat-tab = "invendue"
ET Etat-PA = "invendue" ET Etat-has = "non distribuée" ET Etat-lingot =
"invendu" ET Etat-négociation = "terminée" ET Etat-enchères = "terminée"

Si Etat-propriété = "libre" alors

Etat-carte-pro(: CARTE-PROPRIETE(: Num-propriété-pro = Id-case(: CASE-
PROPRIETE(: Etat-propriété = "libre")))) = "invendue"

Si Etat-propriété = "occupée" alors

Etat-carte-pro(: CARTE-PROPRIETE(: Num-propriété-pro = Id-case(: CASE-
PROPRIETE(: Etat-propriété = "occupée")))) = "vendue"

Si Etat-carte-pro(: CARTE-PROPRIETE) = "invendue" alors

Num-joueur-pro(: CARTE-PROPRIETE) = 0

6° SPECIFICATION FORMELLE DES JALONS.

A. Abréviations utilisée :

dép = sur la case-départ
has = sur une case-hasard
cou = sur la case-courtage
pre = sur une case-prêt
art = sur une case-art
bou = sur une case-bourse
or = sur la case-or
prop = sur une case-propriété
agd = argent que le joueur reçoit au début (selon le nombre de joueurs)
mj = montant contenu dans Montant-joueur[Ji]
s = somme des points obtenus aux dés
p = identifiant de la position du joueur avant qu'il n'avance d'une case
mf = montant contenu dans Montant-fonds
na = nombre de case dont le joueur a déjà avancé
mc = montant contenu dans Montant-cour[Case-courtage]
mp = montant contenu dans Montant-prêt[Prêt]
cgpei = carte qui a le plus gros pourcentage parmi les cartes propriété d'une case invendues
cpgr = carte de banque avec le plus gros revenu
nbrej = nombre de joueurs
nbref = nombre de joueurs en faillite
Sj = série d'un joueur j
Ji = joueur i
Jj = joueur j
JGA = joueur désigné par le GA

B. Première partie :

J1 : Etat-jeu = "fermé"
J1.1 : Nbre-joueur-jeu = x, où $2 \leq x \leq 6$
J1.2 : Montant-joueur = agd
J2 : Etat-jeu = "ouvert" ET Montant-joueur = agd
ET Etat-GA = "normal"
J2.1 : Joueur-désigné[GA] = Ji ET Etat-GA = "en proposition"
J2.2 : Réponse-joueur[Ji] = ("oui" OU "non")
J2.3 : [Réponse-joueur[Ji] = "oui"
ET Montant-joueur[Ji] = mj - Prix-PA[PA tirée]
ET Poss-PA-num = Id-carte[PA tirée]
ET Montant-FRIK[GA] = Montant-FRIK[GA] + Prix-PA[PA tirée]]
OU
[Réponse-joueur[Ji] = "non"]
J3 : Etat-GA = "normal"
ET Etat-joueur[Ji] = "en attente" où $2 \leq i \leq nbrej$
J3.1 : Etat-joueur[Ji] = "courant"
ET Etat-joueur[Jj] = "en attente" avec $j \neq i$
ET Joueur-désigné[GA] = Ji
J4 : Etat-dé = i, $1 \leq i \leq 6$
ET Somme-dé[Jeu] = s
ET Nbre-avancé[Jeu] = 0
J4.0.1 : Position-joueur[Ji] = Succ-case[p]
ET Nbre-avancé[Jeu] = na + 1
J4.0.2 : [Position-joueur[Ji] = pre
ET Gère-prêt-num[Case-prêt] Num-prêt[Prêt : Num-joueur-
prêt = Ji] $\neq \emptyset$
ET Etat-prêt = "autorisation de réclamer"]

OU

[Position-joueur[J1] = pre
ET Gère-prêt-num[Case-prêt] Num-prêt[Prêt : Num-joueur-prêt = Ji]
= Ø]

J4.0.3 : [Position-joueur[J1] = dép
ET Montant-fonds = mf + 100000
ET Montant-joueur[J1] = mj + (0.05 * taille{Poss-act-num} * mf)]

J4.1 : Nbre-avancé[Jeu] = Somme-dé[Jeu]

ET Etat-case = "déclenchée"

J4.1.1 : Position-joueur[J1] = has

ET [(Type-has = "catastrophe"

ET Etat-has = "non payée")

OU

(Type-has = "privilège"

ET Etat-has = "non profitée")]

ET Num-joueur-case = Num-joueur-case U {Num-joueur[J1]}

ET Position-joueur[J1] = Id-case

J4.1.2 : Type-has = "catastrophe"

ET Etat-has[carte tirée] = "payée"

J4.1.3 : Type-has = "privilège"

ET [(Etat-has = "profitée")

OU

(Etat-has = "non profitée"

ET Poss-has-num = Poss-has-num U {Id-carte})]

J4.1.4 : Position-joueur[J1] = cour

ET Montant-joueur[J1] = mj + mc

ET Montant-cour = 0

ET Num-joueur-case = Num-joueur-case U {Num-joueur[J1]}

ET Position-joueur[J1] = Id-case

J4.1.5 : Position-joueur[J1] = pre

ET [(Montant-joueur[J1] = mj + (mp * 1.5)

ET Gère-prêt-num = Gère-prêt-num \ {num-prêt}

ET Montant-FRIK = Montant-FRIK - (mp * 1.5))

OU

(Montant-joueur[J1] = mj)]

ET Num-joueur-case = Num-joueur-case U {Num-joueur[J1]}

ET Position-joueur[J1] = Id-case

J4.1.6 : Position-joueur[J1] = pre

ET [(Réponse-joueur[J1] = "oui"

ET Montant-prêt = x

ET Etat-prêt = "interdiction de réclamer"

ET Géré-par-case-prêt-num = Id-case

ET Num-joueur-prêt = Num-joueur[J1]

ET Gère-prêt-num[Case-prêt] = num-prêt

ET Montant-joueur[J1] = mj - x

ET Montant-FRIK = Montant-FRIK + x)

OU

(Réponse-joueur[J1] = "non"

ET Montant-prêt = 0)]

ET Num-joueur-case = Num-joueur-case U {Num-joueur[J1]}

ET Position-joueur[J1] = Id-case

J4.1.7 : Position-joueur[J1] = art

ET [(Réponse-joueur[J1] = "oui"

ET Montant-joueur[J1] = mj + {Revente-tab[tableaux choisis]}

ET Num-joueur-tab[tableaux choisis] = 0

ET Etat-tab[tableaux choisis] = "retirée du jeu"

ET Poss-tab-num = Poss-tab-num \ {Id-carte[tableaux choisis]}

ET Montant-FRIK = Montant-FRIK - {Revente-tab

[tableaux choisis]})

OU

(Réponse-joueur[Ji] = "non")]

ET Num-joueur-case = Num-joueur-case U { Num-joueur[Ji] }

ET Position-joueur[Ji] = Id-case

J4.1.8 : Position-joueur[Ji] = art

ET [(Réponse-joueur[Ji] = "oui"

ET Montant-joueur[Ji] = mj - achat-tab[tableau
choisi]

ET Num-joueur-tab[tableau choisi] = Num-joueur

ET Etat-tab[tableau choisi] = "vendu"

ET Poss-tab-num = Poss-tab-num U { Id-carte[tableau choisi] }

ET Montant-FRIK = Montant-FRIK + achat-tab[tableau choisi])

OU

(Réponse-joueur[Ji] = "non")]

ET Num-joueur-case = Num-joueur-case U { Num-joueur[Ji] }

ET Position-joueur[Ji] = Id-case

J4.1.9 : Position-joueur[Ji] = bou

ET [(Réponse-joueur[Ji] = "oui"

ET Montant-joueur = mj + 50000 * (Nbre d'actions
revendues)

ET Montant-FRIK = Montant-FRIK - 50000 * (Nbre
d'actions revendues)

ET Num-joueur-act[actions revendues] = 0

ET Etat-act[actions revendues] = "invendues"

ET Poss-act-num = Poss-act-num \ { Id-carte [actions
revendues] })

OU

(Réponse-joueur[Ji] = "non")]

ET Num-joueur-case = Num-joueur-case U { Num-joueur[Ji] }

ET Position-joueur[Ji] = Id-case

J4.1.10 : Position-joueur[Ji] = bou

ET [(Réponse-joueur[Ji] = "oui"

ET Montant-joueur = mj - 50000 * (Nbre d'actions achetées)

ET Montant-FRIK = Montant-FRIK + 50000 * (Nbre
d'actions achetées)

ET Num-joueur-act[actions achetées] = Num-joueur

ET Etat-act[actions achetées] = "vendue"

ET Poss-act-num = Poss-act-num U { Id-carte [actions
achetées] })

OU

(Réponse-joueur[Ji] = "non")]

ET Num-joueur-case = Num-joueur-case U { Num-joueur[Ji] }

ET Position-joueur[Ji] = Id-case

J4.1.11 : Position-joueur[Ji] = or

ET [(Réponse-joueur[Ji] = "oui"

ET Montant-joueur = mj - 200000

ET Montant-FRIK = Montant-FRIK + 200000

ET Num-joueur-lingot = Num-joueur

ET Etat-lingot = "vendu"

ET Poss-lingot-num = Poss-lingot-num U { num-lingot })

OU

(Réponse-joueur[Ji] = "non")]

ET Num-joueur-case = Num-joueur-case U { Num-joueur[Ji] }

ET Position-joueur[Ji] = Id-case

J4.1.12 : • Position-joueur[Ji] = prop

ET Etat-propriété = "Semi-libre"

ET Type-propriété = ("société" **OU** "industrie" **OU**
"commerce" **OU** "immeuble")

ET [(Montant-joueur[Ji] = mj - Revenu-pro[Carte-propriété vendues]

ET Num-carte-pro[Case-propriété] Num-carte-pro-

```

    joueur[Ji] = Ø )
    OU
    ( Montant-joueur[Ji] = mj
      ET Num-carte-pro[Case-propriété] Num-carte-pro-
        joueur[Ji] ≠ Ø ) ]
    ET Num-joueur-case = Num-joueur-case U {Num-joueur[Ji]}
    ET Position-joueur[Ji] = Id-case
    OU • Position-joueur[Ji] = prop
      ET Etat-propriété = "Semi-libre"
      ET Type-propriété = "banque"
      ET [ ( Montant-joueur[Ji] = mj - Revenu-bq[Carte-propriété
        vendues]
          ET Num-carte-pro[Case-propriété] Num-carte-bq-
            joueur[Ji] = Ø )
          OU
          ( Montant-joueur[Ji] = mj
            ET Num-carte-pro[Case-propriété] Num-carte-bq-
              joueur[Ji] ≠ Ø ) ]
      ET Num-joueur-case = Num-joueur-case U {Num-joueur[Ji]}
      ET Position-joueur[Ji] = Id-case
    J4.1.13 : • Position-joueur[Ji] = prop
      ET Etat-propriété = "occupée"
      ET Type-propriété = ( "société" OU "industrie" OU
        "commerce" OU "immeuble" )
      ET [ ( Montant-joueur[Ji] = mj - Revenu-pro[Carte-propriété
        vendues] * coeff-série
          ET Num-carte-pro[Case-propriété] Num-carte-pro-
            joueur[Ji] = Ø )
          OU
          ( Montant-joueur[Ji] = mj - Revenu-pro[Carte-propriété
            vendues]
            ET Num-carte-pro[Case-propriété] Num-carte-pro-
              joueur[Ji] = Ø )
          OU
          ( Montant-joueur[Ji] = mj
            ET Num-carte-pro[Case-propriété] Num-carte-
              pro-joueur[Ji] ≠ Ø ) ]
      ET Num-joueur-case = Num-joueur-case U {Num-joueur[Ji]}
      ET Position-joueur[Ji] = Id-case
    OU • Position-joueur[Ji] = prop
      ET Etat-propriété = "occupée"
      ET Type-propriété = "banque"
      ET [ ( Montant-joueur[Ji] = mj - Revenu-bq[Carte-propriété
        vendues] * coeff-carte-bq
          ET Num-carte-pro[Case-propriété] Num-carte-bq-
            joueur[Ji] = Ø )
          OU
          ( Montant-joueur[Ji] = mj
            ET Num-carte-pro[Case-propriété] Num-carte-bq-
              joueur[Ji] ≠ Ø ) ]
      ET Num-joueur-case = Num-joueur-case U {Num-joueur[Ji]}
      ET Position-joueur[Ji] = Id-case
    J4.1.14 : Position-joueur[Ji] = prop
      ET Etat-propriété = ( "semi-libre" OU "libre" )
      ET Etat-GA = "en proposition"
      ET [ ( Etat-carte-pro[cgpei] = "invendue" )
          OU
          ( Etat-carte-bq[cpgr] = "invendue" ) ]

```

J4.1.15 : • Position-joueur[Ji] = prop
 ET Etat-propriété =("semi-libre" OU "libre")
 ET Type-propriété =("société" OU "industrie" OU
 "commerce" OU "immeuble")
 ET [(Réponse-joueur[Ji] = "oui"
 ET Montant-joueur[Ji] = mj - Prix-pro[cgpei]
 ET Etat-carte-pro[cgpei] = "vendue"
 ET Num-carte-pro-joueur = Num-carte-pro-joueur U
 {Id-carte[cgpei]}
 ET Num-joueur-pro[Carte-propriété] = Num-joueur[Ji])
 OU
 (Réponse-joueur[Ji] = "non")]
 OU • Position-joueur[Ji] = prop
 ET Etat-propriété =("semi-libre" OU "libre")
 ET Type-propriété = "banque"
 ET [(Réponse-joueur[Ji] = "oui"
 ET Montant-joueur[Ji] = mj - Prix-bq[cpgr]
 ET Etat-carte-bq[cpgr] = "vendue"
 ET Num-carte-bq-joueur = Num-carte-bq-joueur U
 {Id-carte[cpgr]}
 ET Num-joueur-pro[Carte-propriété] = Num-joueur[Ji])
 OU
 (Réponse-joueur[Ji] = "non")]
 J4.1.16 : Position-joueur[Ji] = prop
 ET Etat-propriété =("semi-libre" OU "libre")
 ET Etat-GA = "aux enchères"
 ET [(Etat-carte-pro[cgpei] = "invendue")
 OU
 (Etat-carte-bq[cpgr] = "invendue")]
 ET Etat-enchères = "autorisées"
 J4.1.17 : • Etat-propriété =("semi-libre" OU "libre")
 ET Etat-GA = "aux enchères"
 ET [(Réponse-joueur[JGA] = "oui"
 ET Montant-joueur[JGA] = mj - Prix-pro[cgpei]
 ET Etat-carte-pro[cgpei] = "vendue"
 ET Num-carte-pro-joueur = Num-carte-pro-joueur U
 {Id-carte[cgpei]}
 ET Num-joueur-pro[Carte-propriété] = Num-
 joueur[JGA]
 ET Etat-enchères = "en cours")
 OU
 (Réponse-joueur[JGA] = "non"
 ET Etat-carte-pro[cgpei] = "invendue"
 ET Invendus[Enchères] = Invendus[Enchères] U {Id-carte[cgpei]
 ET Etat-enchères = "terminée")]
 OU • Etat-propriété =("semi-libre" OU "libre")
 ET Etat-GA = "aux enchères"
 ET [(Réponse-joueur[JGA] = "oui"
 ET Montant-joueur[JGA] = mj - Prix-bq[cpgr]
 ET Etat-carte-bq[cpgr] = "vendue"
 ET Num-carte-bq-joueur = Num-carte-bq-joueur U
 {Id-carte[cpgr]}
 ET Num-joueur-pro[Carte-propriété] = Num-
 joueur[JGA] ET Etat-enchères = "en cours")
 OU
 (Réponse-joueur[JGA] = "non"
 ET Etat-carte-bq[cpgr] = "invendue"
 ET Invendus[Enchères] = Invendus[Enchères] U {Id-carte[cpgr]
 ET Etat-enchères = "terminée")]

J5 : Etat-GA = "normal"
 ET Etat-joueur = ("en attente" OU "en faillite")
 ET Etat-case = "normal"

J6 : Nbre-joueur[Jeu] = Nbrej - nbref

J6.0.1 : Réponse-joueur[Ji] = "oui"
 ET Etat-joueur[Ji] = "offre de vente"
 ET Etat-négociation = "en cours"

J6.0.2 : Réponse-joueur[Ji] = "oui"
 ET Etat-joueur[Ji] = "offre d'achat"
 ET Etat-négociation = "en cours"

J6.0.3 : Réponse-joueur[Ji] = "oui"
 ET Etat-joueur[Ji] = "offre d'échange"
 ET Etat-négociation = "en cours"
 ET Joueur-concerné = Jj avec $j \neq i$

J6.1 : (Etat-négociation = "en cours"
 ET Etat-GA = "en négociation")

J6.2 : Réponse-joueur-offre[Jj] = ("oui" OU "non")
 OU
 Réponse-joueur[Ji] = "non"

J6.3 : Etat-négociation = "terminée"
 ET Etat-GA = "en négociation"
 ET Etat-joueur[Ji] = "plus d'offre", $2 \leq i \leq \text{nbref}$

J7 : Etat-GA = "normal"

J7.1 : (Etat-série[Sj] = "en cours"
 ET Etat-série[Si] = ("disponible" OU "complétée"), avec $i \neq j$
 ET Etat-joueur[Ji] = "en manipulation"
 ET Réponse-joueur = "oui")
 OU
 (Etat-joueur[Ji] = "en manipulation"
 ET Réponse-joueur = "non")

J7.2 : (Etat-série[Sj] = "en cours"
 ET Etat-série[Si] = ("disponible" OU "complétée"), avec $i \neq j$
 ET Etat-joueur[Ji] = "en manipulation"
 ET Réponse-joueur = "oui")
 OU
 (Etat-joueur[Ji] = "en manipulation"
 ET Réponse-joueur = "non")

J7.3 : (Etat-série[Sj] = "complétée"
 ET Etat-joueur[Ji] = "plus de manipulation" $2 \leq i \leq \text{nbref}$

J8 : Etat-joueur = "en attente"
 ET Etat-série = "complétée"

J9 : Nbre-joueur[Jeu] = ("0" OU "1")

J10 : Etat-jeu = "fermé"

B. Seconde partie : spécification formelle contrôlée et complétée :

Le jalon J1.2 correspond plutôt à un nouvel état de joueur, à savoir que le joueur est "prêt à jouer". Cela nécessite à nouveau la création d'un autre état : celui de "créé" pour pouvoir définir de manière formelle l'état des joueurs lorsqu'ils viennent d'être créés (cfr J1.1).

J1.2 : Etat-joueur[Ji] = "prêt à jouer" où $2 \leq i \leq \text{nbref}$

J1.1 : Nbre-joueur-jeu = x, où $2 \leq x \leq 6$

ET Etat-joueur[Ji] = "créé" avec $2 \leq i \leq \text{nbref}$

On trouvera en fin de l'annexe le nouveau cycle de vie du JOUEUR

L'état "vendue" de Etat-PA n'a pas été coché. Il s'agit en fait d'un oubli dans le jalon J2.3.

L'état est donc ajouté dans le jalon.

J2.3 : [Réponse-joueur[Ji] = "oui"

ET Montant-joueur[Ji] = mj - Prix-PA[PA tirée]

ET Poss-PA-num = Id-carte[PA tirée]

ET Montant-FRIK[GA] = Montant-FRIK[GA] + Prix-PA[PA
tirée]

ET Etat-PA[PA-tirée] = "vendue"]

OU

[Réponse-joueur[Ji] = "non"]

Les états "invendue" de Etat-tab, "invendue" de Etat-PA, "non distribuée" de Etat-has, "invendu" de Etat-lingot n'ont pas été cochés. Cela fait apparaître le problème de l'initialisation.

J1 : Etat-jeu = "fermé"

ET Etat-GA = "normal"

ET Etat-propriété = "libre"

ET Etat-carte-pro = "invendue"

ET Etat-carte-bq = "invendue"

ET Etat-act = "invendue"

ET Etat-tab = "invendue"

ET Etat-PA = "invendue"

ET Etat-has = "non distribuée"

ET Etat-lingot = "invendu"

ET Etat-négociation = "terminée"

ET Etat-enchères = "terminée"

7° DESCRIPTION DU PLAN DE SYSTEME ET DU DIAGRAMME DE TRANSITION DE CHAQUE ACTEUR.

Les plans de système et les diagrammes de transition des acteurs se trouvent en fin d'annexe. Les relevés des accumulations, des acteurs externes et des processus pour chaque acteur sont ci-dessous.

1. Le meneur de jeu :

Les accumulations relevées pour le système du meneur de jeu sont :

A1 : Jeu

A2 : Enchères

A3 : Négociation

A4 : Dé

A5 : PA

Les acteurs externes qui interagissent avec le système sont :

E3 : Joueur

E4 : Case-Propriété

E5 : Case-Bourse

E6 : Case-Art

E7 : Case-Prêt

E8 : Case-Hasard

E9 : Case-Courtage

E10 : Case-Or

E11 : Case-Départ

Les 15 processus apparaissant dans les diagrammes sont :

P1 : Le MJ demande le nombre de joueurs, distribue l'argent et ouvre le jeu (T10, T20, T30)

P2 : Le MJ propose aux joueurs d'acheter une PA (T40, T60, T70, T80)

P3 : Le MJ désigne le joueur courant et celui-ci lance les dés (T90, T100)

P4 : Le joueur avance du nombre de cases voulu (T110, T115, T120, T130, T140, T150, T155, T160, T170)

P5 : Si case hasard, gérer d'après le type de la carte (T180, T190, T200, T400, T410)

P6 : Si case courtage, donner au joueur la somme contenue dans la case (T210, T420)

- P7 : Si case prêt, gérer réclamation argent et prêt du joueur (T220, T230, T240, T440)
- P8 : Si case art, gérer vente et achat de tableaux (T250, T260, T270, T460)
- P9 : Si case bourse, gérer vente et achat actions (T280, T290, T300, T480)
- P10 : Si case or, gérer achat d'un lingot d'or (T310, T490)
- P11 : Si case propriété, gérer paiement revenu, achat de titre par joueur, vente de titre aux enchères (T320, T330, T335, T340, T350, T360, T365, T370, T375, T380, T390, T395)
- P12 : Elimination des joueurs en faillite (T500)
- P13 : Gérer proposition de négociation des joueurs (T502, T504, T506, T510, T512, T515, T525, T530, T535, T540)
- P14 : Gérer manipulation des séries (T550, T555, T560, T565, T570, T580, T590, T593, T595)
- P15 : Voir si fin du jeu par le nombre de joueurs restant (T600, T605, T610)

2. La caméra :

L'accumulation relevée pour le système de la caméra est :

A1 : Données relatives au déplacement et à l'affichage

Les acteurs externes qui interagissent avec le système sont :

- E1 : MJ
- E3 : Joueur
- E4 : Case-Propriété
- E5 : Case-Bourse
- E6 : Case-Art
- E7 : Case-Prêt
- E8 : Case-Hasard
- E9 : Case-Courtage
- E10 : Case-Or
- E11 : Case-Départ

Les 19 processus apparaissant dans les diagrammes sont :

- P1 : Afficher demande du nombre de joueurs (T10)
- P2 : Afficher proposition d'achat PA (T40)
- P3 : Afficher résultat des dés (T100, T605)
- P4 : Déplacer le joueur d'une case
- P5 : Afficher dépôt 100000\$ dans le fonds boursier et bénéfice des actions (T130)
- P6 : Afficher case hasard et carte hasard (T180)
- P7 : Afficher case courtage et montant contenu (T210)
- P8 : Afficher case prêt et proposition de prêt (T220, T230, T240)
- P9 : Afficher case art, proposition vente et achat de tableaux (T250, T260, T270)
- P10 : Afficher case bourse, proposition vente et achat d'actions (T280, T290, T300)
- P11 : Afficher case lingot et proposition achat de lingot (T310)
- P12 : Afficher case propriété, paiement revenu, vente de titre (T320, T330, T340, T350, T370, T380)
- P13 : Afficher élimination du joueur (T500)
- P14 : Afficher proposition de négociation (T502, T504, T506, T525)
- P15 : Afficher manipulation de séries (T550, T555, T560, T565, T570, T580, T590, T593, T595)
- P16 : Placer les joueurs sur la case départ (T30)
- P17 : Afficher tableau de bord du joueur (T90)
- P18 : Afficher fermeture du jeu (T610)
- P19 : Afficher début du jeu (T10)

3. Le joueur :

Aucune accumulation n'est relevée pour le système du joueur.

Les acteurs externes qui interagissent avec le système sont :

- E1 : MJ
- E2 : Caméra
- E4 : Case-Propriété
- E5 : Case-Bourse
- E6 : Case-Art
- E7 : Case-Prêt
- E8 : Case-Hasard
- E9 : Case-Courtage
- E10 : Case-Or
- E11 : Case-Départ

Les 20 processus apparaissant dans les diagrammes sont :

- P1 : Recevoir l'argent pour jouer (T20)
- P2 : Le joueur achète éventuellement une PA (T50, T60)
- P3 : Devenir le joueur courant et lancer les dés (T90, T100)
- P4 : Avancer d'une case (T110, T140, T150)
- P5 : Déclencher autorisation de réclamer (T120)
- P6 : Participer au passage par dessus case départ (T130)
- P7 : Payer la somme de la carte hasard (T190)
- P8 : Profiter du privilège de la carte hasard (T200)
- P9 : Empocher la somme contenue dans la case courtage (T210)
- P10 : Prêter de l'argent au GA (T230, T240)
- P11 : Vendre et acheter des tableaux (T250, T260, T270)
- P12 : Vendre et acheter des actions (T280, T290, T300)
- P13 : Acheter un lingot d'or (T310)
- P14 : Payer le revenu d'une propriété (T320, T330)
- P15 : Acheter un titre de propriété (T360, T365)
- P16 : Participer à la vente aux enchères (T375, T380, T390)
- P17 : Etre éliminé (T500)
- P18 : Faire une offre de vente, d'achat ou d'échange (T502, T504, T506, T510, T512, T515, T525, T530, T535)
- P19 : Refuser ou accepter l'offre (T520)
- P20 : Manipuler une de ses séries (T555, T560, T565, T570, T580, T590, T593, T595)

4. La case propriété :

Les accumulations relevées pour la case propriété sont :

- A1 : Carte-Propriété
- A2 : Carte-Banque

Les acteurs externes qui interagissent avec le système sont :

- E1 : MJ
- E2 : Caméra
- E3 : Joueur

Les 5 processus apparaissant dans les diagrammes sont :

- P1 : Déclencher le paiement si la propriété est semi-libre (T320)
- P2 : Déclencher le paiement si la propriété est occupée (T330)
- P3 : Déclencher proposition d'achat du titre si la propriété est libre (T340, T360, T365)
- P4 : Déclencher proposition d'achat du titre si la propriété est semi-libre (T350, T360, T365)
- P5 : Déclencher la vente aux enchères si le titre n'est pas vendu (T370)

5. La case bourse :

L'accumulation relevée pour le système de la case bourse est :

A1 : Actions

Les acteurs externes qui interagissent avec le système sont :

E1 : MJ

E2 : Caméra

E3 : Joueur

Les 2 processus apparaissant dans les diagrammes sont :

P1 : Déclencher la vente d'actions (T280)

P2 : Déclencher achat d'actions (T290, T300)

6. La case art :

L'accumulation relevée pour le système de la case art est :

A1 : Tableaux

Les acteurs externes qui interagissent avec le système sont :

E1 : MJ

E2 : Caméra

E3 : Joueur

Les 2 processus apparaissant dans les diagrammes sont :

P1 : Déclencher la vente de tableaux (T250)

P2 : Déclencher achat de tableaux (T260, T270)

7. La case hasard :

L'accumulation relevée pour le système de la case hasard est :

A1 : Hasard

Les acteurs externes qui interagissent avec le système sont :

E1 : MJ

E2 : Caméra

E3 : Joueur

Les 3 processus apparaissant dans les diagrammes sont :

P1 : Donner carte de hasard (T180)

P2 : Déclencher paiement de la carte "catastrophe" (T190)

P3 : Déclencher profiter de la carte "privilège" (T200)

8. La case prêt :

L'accumulation relevée pour le système de la case prêt est

A1 : Prêt

Les acteurs externes qui interagissent avec le système sont :

E1 : MJ

E2 : Caméra

E3 : Joueur

Les 2 processus apparaissant dans les diagrammes sont :

P1 : Déclencher changement ou non état du prêt (T120)

P2 : Déclencher réception d'un prêt (T230)

9. La case or :

L'accumulation relevée pour le système de la case or est :

A1 : Lingot

Les acteurs externes qui interagissent avec le système sont :

E1 : MJ

E2 : Caméra
E3 : Joueur

Le processus apparaissant dans les diagrammes est :
P1 : Lancer la proposition d'achat d'un lingot (T310)

10. La case courtage :

Il n'y a pas d'accumulation pour le système de la case courtage.
Les acteurs externes qui interagissent avec le système sont :

E1 : MJ
E2 : Caméra
E3 : Joueur

Le processus apparaissant dans les diagrammes est :
P1 : Faire empocher la somme de courtage (T210)

11. La case départ :

L'accumulation relevée pour le système de la case départ est :
A1 : Fonds-Boursier

Les acteurs externes qui interagissent avec le système sont :

E1 : MJ
E2 : Caméra
E3 : Joueur

Le processus apparaissant dans les diagrammes est :
P1 : Déclencher dépôt et distribution de l'argent du fonds boursier (T130)

8° SPECIFICATION PAR PREDICATS.

1. Dictionnaire des prédicats utilisés.

Acheté-Act (Joueur-x, Nbre) : le joueur Joueur-x a acheté un nombre Nbre d'actions.
Acheté-Tab (Joueur-x, Oui/Non) : le joueur Joueur-x a/n'a pas acheté de tableau.
Avancé-Nbre-Voulu (Oui/Non) : indique si le joueur courant a/n'a pas avancé du nombre de cases voulu.
Déclenchement-Départ (Oui/Non) : indique si le déclenchement par-dessus la case départ est/n'est pas effectué.
Déclenchement-Prêt (Oui/Non) : indique si le déclenchement par-dessus la case prêt est/n'est pas effectué.
Déposé (GA, Somme) : le GA a déposé la somme d'argent Somme dans le fonds boursier.
Etat-Case (Id-case, Etat-x) : la case identifiée par Id-case est à l'état Etat-x.
Etat-Dé (i, j) : l'état du Dé-1 = i et l'état du Dé-2 = j, $1 \leq i, j \leq 6$.
Etat-Enchères (Etat-x) : les enchères sont dans l'état Etat-x.
Etat-GA (Etat-x) : le GA est dans l'état Etat-x.
Etat-has (Carte-has-y, Etat-has) : la carte de hasard Carte-has-y est dans l'état Etat-has.
Etat-Jeu (Etat-x) : le jeu est dans l'état Etat-x.
Etat-Joueur (Joueur-x, Etat-y) : le joueur Joueur-x est dans l'état Etat-y.
Etat-Lingot (Lingot-x, Etat-y) : le lingot Lingot-x est dans l'état Etat-y.
Etat-Négociation (Etat-x) : la négociation est dans l'état Etat-x.
Etat-Prêt (Prêt-x, Id-case, Joueur-y, Etat-z) : le prêt Prêt-x effectué par le joueur Joueur-y sur la case de prêt Id-case est dans l'état Etat-z.
Etat-Propriété (Id-case, Etat-y) : l'état de la case propriété Id-case est dans l'état Etat-y.
Etat-Tab (Tab-a, Etat-y) : le tableau Tab-a est dans l'état Etat-y.
Etat-Titre (Titre-y, Etat-z) : le titre Titre-y est dans l'état Etat-z.
Joueur-A-Avancé (Oui/Non) : indique si le joueur a/n'a pas avancé.
Joueur-Désigné-GA (Joueur-x) : le joueur Joueur-x est désigné par le GA.

Joueur-Eliminés (Oui/Non) : les joueurs en faillite ont/n'ont pas été éliminés.
 Nbre-Avancé (1) : le nombre de case dont le joueur courant a déjà avancé est 1.
 Nbre-Jeu (Nbre) : il y a un nombre Nbre de joueurs dans le jeu.
 Num-Joueur-Has (Carte-has-y, Joueur-x) : la carte de hasard Carte-has-y est possédée par le Joueur-x.
 Offre-Effectuée (Oui/Non) : l'offre du joueur est/n'est pas effectuée.
 Position (Joueur-x, Type-case, Id-case) : le joueur Joueur-x est sur une case de type Type-case dont l'identifiant est Id-case.
 Possède-Act-N (Joueur-x, Nbre) : le joueur Joueur-x possède un nombre Nbre d'actions.
 Possède-Has (Joueur-x, Carte-has-y) : le joueur Joueur-x possède la carte de hasard Carte-has-y.
 Possède-Lingot (Joueur-x, Lingot-y) : le joueur Joueur-x possède le lingot d'or Lingot-y.
 Possède-PA (Joueur-x, Id-carte-PA) : le joueur Joueur-x possède la PA identifiée par Id-carte-PA.
 Possède-Prêt (Joueur-x, Prêt-y, Id-case, Montant, Etat-z) : le joueur Joueur-x possède un prêt Prêt-y sur la case de prêt Id-case, pour une somme Montant et qui est à l'état Etat-z.
 Possède-Prêt-N (Joueur-x, Id-case, Nbre-prêt) : le joueur Joueur-x possède un nombre Nbre-prêt de prêts sur la case de prêt Id-case.
 Possède-Tab (Joueur-x, Tab-y) : le joueur Joueur-x possède le tableau Tab-y.
 Possède-Tab-N (Joueur-x, Nbre) : le joueur Joueur-x possède un nombre Nbre de tableaux.
 Possède-Titre (Joueur-x, Titre-y) : le joueur Joueur-x possède le titre Titre-y.
 Possède-Titre-N (Joueur-x, Nbre, Id-case) : le joueur Joueur-x possède un nombre Nbre de titres de la case Id-case.
 Provoqué-début (Oui/Non) : l'usager du jeu a/n'a pas provoqué le début du jeu.
 Reçu (Joueur-x, Somme) : le joueur Joueur-x a reçu la somme d'argent Somme provenant du fonds boursier.
 Reçu-Courtage (Joueur-x, Argent) : le joueur Joueur-x a reçu la somme d'argent Argent de la case courtage.
 Réponse-Joueur (Joueur-x, Oui/Non) : l'usager du joueur Joueur-x a répondu Oui/Non à la proposition du GA.
 Réponse-Joueur-Offre (Joueur-x, Oui/Non) : l'offre du joueur a été acceptée/refusée par le joueur Joueur-x (si Joueur-x = 0, alors refusée par tous les joueurs).
 Revenu-Payé (Oui/Non) : le joueur courant a/n'a pas payé le revenu.
 Somme-Dés (k) : la somme des dés (=i+j) vaut k.
 Type-Has (Carte-has-y, Type-carte) : la carte de hasard Carte-has-y est de type Type-carte).
 Vendu-Act (Joueur-x, Oui/Non) : le joueur Joueur-x a/n'a pas vendu d'actions.
 Vendu-Tab (Joueur-x, Oui/Non) : le joueur Joueur-x a/n'a pas vendu de tableaux.

2. Spécification par prédicats des états communs aux acteurs.

Le joueur courant a lancé les dés :

Joueur-Désigné-GA (joueur-x)
 ET Etat-Joueur (joueur-x, courant)
 ET Etat-Dés (i, j)
 ET Somme-Dés (k)
 ET Nbre-Avancé (0)

Joueur sur case-prêt normale :

Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-prêt, p)
 ET Etat-Case (p, normale)

Joueur sur case-départ normale :

Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-départ, p)
 ET Etat-case (p, normale)

Joueur a avancé d'une case :
 Nbre-Avancé (na+1)
 ET Joueur-A-Avancé (oui)
 Le prêt a changé d'état s'il existe :
 Etat-Joueur (joueur-x, courant)
 ET { [Possède-Prêt-N (joueur-x, id-case, 1)
 ET Etat-Prêt (prêt-y, id-case, joueur-x, autorisation de réclamer)]
 OU
 [Possède-Prêt-N (joueur-x, id-case, 0)] }
 Le joueur a reçu argent s'il possède actions :
 Etat-Joueur (joueur-x, courant)
 ET Déposé (GA, 100000)
 ET { [Reçu (joueur-x, $0.05 \times \text{nbre-act} \times \text{mf}$)
 ET Possède-Act-N (joueur-x, nbre-act) avec $1 \leq \text{nbre-act} \leq 16$]
 OU
 [Possède-Act-N (joueur-x, 0)] }
 Pas avancé du nbre de cases voulu :
 Avancé-Nbre-Voulu (non)
 Joueur a avancé du nbre de cases voulu :
 Avancé-Nbre-Voulu (oui)
 ET Etat-case (p, déclenchée)
 Joueur sur case-hasard déclenchée :
 Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-hasard, p)
 ET Etat-Case (p, déclenchée)
 Le joueur a reçu carte-hasard :
 Etat-Joueur (joueur-x, courant)
 ET Possède-Has (joueur-x, carte-has-y)
 Type de la carte-has = "catastrophe" :
 Type-Has (carte-has-y, catastrophe)
 ET Etat-Has (carte-has-y, non payée)
 Type de la carte-has = "privilège" :
 Type-Has (carte-has-y, privilège)
 ET Etat-Has (carte-has-y, non profitée)
 Joueur en faillite et état-has = "non payée" :
 Etat-Joueur (joueur-x, en faillite)
 ET Joueur-Désigné-GA (joueur-x)
 Etat-joueur = "courant" et état-has = "payée" :
 Etat-Joueur (joueur-x, courant)
 ET Etat-Has (carte-has-y, payée)
 ET Num-Joueur-Has (carte-has-y, 0)
 Etat-has = "profitée" :
 Etat-Has (carte-has-y, profitée)
 ET Num-Joueur-Has (carte-has-y, 0)
 Joueur sur case-courtage déclenchée :
 Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-courtage, p)
 ET Etat-Case (p, déclenchée)
 Le joueur a empoché la somme courtage :
 Etat-Joueur (joueur-x, courant)
 ET Reçu-Courtage (joueur-x, argent-courtage)
 Joueur sur case-prêt déclenchée :
 Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-prêt, p)
 ET Etat-Case (p, déclenchée)
 Le joueur a évt prêté de l'argent au GA sur cette case :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET { [Possède-Prêt (joueur-x, prêt-y, id-case, montant, interdiction de réclamer)]

ET Possède-Prêt-N (joueur-x, id-case, 1)]
 OU
 [Possède-Prêt-N (joueur-x, id-case, 0)] }
 Le joueur possède des tableaux :
 Etat-Joueur (joueur-x, courant)
 ET Possède-Tab-N (joueur-x, nbre) avec 1 ≤ nbre ≤ 5
 Joueur sur case-art déclenchée :
 Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-art, p)
 ET Etat-Case (p, déclenchée)
 Le joueur a évt vendu des tableaux :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET { [Vendu-Tab (joueur-x, oui)
 ET Possède-Tab-N (joueur-x, n1) avec 0 ≤ n1 ≤ nbre]
 OU
 [Vendu-Tab (joueur-x, non)] }
 Le joueur a évt acheté un tableau :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET { [Possède-Tab (joueur-x, tab-a)
 ET Etat-Tab (tab-a, vendu)
 ET Acheté-Tab (joueur-x, oui)]
 OU
 [Acheté-Tab (joueur-x, non)] }
 Joueur sur case-or déclenchée :
 Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-or, p)
 ET Etat-Case (p, déclenchée)
 Le joueur a évt acheté un lingot d'or :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET { [Possède-Lingot (joueur-x, lingot-a)
 ET Etat-Lingot (lingot-a, vendu)]
 OU
 [Etat-Lingot (lingot-a, invendu)] }
 Joueur sur case-bourse déclenchée :
 Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-bourse, p)
 ET Etat-Case (p, déclenchée)
 Le joueur possède actions :
 Etat-Joueur (joueur-x, courant)
 ET Possède-Act-N (joueur-x, nbre) avec 1 ≤ nbre ≤ 16
 Joueur a évt vendu des actions :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET { [Vendu-Act (joueur-x, oui)
 ET Possède-Act-N (joueur-x, n1) avec 0 ≤ n1 ≤ nbre]
 OU
 [Vendu-Act (joueur-x, non)] }
 Le joueur a évt acheté des actions :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET { [Acheté-Act (joueur-x, n1) avec 1 ≤ n1 ≤ 16 - nbre-act-joueur]
 OU
 [Acheté-Act (joueur-x, 0)] }
 Joueur sur case-propriété déclenchée :
 Etat-Joueur (joueur-x, courant)
 ET Position (joueur-x, case-propriété, p)
 ET Etat-Case (p, déclenchée)

Etat-propriété = "semi-libre" :
 Etat-Propriété (p, semi-libre)
 Etat-propriété = "occupée" :
 Etat-Propriété (p, occupée)
 Aucune partie de la propriété n'appartient au joueur :
 Etat-Joueur (joueur-x, courant)
 ET Possède-Titre-N (joueur-x, 0, p)
 Revenu payé :
 Revenu-Payé (oui)
 ET Etat-GA (normal)
 Joueur en faillite :
 Revenu-Payé (non)
 ET Etat-Joueur (joueur-x, en faillite)
 ET Joueur-Désigné-GA (joueur-x)
 ET Etat-GA (normal)
 Etat-propriété = "libre" :
 Etat-Propriété (p, libre)
 Le joueur courant a acheté le titre :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET Possède-Titre (joueur-x, titre-y)
 ET Etat-Titre (titre-y, vendue)
 Le joueur courant n'a pas acheté le titre :
 Etat-Joueur (joueur-x, courant)
 ET Etat-GA (normal)
 ET Etat-Titre (titre-y, invendue)
 ET Etat-Enchères (autorisée)
 Le titre est vendu à un joueur quelconque :
 Etat-Joueur (joueur-x, courant)
 ET Etat-Joueur (joueur-z, en attente)
 ET Etat-Enchères (terminée)
 ET Possède-Titre (joueur-z, titre-y)
 ET Etat-Titre (titre-y, vendue)
 ET Etat-GA (normal)
 Le titre n'est pas vendu :
 Etat-Joueur (joueur-x, courant)
 ET Etat-Enchères (terminée)
 ET Etat-Titre (titre-y, invendue)
 ET Etat-GA (normal)
 Les joueurs en faillite sont éliminés :
 Joueurs-Eliminés (oui)

3. Spécification par prédicats des états propres aux acteurs.

a. Le meneur de jeu :

Usager a provoqué début du jeu :
 Provoqué-Début (oui)
 Jeu ouvert, joueurs prêts à jouer :
 Etat-Jeu (ouvert)
 ET Etat-Joueur (joueur-x, prêt à jouer) avec $2 \leq x \leq \text{Nbre-joueur}$
 ET Etat-GA (en proposition)
 Etat-GA = "normal" :
 Etat-GA (normal)
 Joueurs en attente :
 Etat-GA (normal)
 ET { [Etat-Joueur (joueur-x, en attente)
 ET Possède-PA (joueur-x, PA-y)]
 OU
 [Etat-Joueur (joueur-x, en attente)] } avec $2 \leq x \leq \text{Nbre-joueur}$

Les joueurs ont fini de négocier :

Etat-Joueur (joueur-x, plus d'offre) avec $2 \leq x \leq \text{Nbre-joueur}$

Les joueurs ont fini de manipuler leurs séries :

Etat-Joueur (joueur-x, plus de manipulation)
avec $2 \leq x \leq \text{Nbre-joueur}$

Il reste moins de 2 joueurs et le jeu est fermé :

[Nbre-Jeu (0) ET Etat-Jeu (fermé)]

OU [Nbre-Jeu (1) ET Etat-Jeu (fermé)]

Il reste plus de un joueur :

Nbre-Jeu (x) avec $2 \leq x \leq 6$

b. Le joueur :

MJ sait nbre joueur et état-joueur = "créé" :

Etat-Jeu (ouvert)

ET Etat-Joueur (joueur-x, créé) avec $2 \leq x \leq \text{Nbre-joueur}$

Etat-joueur = "prêt à jouer" :

Etat-Joueur (joueur-x, prêt à jouer) avec $2 \leq x \leq \text{Nbre-joueur}$

MJ a proposé d'acheter PA :

Etat-GA (en proposition)

L'usager a répondu à l'offre de PA :

Réponse-Joueur (joueur-x, oui)

OU Réponse-Joueur (joueur-x, non) avec $2 \leq x \leq \text{Nbre-joueur}$

Le joueur a évt acheté une PA :

Etat-GA (normal)

ET Joueur-Désigné-GA (joueur-x)

ET { [Etat-Joueur (joueur-x, en attente)

ET Possède-PA (joueur-x, PA-y)]

OU

[Etat-Joueur (joueur-x, en attente)] }

MJ a désigné ce joueur courant :

Joueur-Désigné-GA (joueur-x)

Déclenchement du prêt effectué :

Déclenchement-Prêt (oui)

Déclenchement départ effectué :

Déclenchement-départ (oui)

Usager a répondu à l'offre :

Etat-Joueur (joueur-x, courant)

ET [Réponse-Joueur (joueur-x, oui)

OU

Réponse-Joueur (joueur-x, non)]

MJ a proposé au joueur de faire un prêt :

Etat-GA (en proposition)

MJ a proposé au joueur de vendre des tableaux et usager a répondu :

Etat-GA (en proposition)

ET Etat-Joueur (joueur-x, courant)

ET [Réponse-Joueur (joueur-x, oui)

OU

Réponse-Joueur (joueur-x, non)]

MJ a proposé au joueur d'acheter un tableau et usager a répondu :

Etat-GA (en proposition)

ET Etat-Joueur (joueur-x, courant)

ET [Réponse-Joueur (joueur-x, oui)

OU

Réponse-Joueur (joueur-x, non)]

MJ a proposé au joueur d'acheter un lingot et usager a répondu :

Etat-GA (en proposition)

ET Etat-Joueur (joueur-x, courant)

ET [Réponse-Joueur (joueur-x, oui)

OU

Réponse-Joueur (joueur-x, non)]

MJ a proposé au joueur de vendre des actions et usager a répondu :

Etat-GA (en proposition)

ET Etat-Joueur (joueur-x, courant)

ET [Réponse-Joueur (joueur-x, oui)

OU

Réponse-Joueur (joueur-x, non)]

MJ a proposé au joueur d'acheter des actions et usager a répondu :

Etat-GA (en proposition)

ET Etat-Joueur (joueur-x, courant)

ET [Réponse-Joueur (joueur-x, oui)

OU

Réponse-Joueur (joueur-x, non)]

Le MJ a ordonné de payer :

Etat-GA (en proposition)

MJ a proposé au joueur d'acheter un titre et usager a répondu à l'offre :

Etat-GA (en proposition)

ET Etat-Joueur (joueur-x, courant)

ET [Réponse-Joueur (joueur-x, oui)

OU

Réponse-Joueur (joueur-x, non)]

MJ a déclenché vente aux enchères :

Etat-GA (aux enchères)

ET Etat-Enchères (en cours)

MJ a proposé aux joueurs de négocier :

Etat-GA (en négociation)

ET Etat-Joueur (joueur-x, en négociation) 2sx≤Nombre-joueur

ET Etat-Négociation (pas commencée)

Offre du joueur effectuée :

Offre-Effectuée (oui)

ET [Etat-Joueur (joueur-x, offre de vente)

OU

Etat-Joueur (joueur-x, offre d'achat)

OU

Etat-Joueur (joueur-x, offre d'échange)]

ET Joueur-Désigné-GA (joueur-x)

ET Etat-Négociation (en cours)

Transaction refusée :

Etat-Joueur (joueur-x, en négociation)

ET Réponse-Joueur-Offre (O, non)

Transaction effectuée :

Etat-Joueur (joueur-x, en négociation)

ET Réponse-Joueur-Offre (joueur-y, oui)

Plus d'offre :

Etat-Joueur (joueur-x, plus d'offre)

ET Joueur-Désigné-GA (joueur-x)

MJ a proposé au joueur de manipuler séries :

Etat-GA (en proposition)

ET Etat-Joueur (joueur-x, en manipulation)

ET Joueur-Désigné-GA (joueur-x)

Manipulations terminées :

Etat-Joueur (joueur-x, plus de manipulation)

c. La case bourse :

Le joueur ne possède pas d'actions :

Etat-Joueur (joueur-x, courant)

ET Possède-Act-N (joueur-x, 0)

d. La case art :

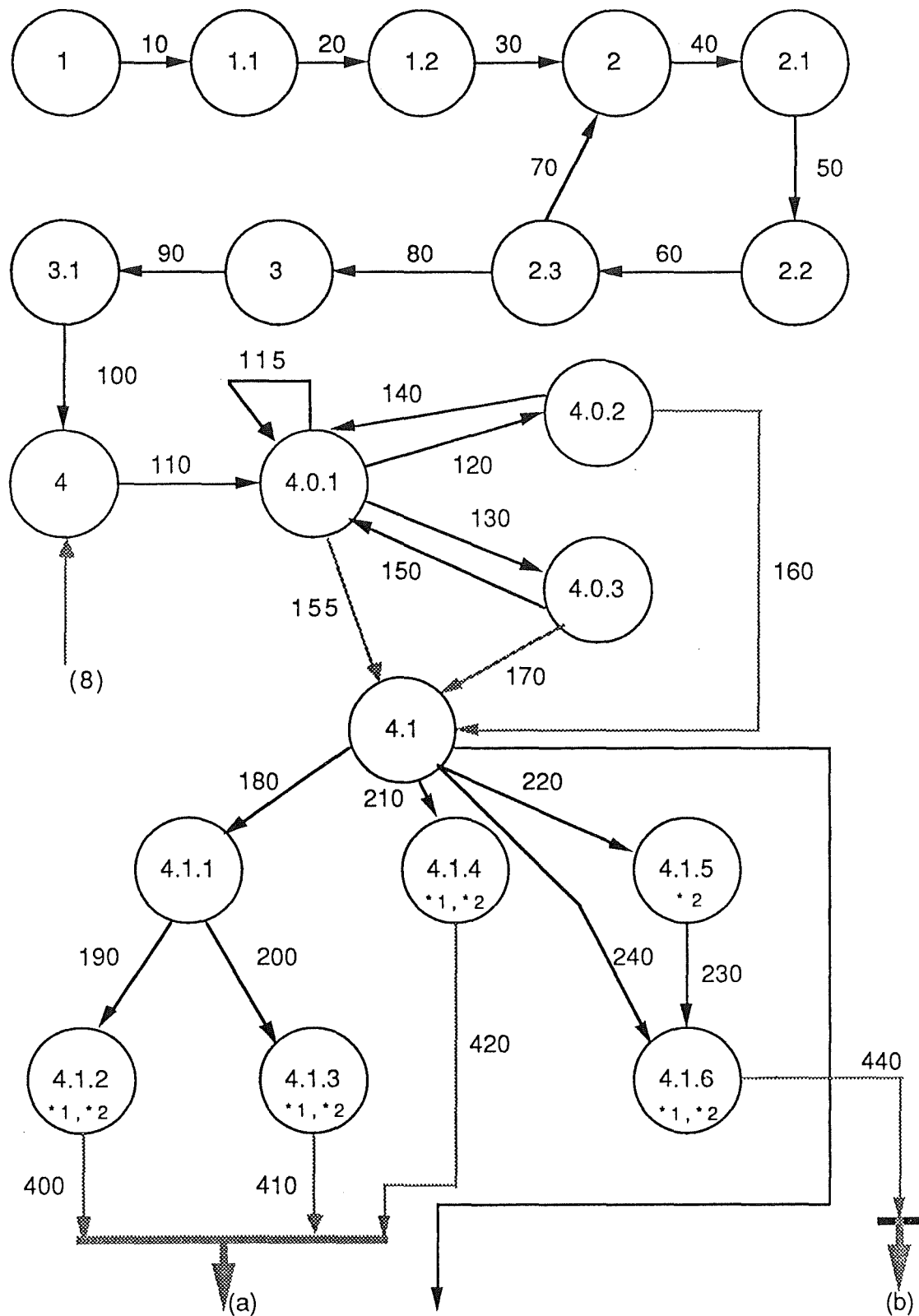
Le joueur ne possède pas de tableaux :

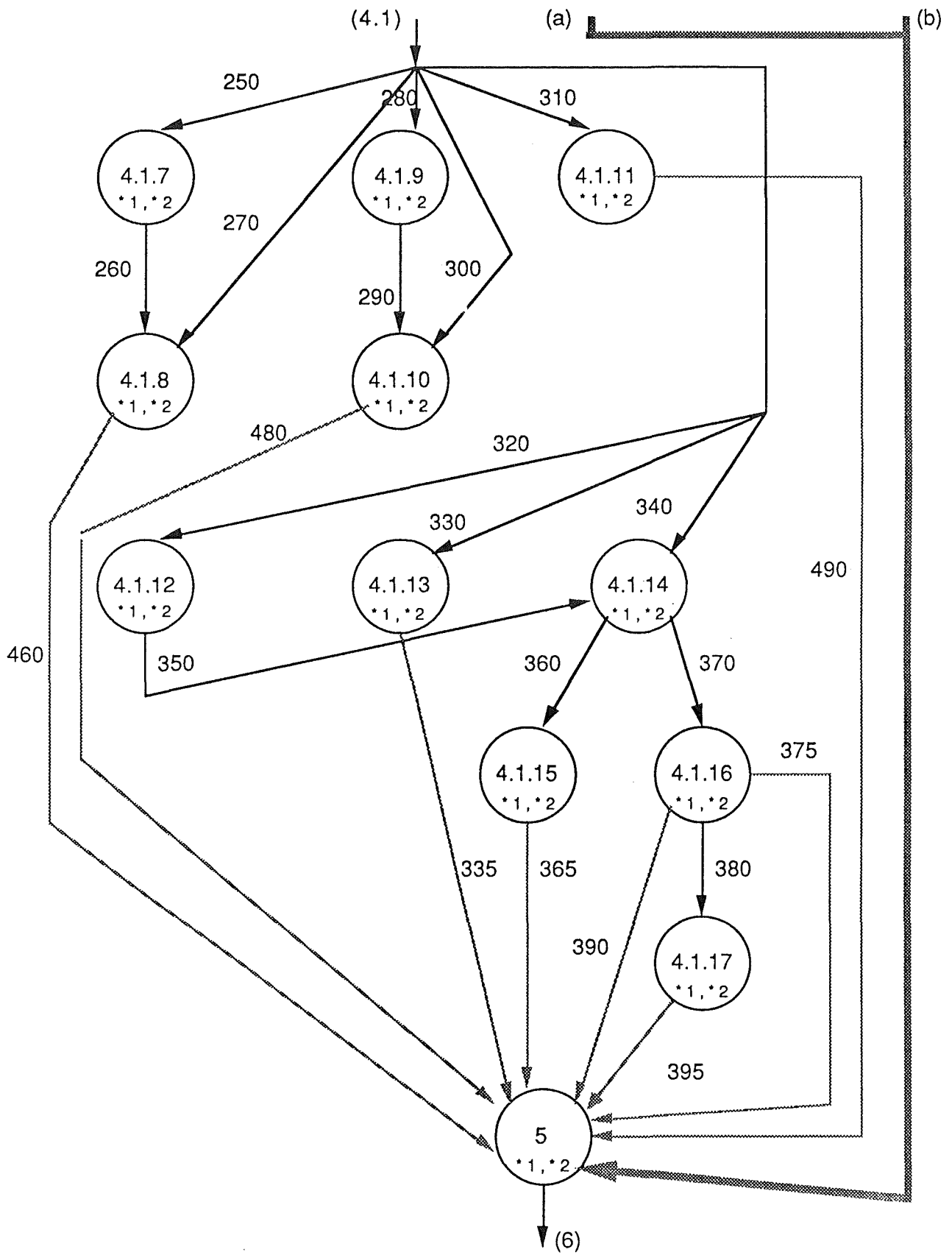
Etat-Joueur (joueur-x, courant)

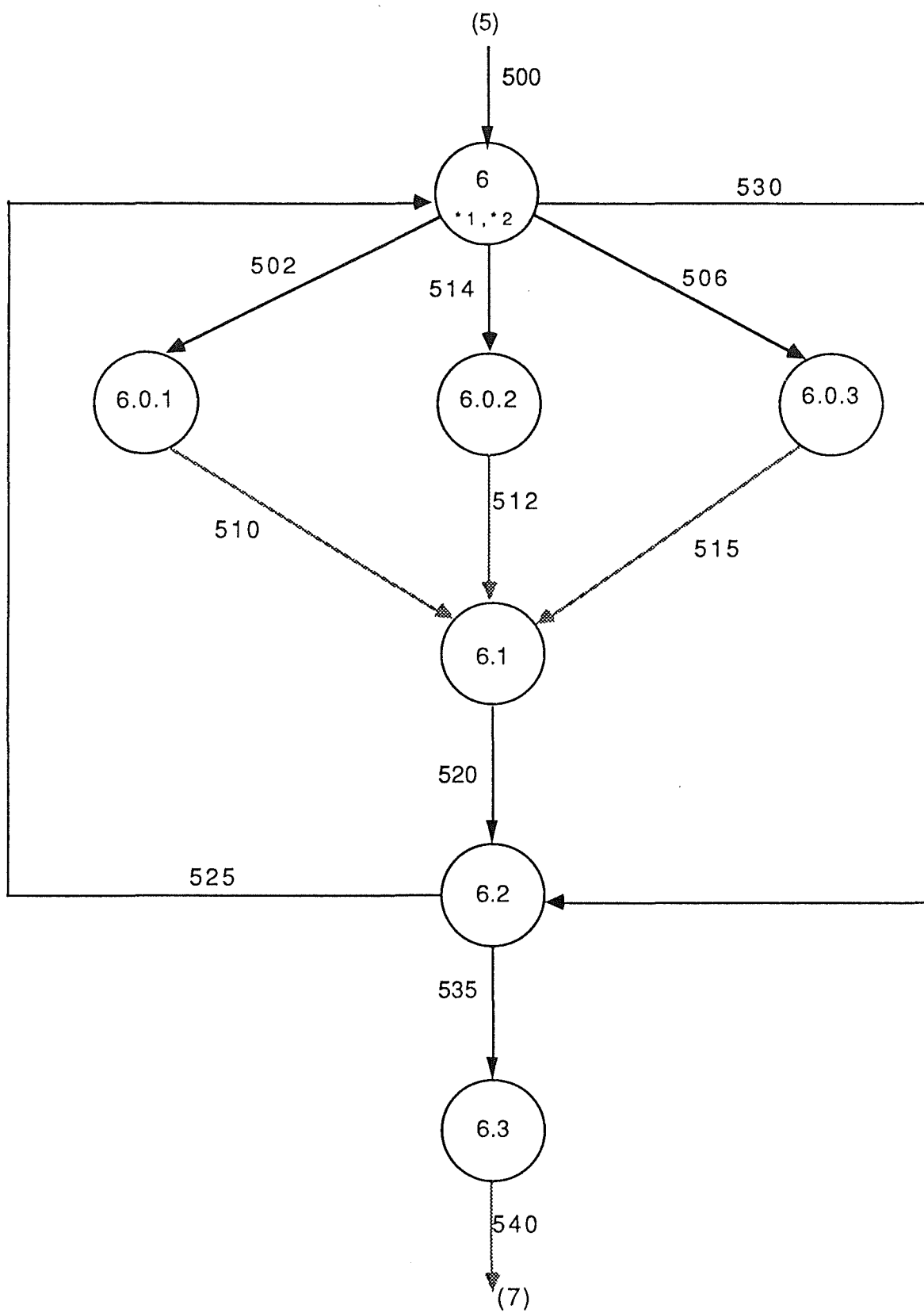
ET Possède-Tab-N (joueur-x, 0)

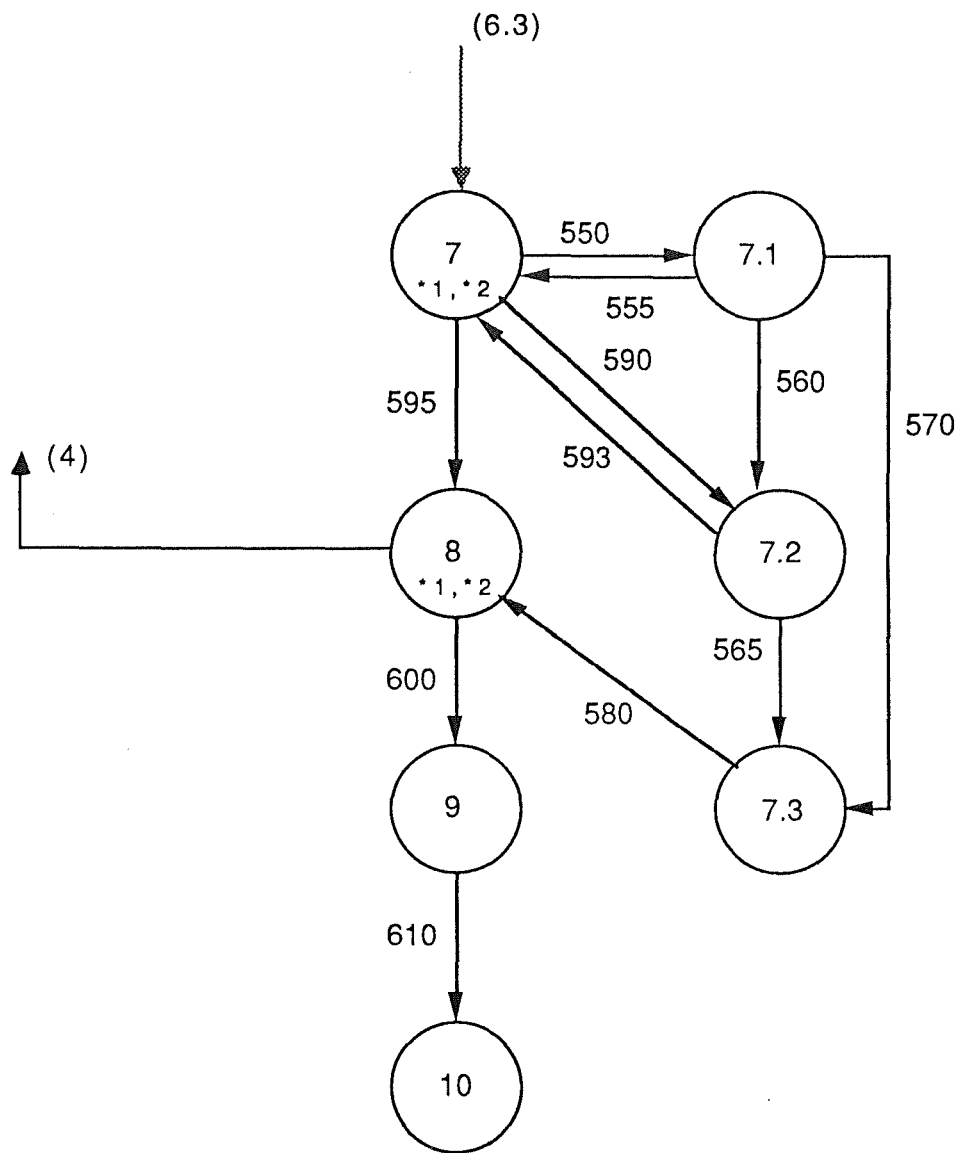
Remarque : Les autres acteurs n'ont pas d'état qui leur soit propre.

Réseau de transitions entre les jalons du scénario

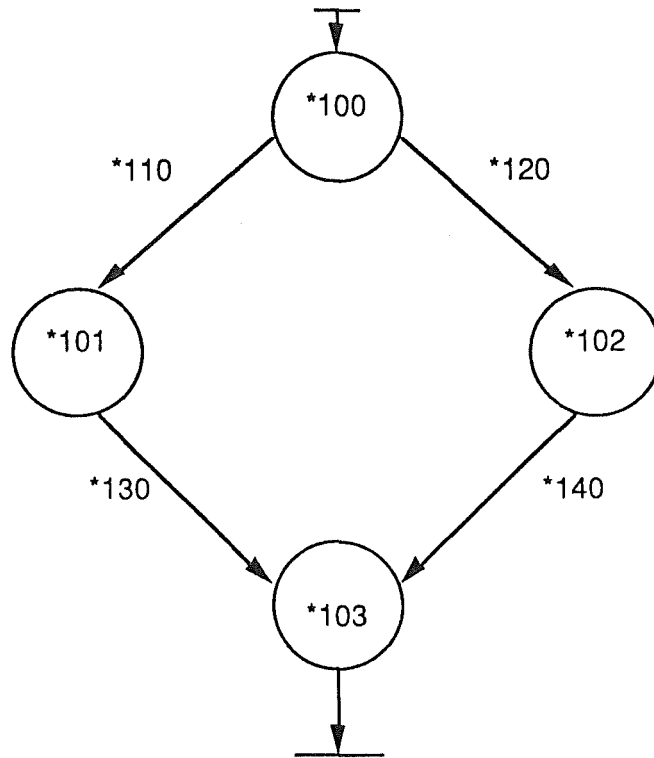




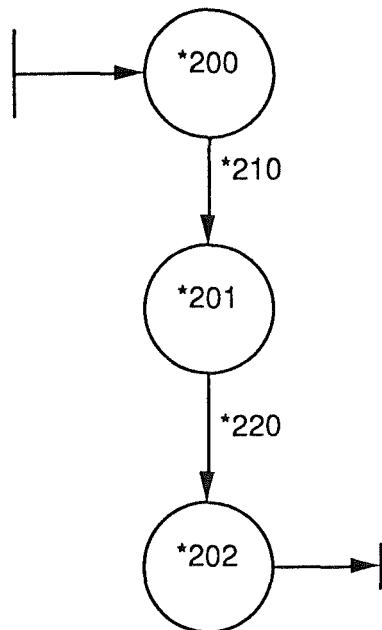




*1 : Remboursement d'un prêt.

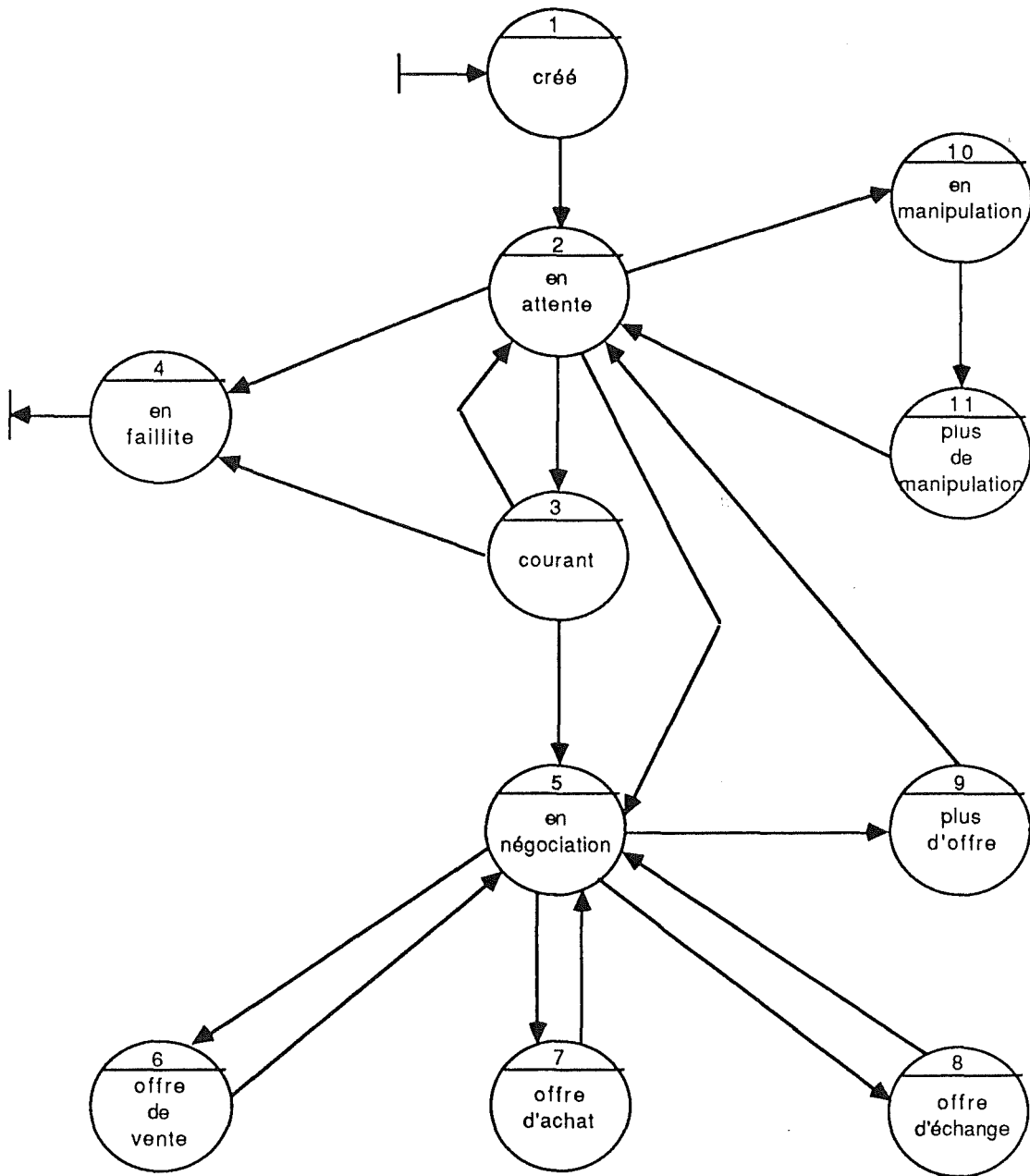


*2 : Vente de lingot d'or.

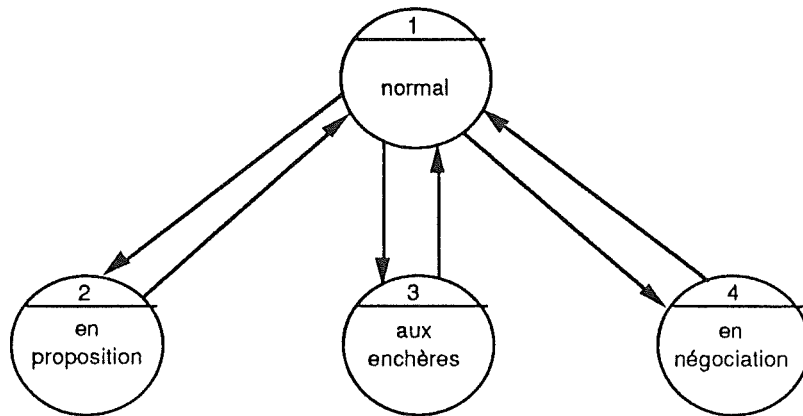


Cycles de vie des différents acteurs

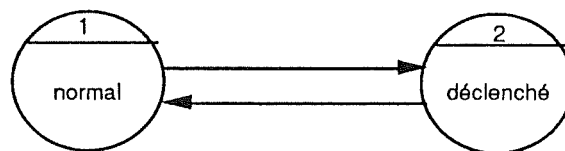
a) Joueur



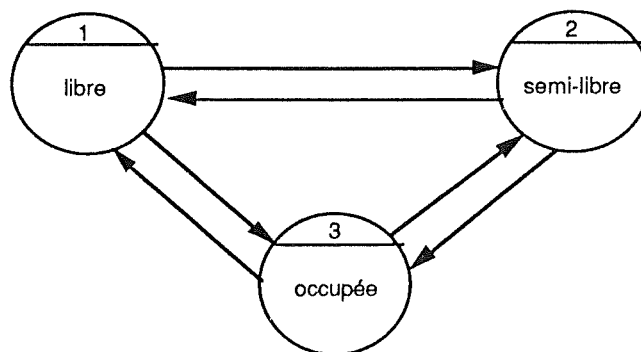
b) GA



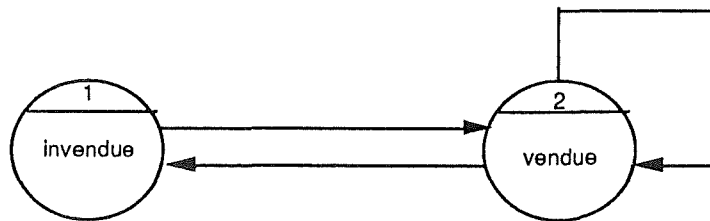
c) Case



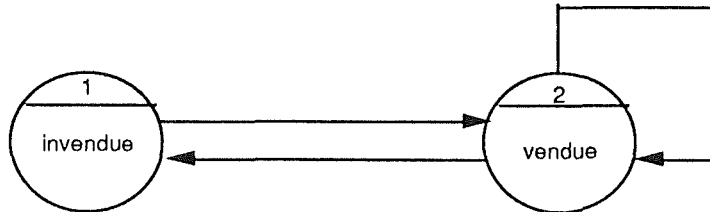
d) Case-propriété



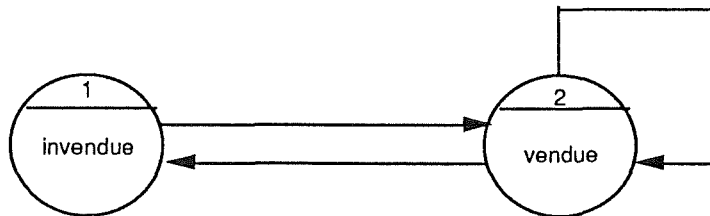
e) Carte-propriété



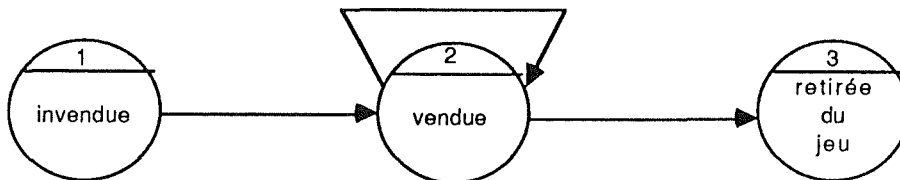
f) Carte-banque



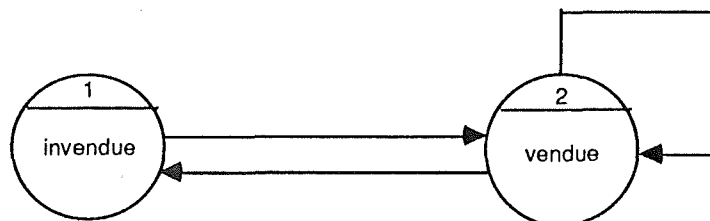
g) Action



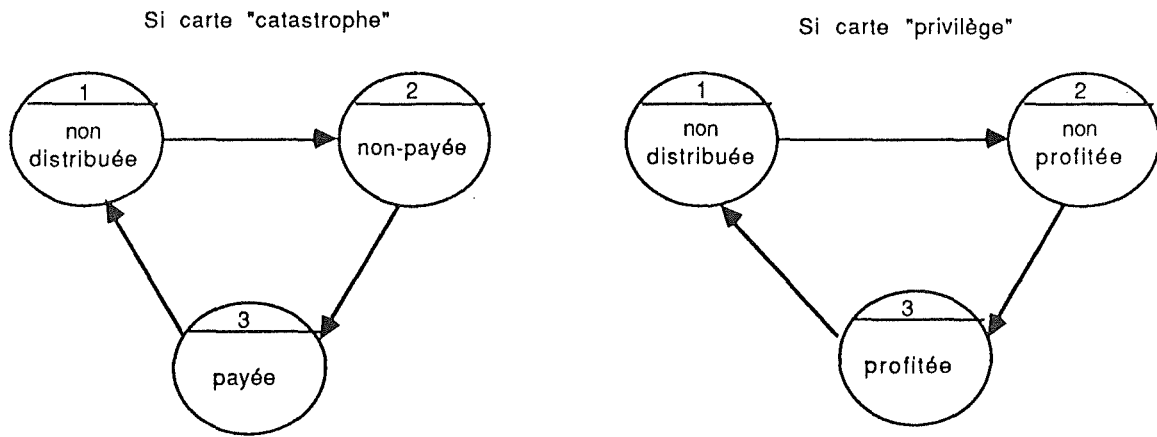
h) Tableau



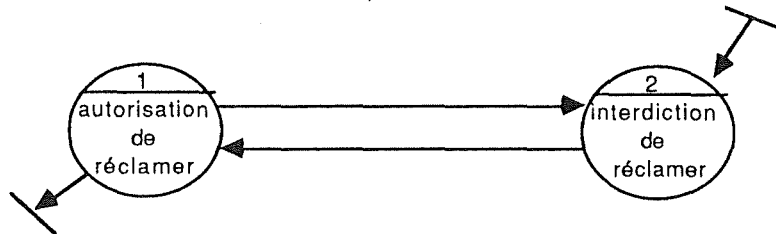
i) PA



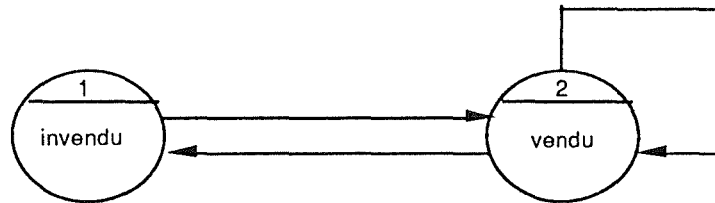
j) Hasard



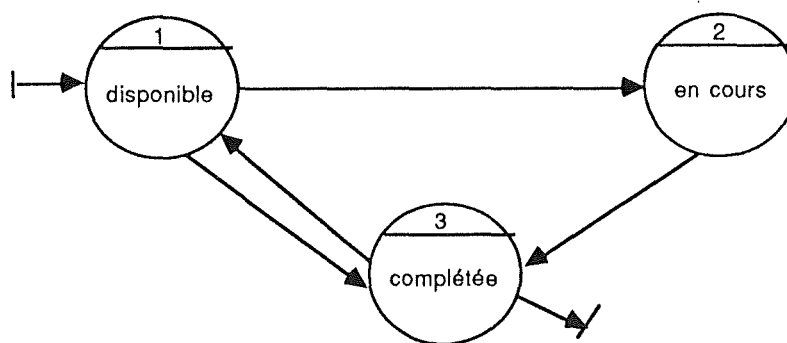
k) Prêt



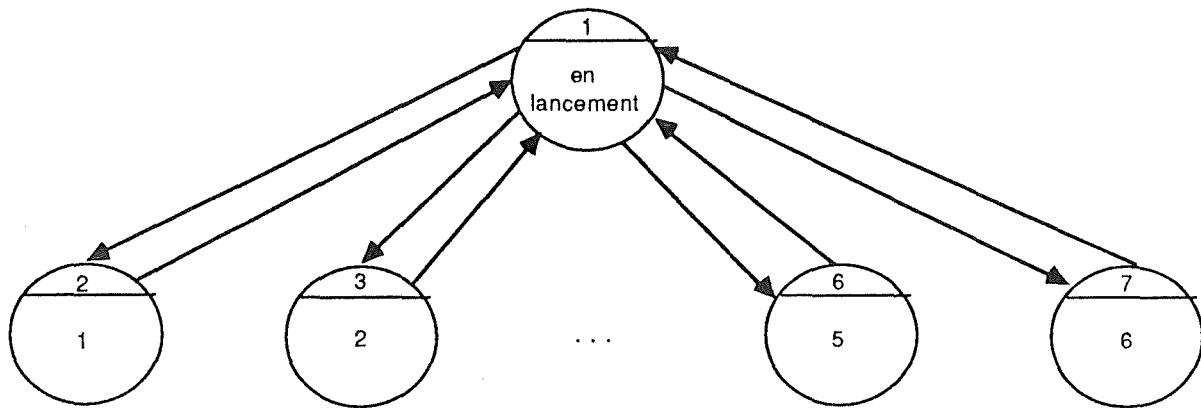
l) lingot



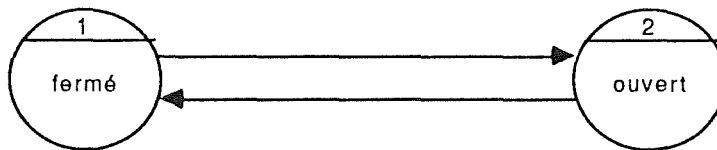
m) Série



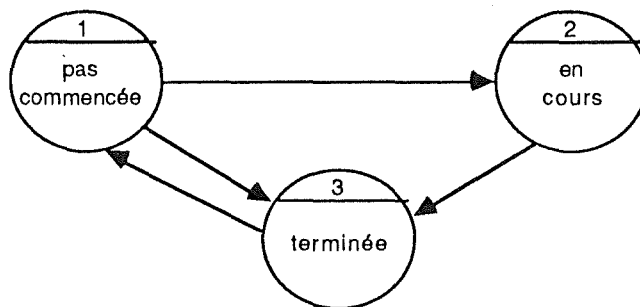
n) Dé



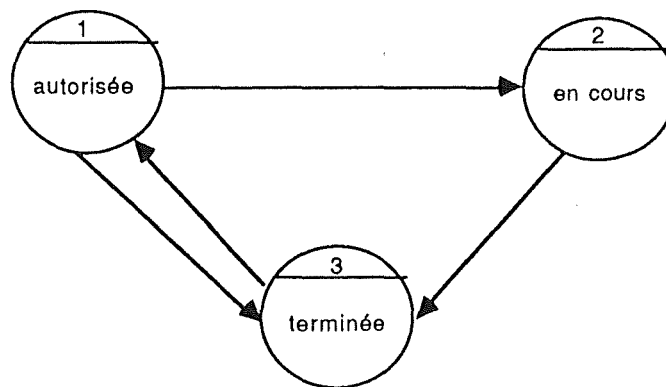
o) Jeu



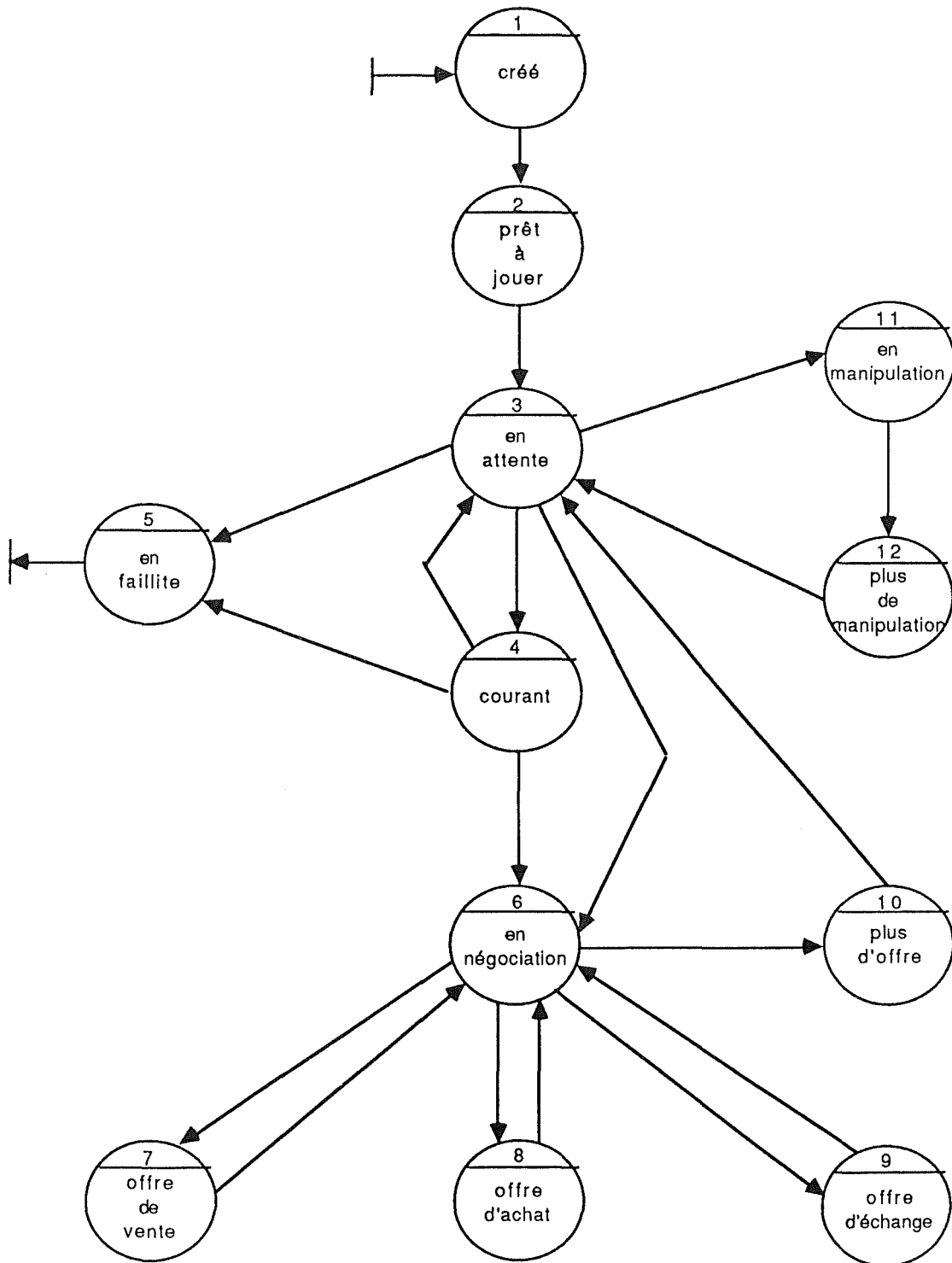
p) négociation



q) Enchères



Nouveau cycle de vie du Joueur.



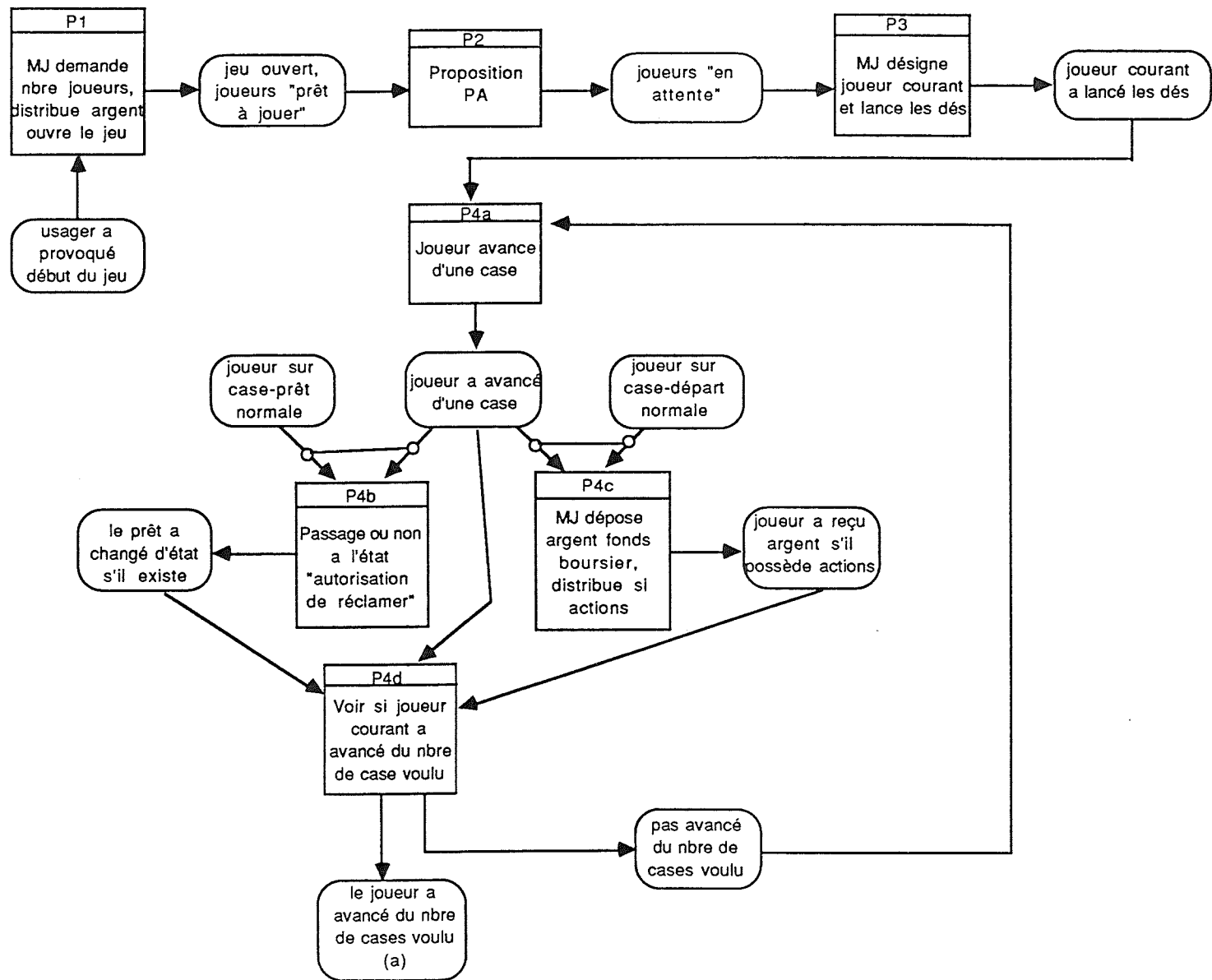


Diagramme de transition du MENEUR DE JEU (1)

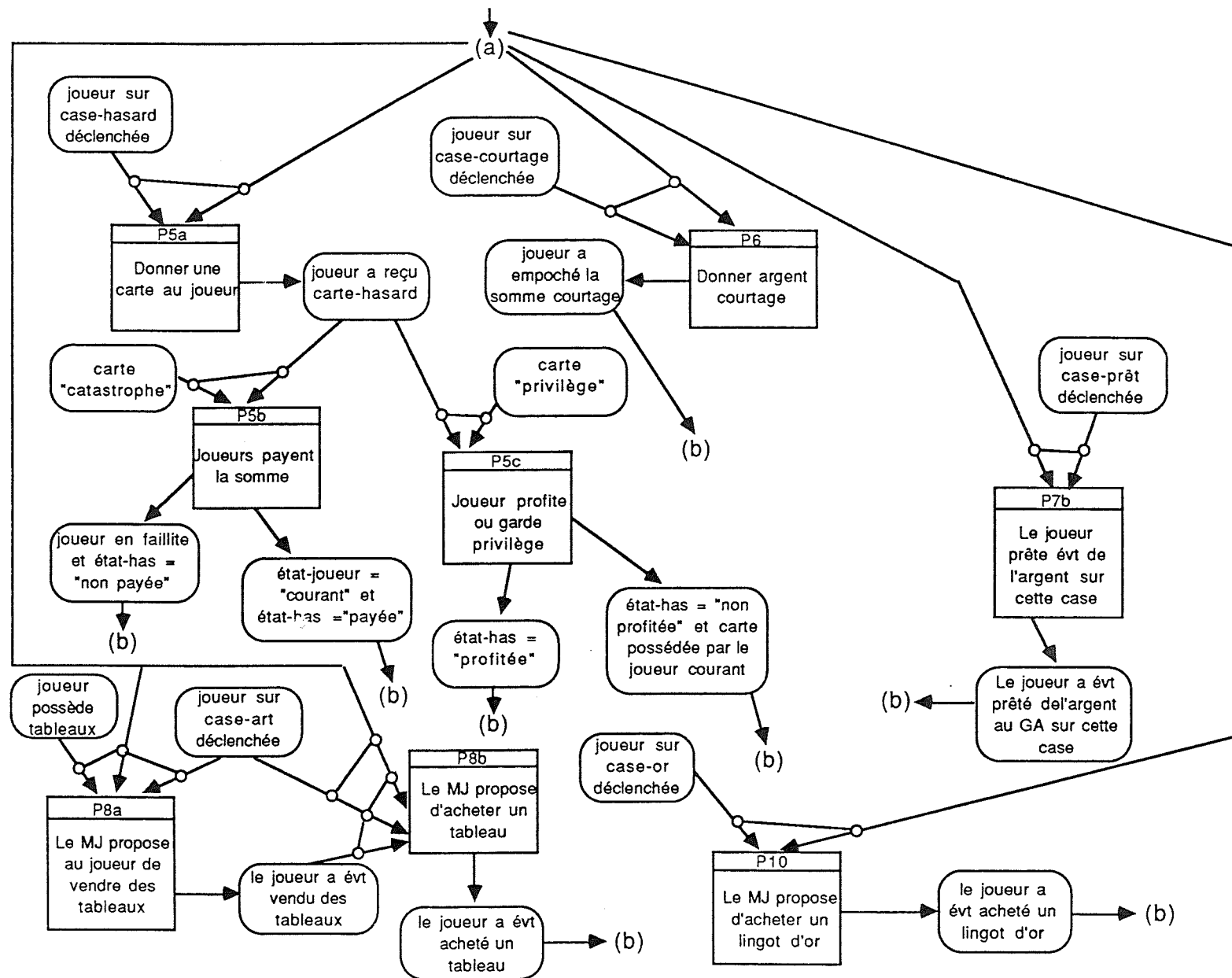


Diagramme de transition du MENEUR DE JEU (2)

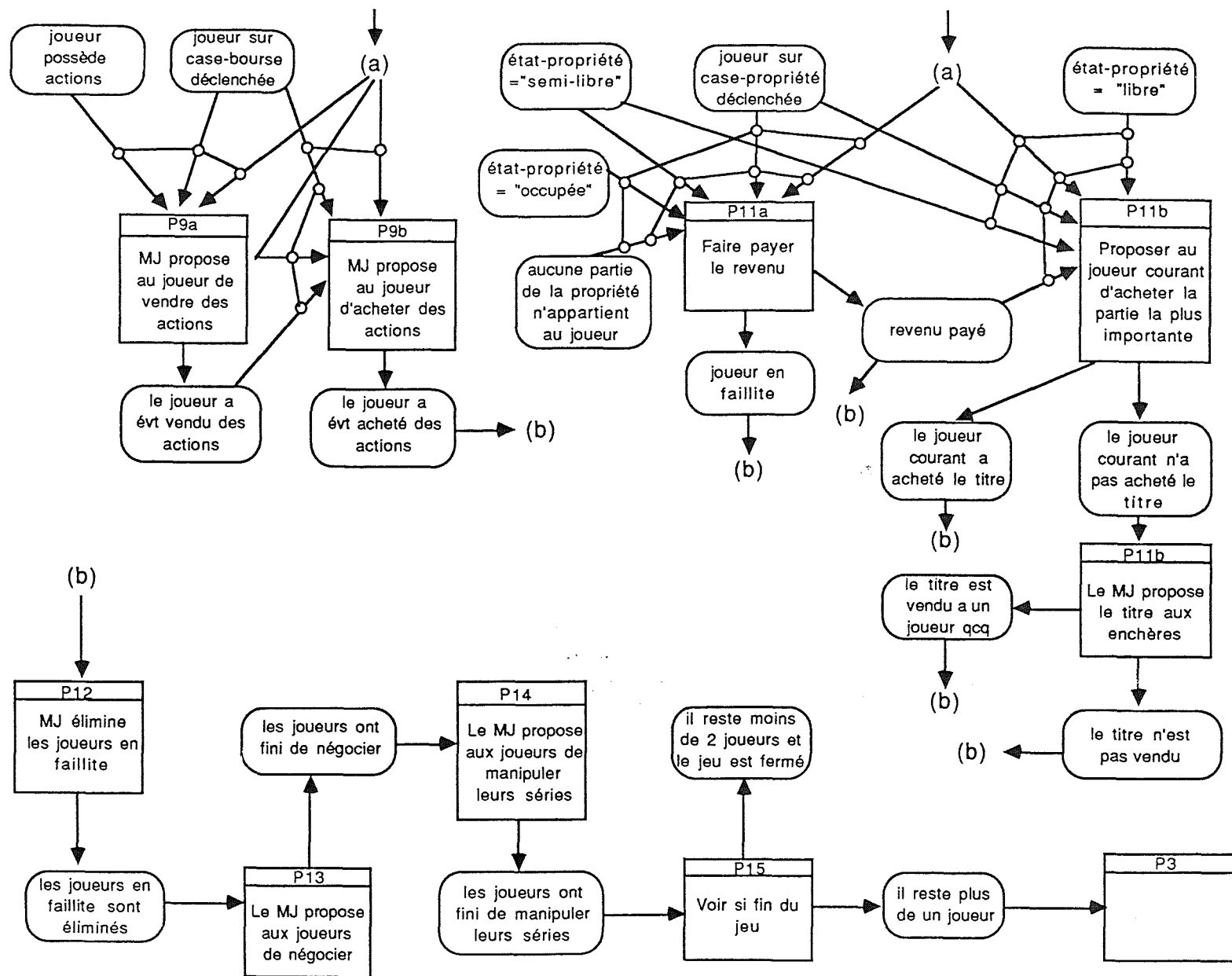
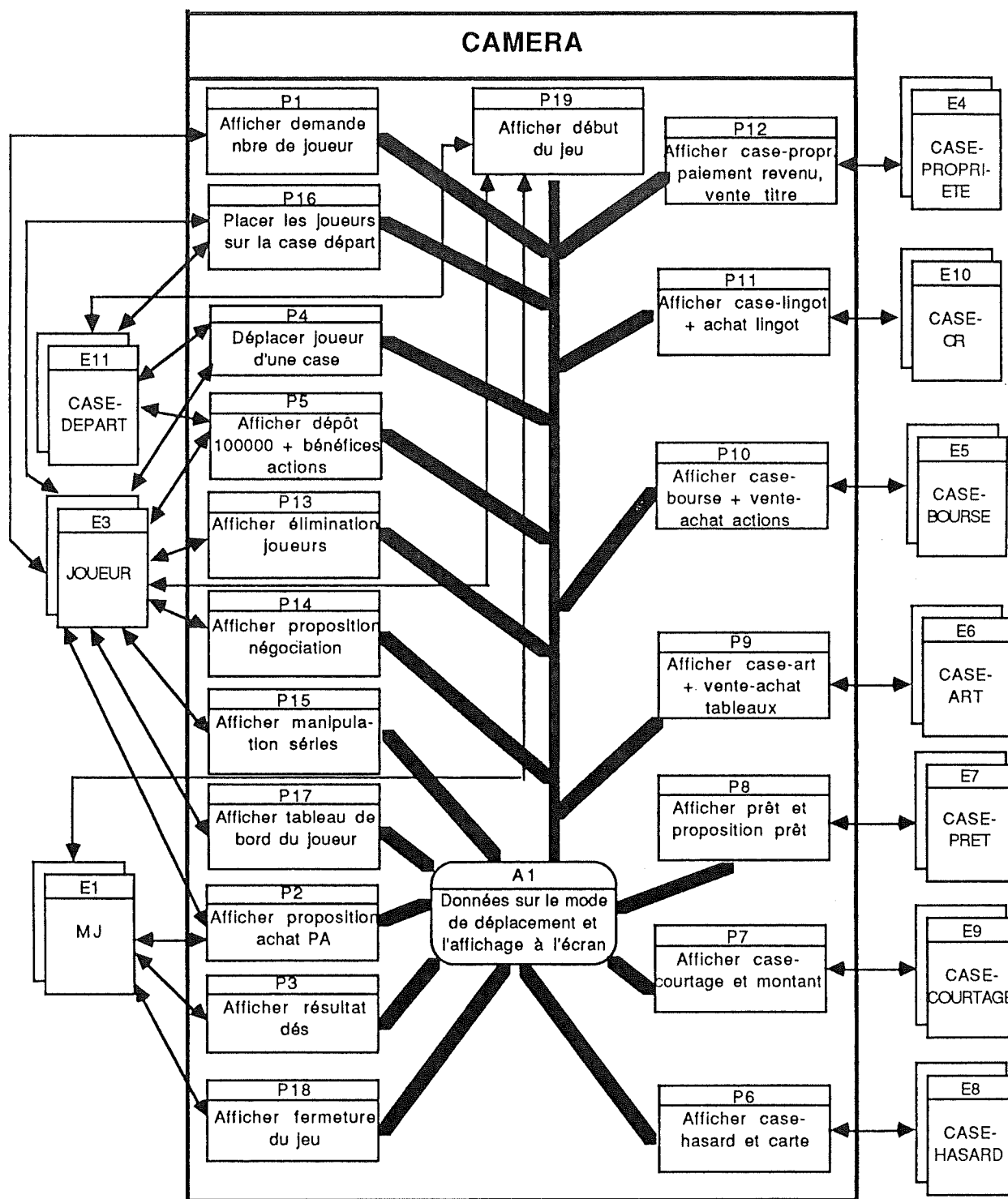


Diagramme de transition du MENEUR DE JEU (3)



Plan de système de la CAMERA

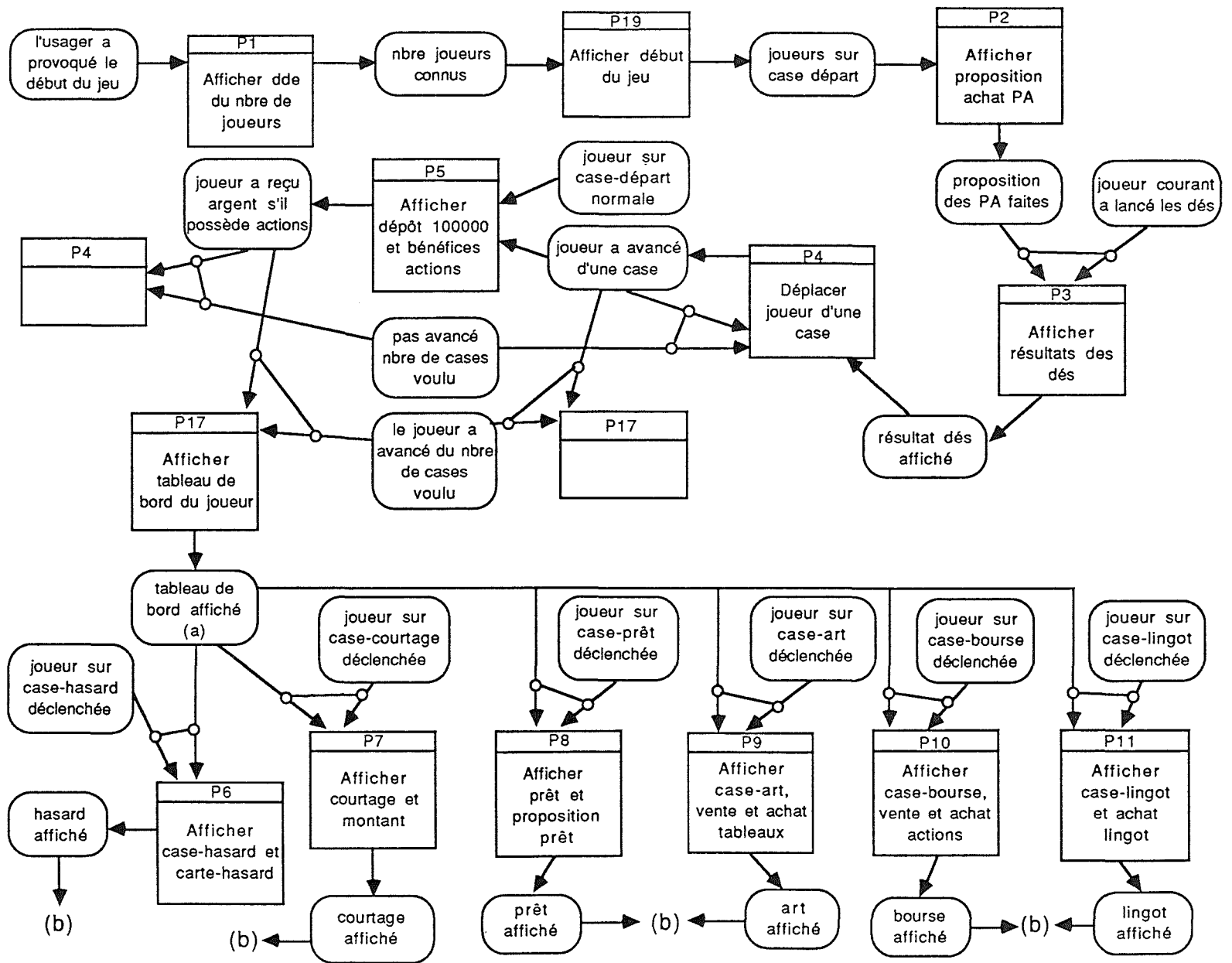


Diagramme de transition de la CAMERA (1)

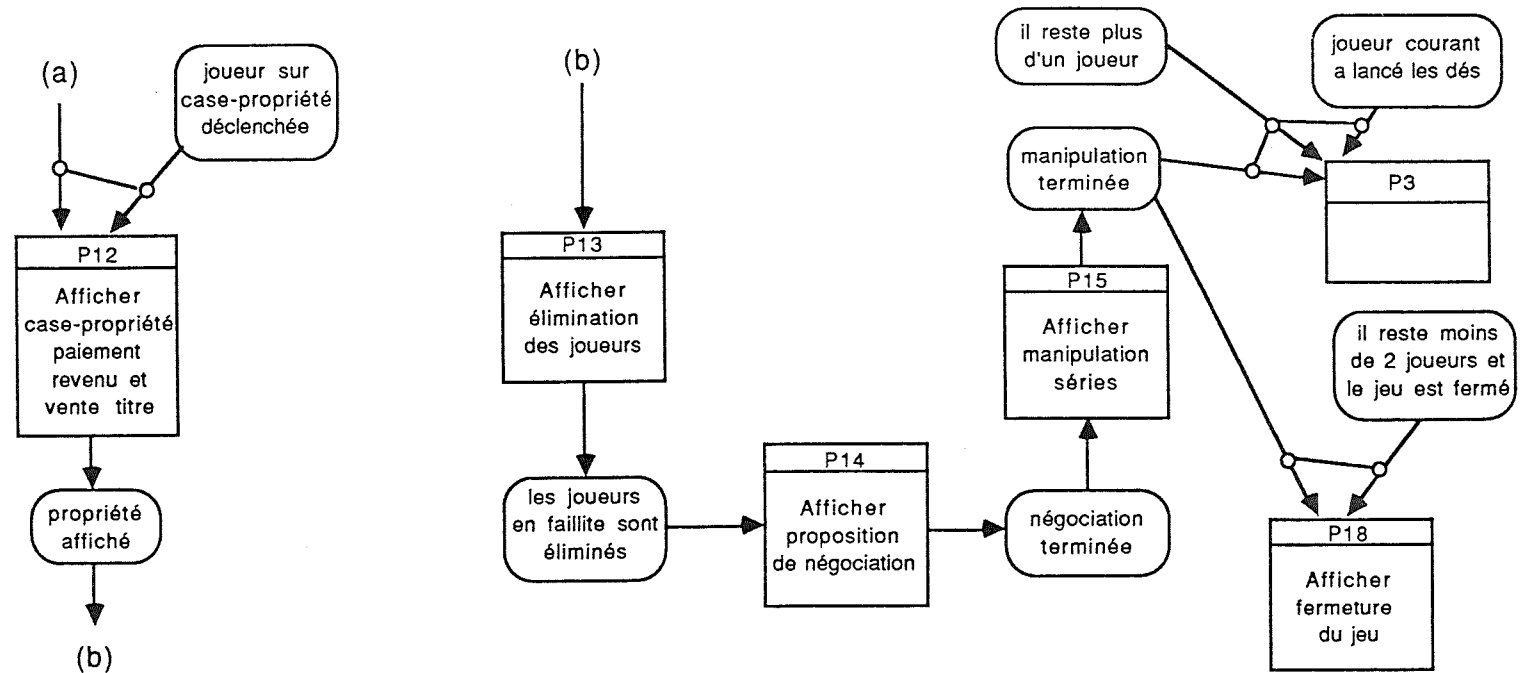
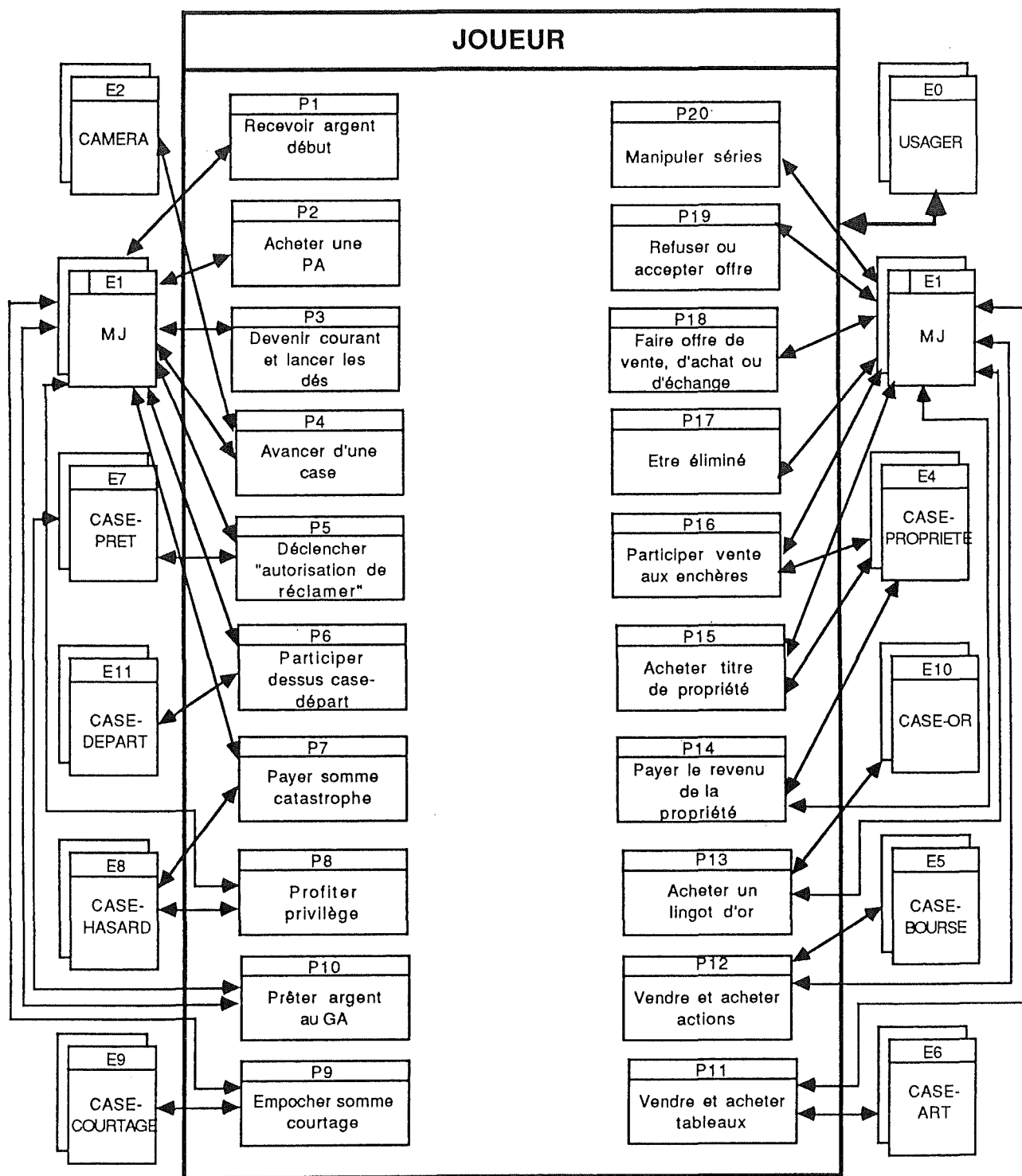
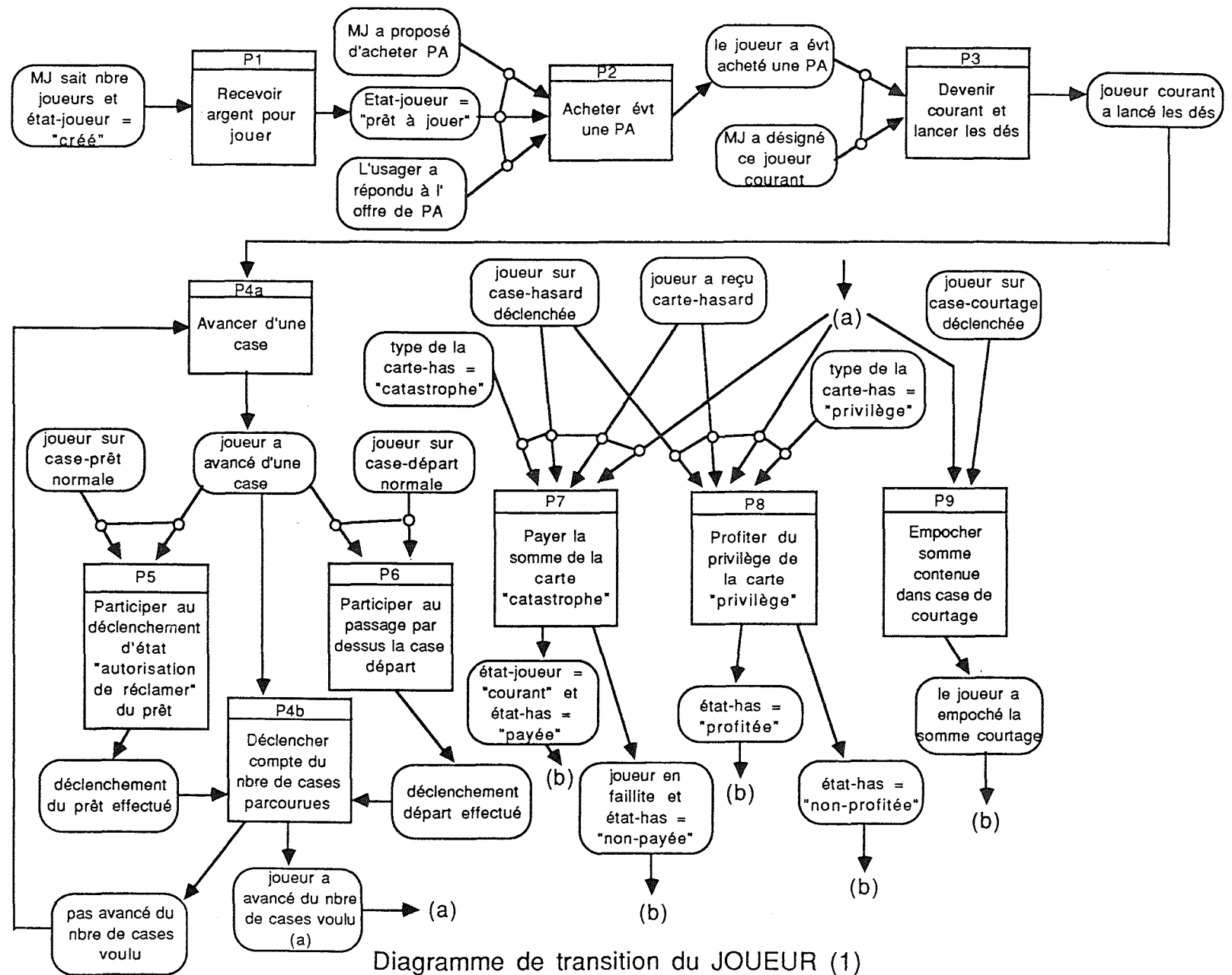


Diagramme de transition de la CAMERA (2)



Plan de système du JOUEUR



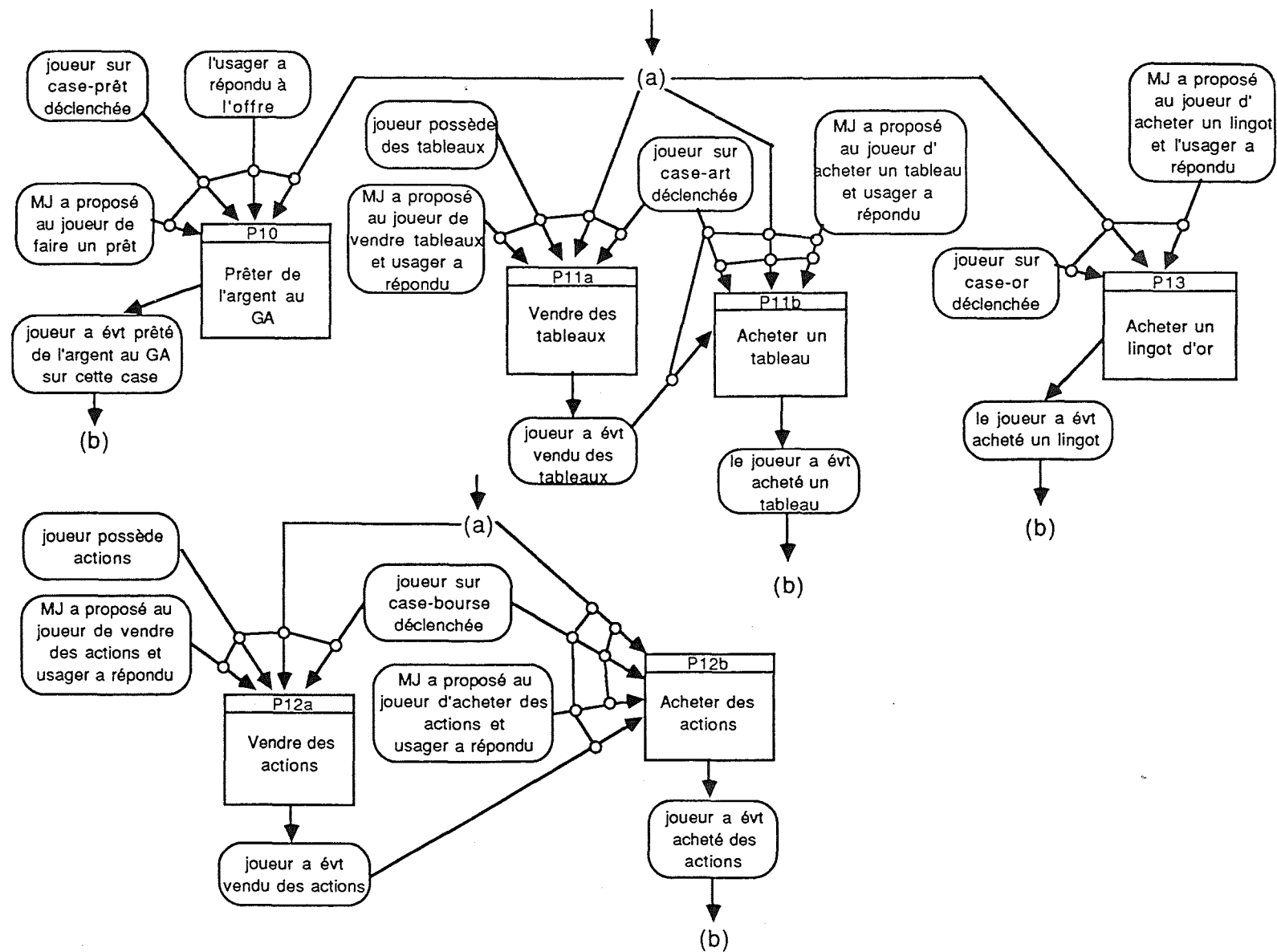


Diagramme de transition du JOUEUR (2)

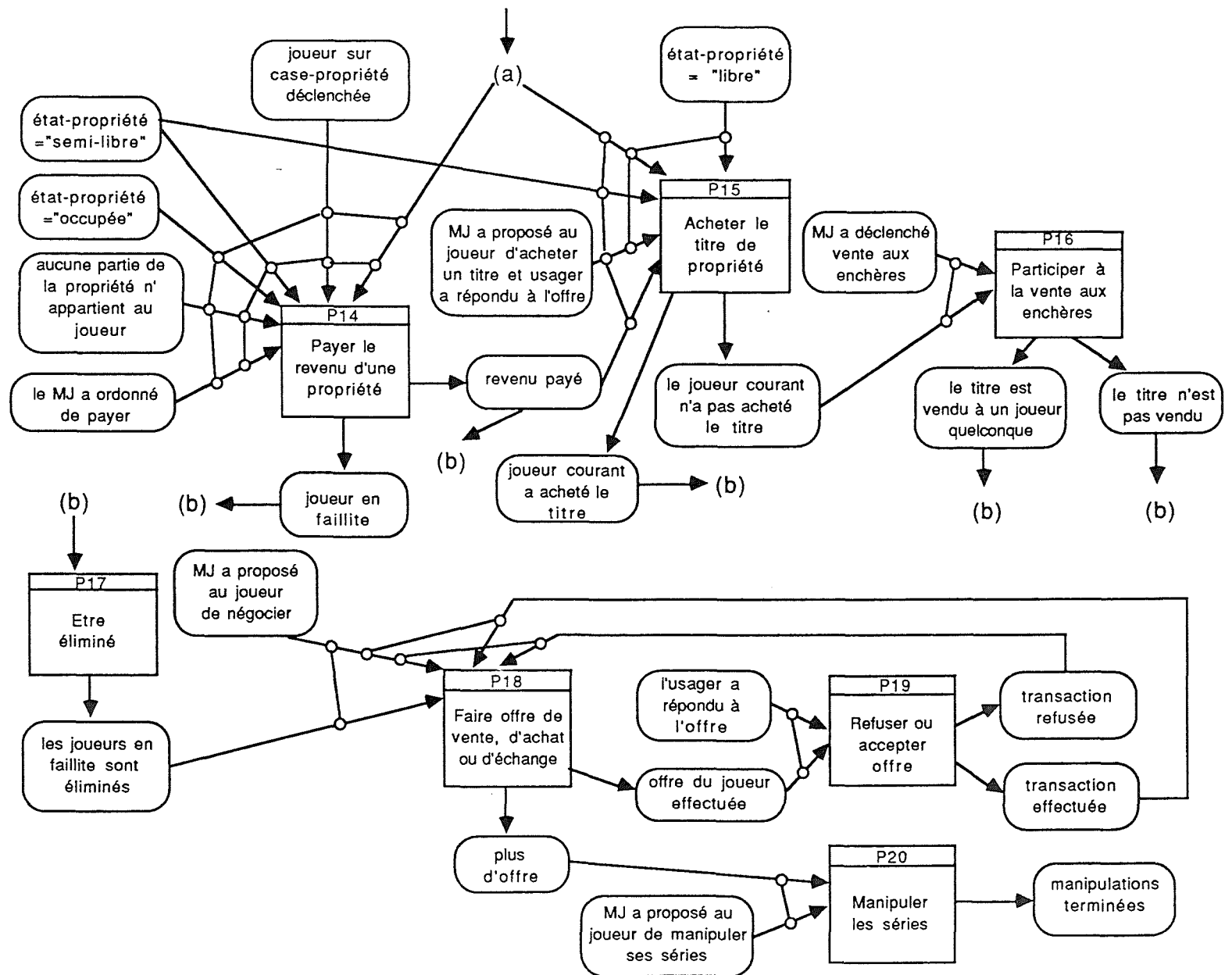
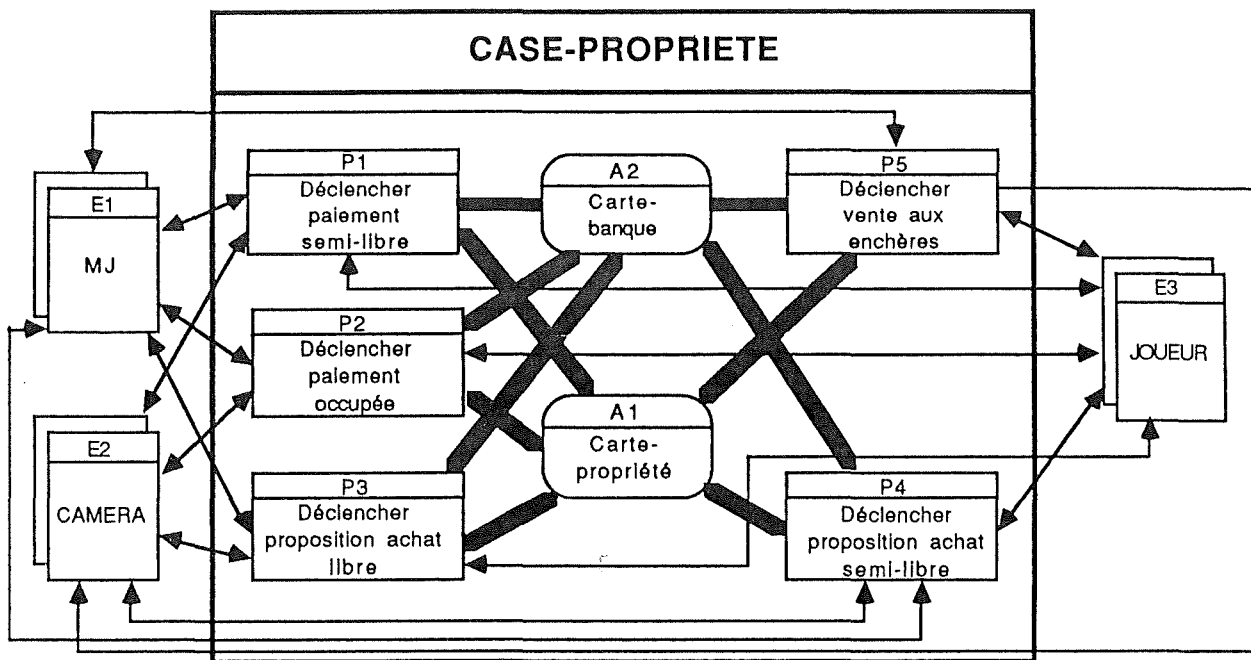
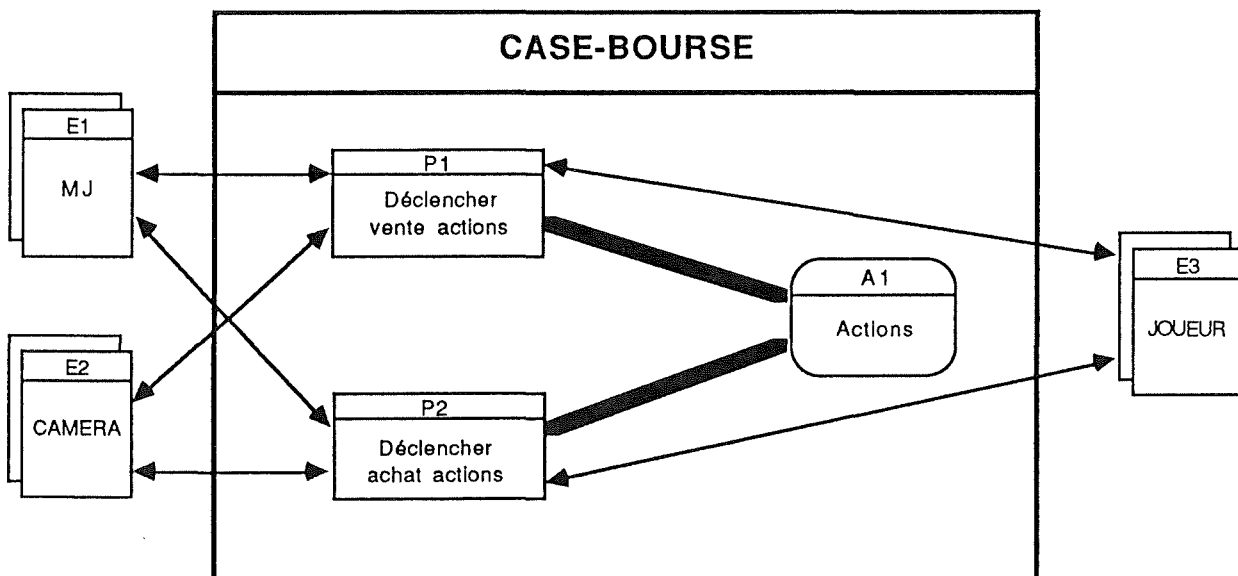


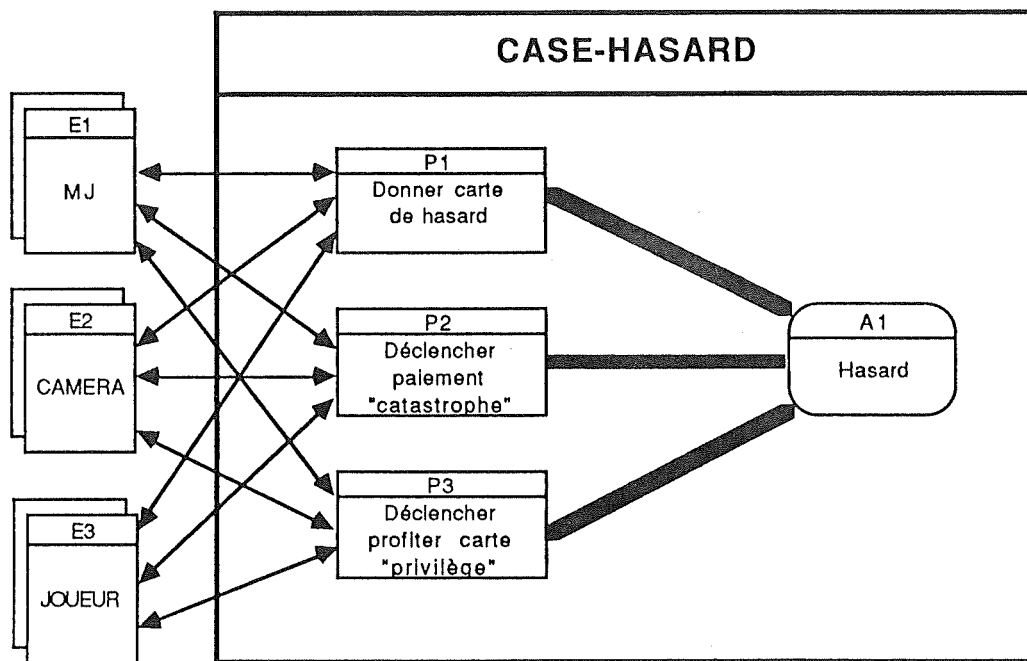
Diagramme de transition du JOUEUR (3)



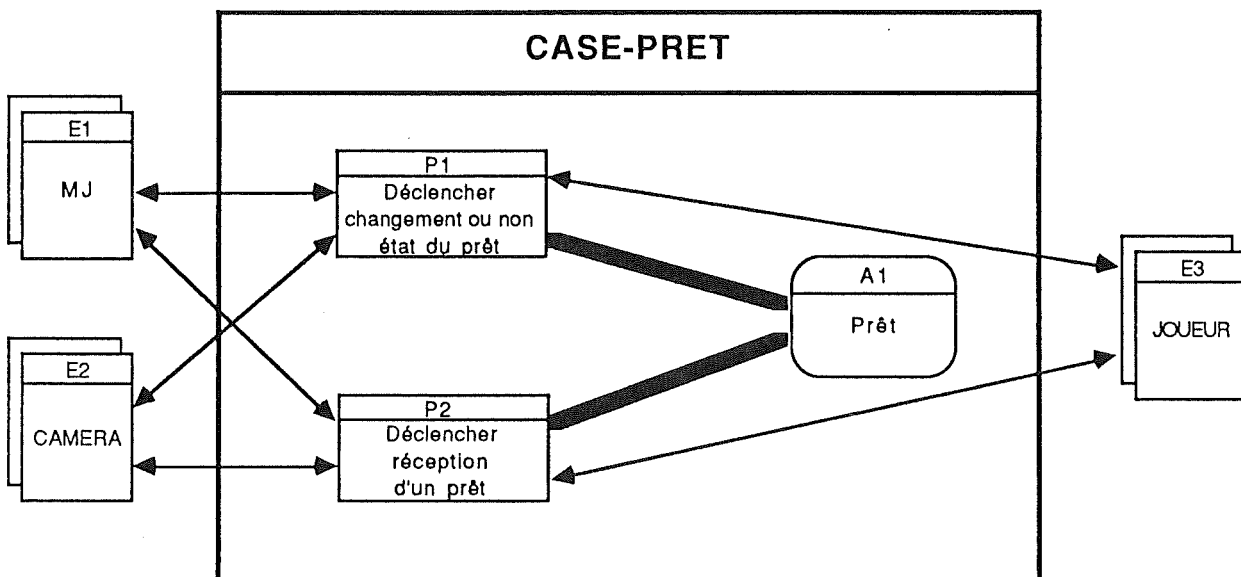
Plan de système de la CASE-PROPRIETE



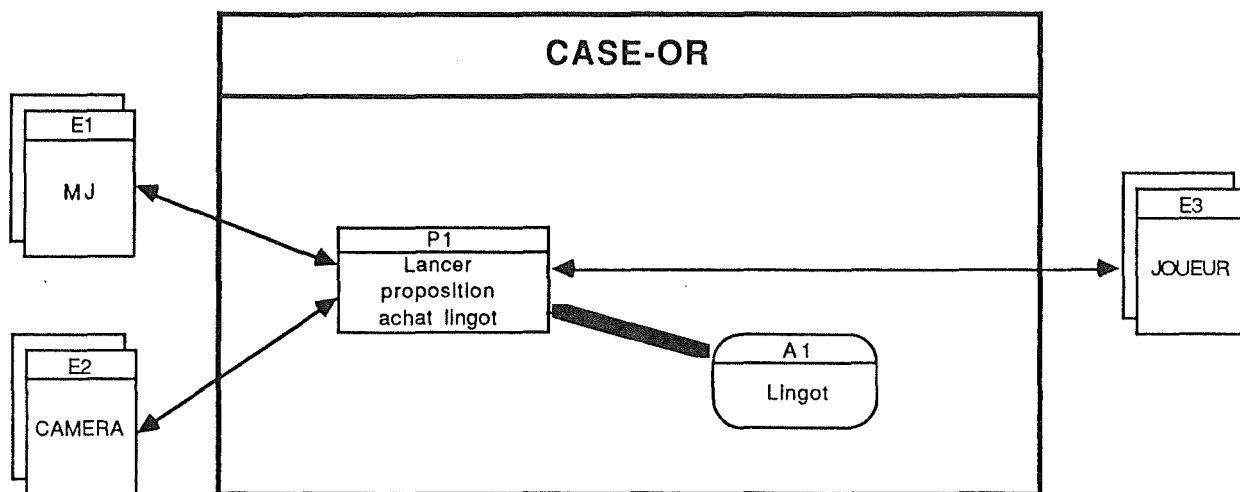
Plan de système de la CASE-BOURSE



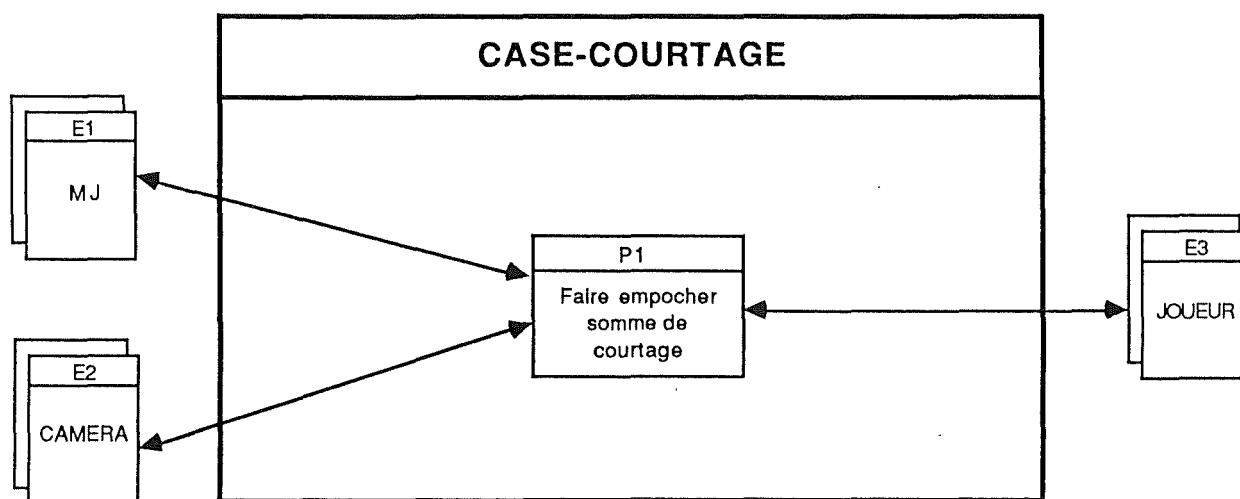
Plan de système de la CASE-HASARD



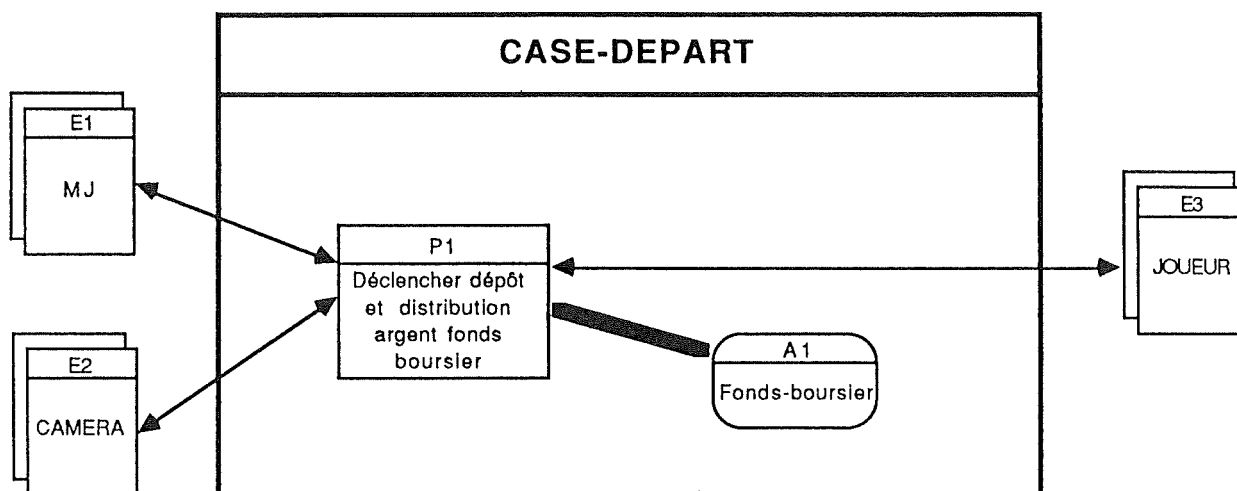
Plan de système de la CASE-PRET



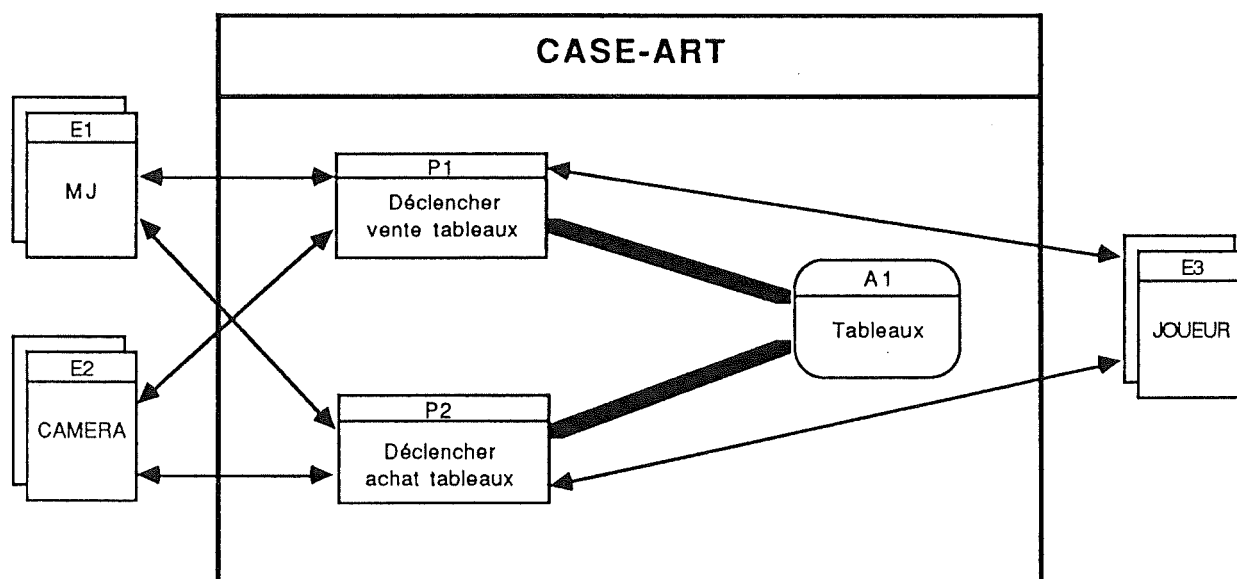
Plan de système de la CASE-OR



Plan de système de la CASE-COURTAGE



Plan de système de la CASE-DEPART



Plan de système de la CASE-ART

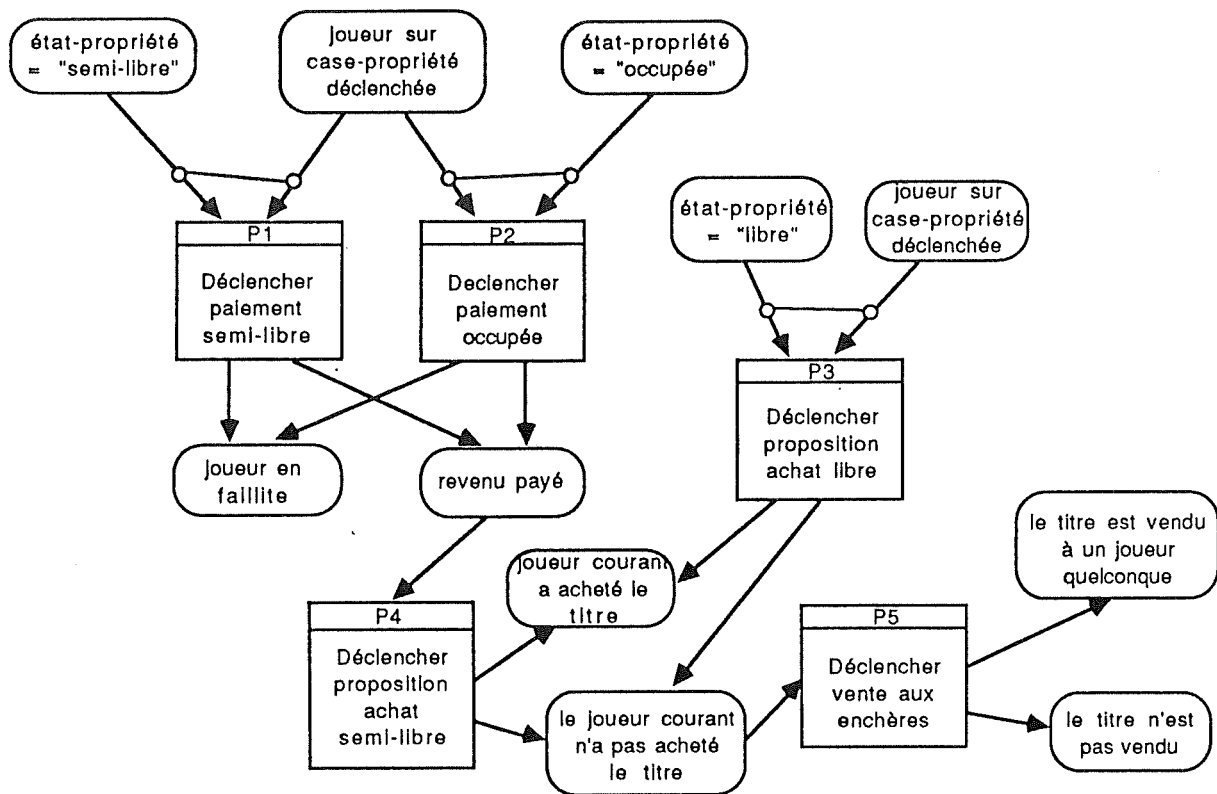


Diagramme de transition de la CASE-PROPRIETE

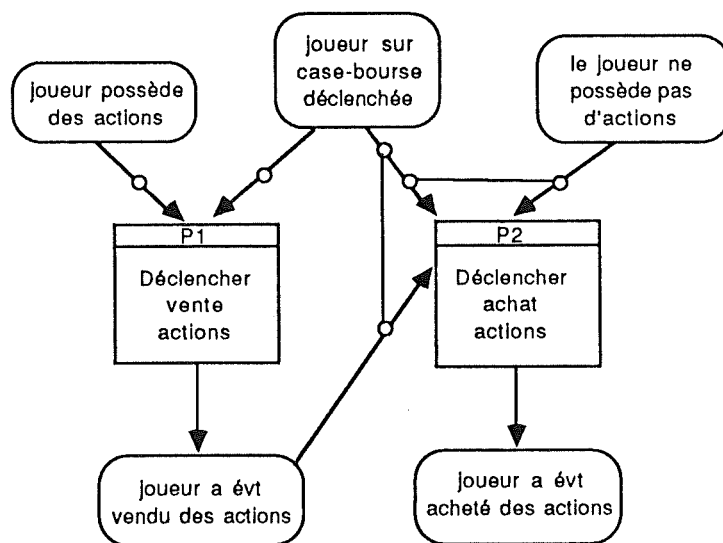


Diagramme de transition de la CASE-BOURSE

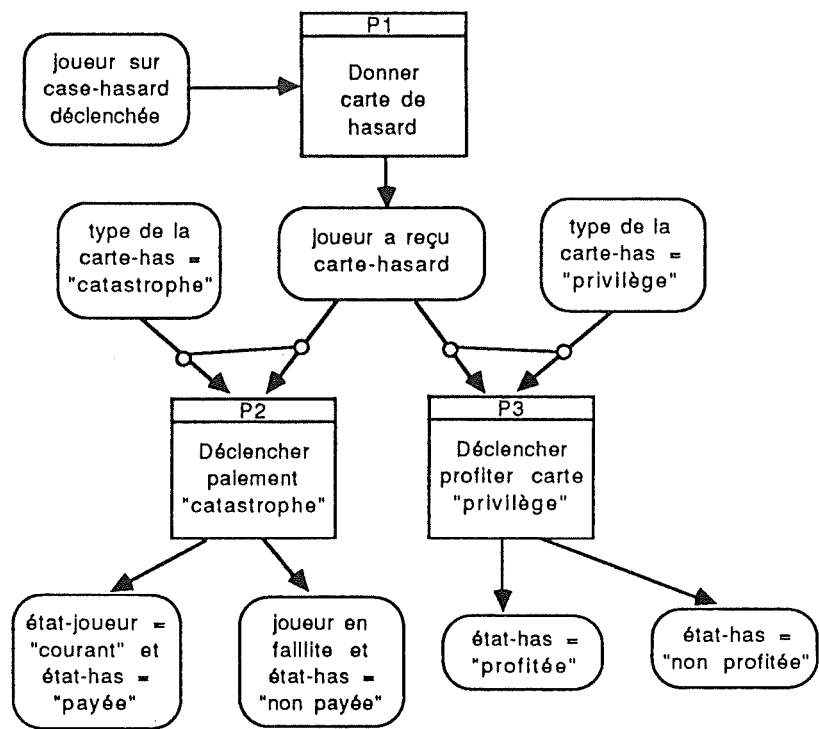


Diagramme de transition de la CASE-HASARD

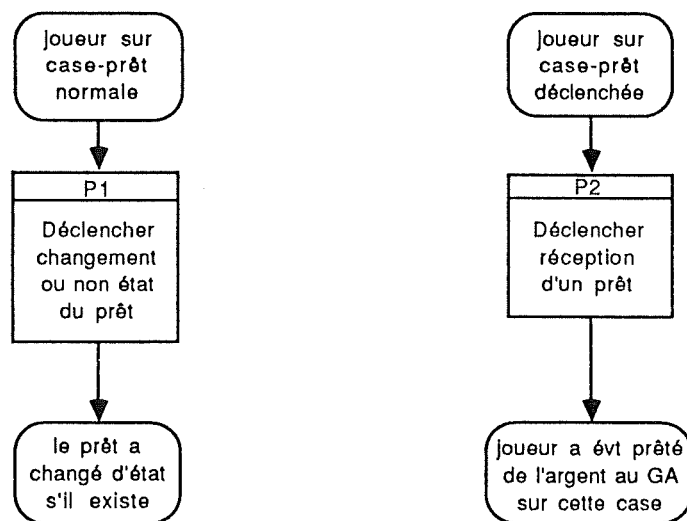


Diagramme de transition de la CASE-PRET

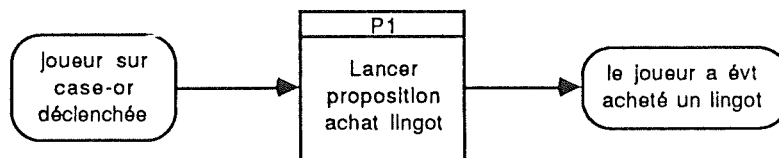


Diagramme de transition de la CASE-OR

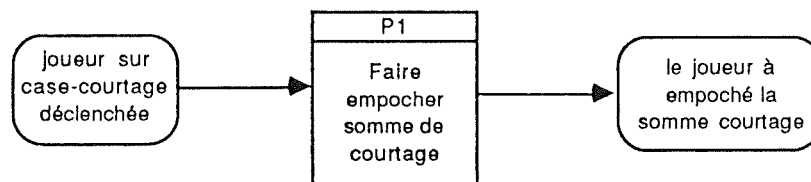


Diagramme de transition de la CASE-COURTAGE

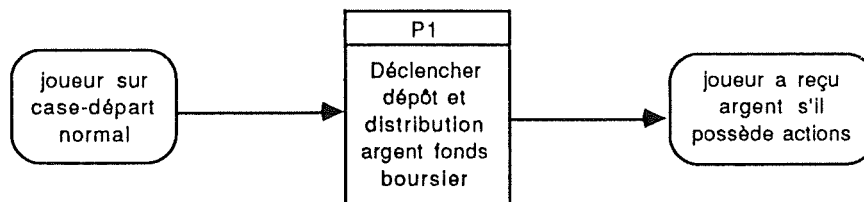


Diagramme de transition de la CASE-DEPART

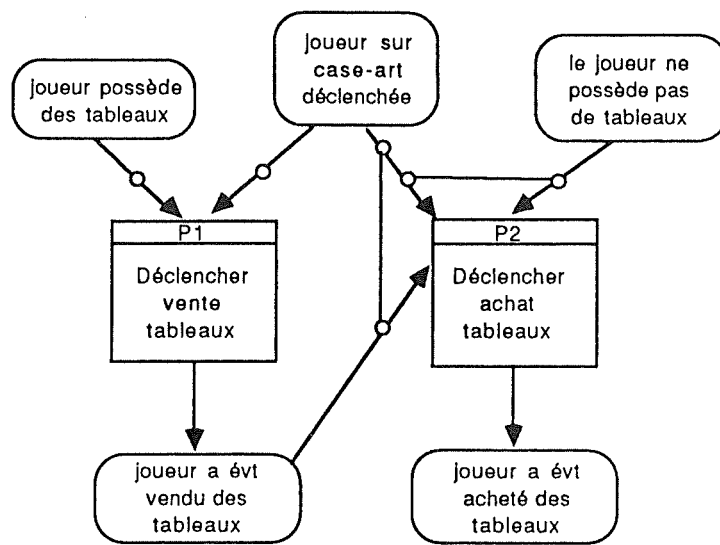


Diagramme de transition de la CASE-ART

Annexe 4 :

Caractéristiques du langage Smalltalk/V

1. Introduction

Cette annexe apporte quelques compléments d'information à propos du langage Smalltalk/V, dans lequel nous avons programmé notre application.

Smalltalk est, avec Simula, le chef de file historique des langages de classes. Il a été créé par Alan Kay dans les années 1970, au Palo Alto Research Center.

Smalltalk se caractérise par l'homogénéité de son univers. Celui-ci est composé seulement d'objets, instances de classes, qui communiquent par envoi de messages. Le système Smalltalk est à la fois un langage, un système d'exploitation et un environnement de programmation.

Les principales caractéristiques de Smalltalk/V sont décrites dans les points suivants.

2. Héritage

En Smalltalk/V, chaque classe n'a qu'une superclasse, bien que plusieurs classes puissent partager la même superclasse. Il n'y a donc pas d'héritage multiple. De cette manière, les classes d'un système forment un arbre dont la racine est la classe appelée "Object", qui est la seule à ne pas posséder de superclasse.

Chaque classe hérite, par défaut, des caractéristiques (variables, méthodes) de sa superclasse. Cela signifie que toute instance d'une classe est aussi une instance de sa superclasse. En suivant la chaîne transitive d'héritages, on déduit que la classe "Object" décrit toutes les caractéristiques minimales de tous les objets.

La figure A4.1. représente sous forme d'arbre d'héritage quelques-unes des principales classes du système Smalltalk.

3. Métaclasse

En vertu du premier principe de Smalltalk, tout objet (et donc aussi les classes) est l'instance d'une classe. Les classes sont des instances particulières appelées **métaclasses**. A chaque classe C est associée implicitement, dès qu'elle est créée, une métaclasse "C-class", qui permet de définir un comportement particulier à la classe C, son unique instance. Toutes les métaclasses sont elles-mêmes instance d'une même classe, appelée "Métaclass" qui spécifie leur

comportement commun. Pour des raisons de cohérence, "Métaclass" est définie comme une instance de "Métaclass-class", elle-même instance de "Métaclass".

Le schéma présenté à la figure A4.2. illustre sous forme d'arbre cette situation.

4. Méthodes

Une méthode est associée à une classe et "décrit une séquence d'actions à accomplir quand un message contenant un sélecteur particulier est reçu par une instance d'une classe particulière". En d'autres mots, "une méthode est un algorithme qui détermine le comportement et la représentation d'un objet. Cela correspond à la définition de fonctions dans d'autres langages. Quand un message est envoyé à un objet, une méthode est évaluée, et un objet est retourné comme résultat" (cf [STV]).

Ces actions à accomplir ou algorithmes sont représentés par des expressions Smalltalk séparées par des points; ces expressions sont évaluées séquentiellement et de gauche à droite.

Supposons, par exemple, qu'un objet d'une classe quelconque ait un nom comme variable propre. On peut imaginer lui associer la méthode suivante :

```
changerNom : nouveauNom.  
    nom := nouveauNom.
```

La méthode dont le sélecteur est changerNom: a pour but de remplacer le contenu de la variable "nom" de l'objet receveur du message par la valeur du paramètre nouveauNom.

5. Définition de classe

Pour définir une classe en Smalltalk/V, il faut lui donner un nom, indiquer qu'elle est sa superclasse, quelles sont ses variables d'instance et de classe et quelles sont ses méthodes d'instance et de classe.

Une variable ou une méthode de classe est une variable ou une méthode qui est unique pour tous les objets de la classe. Une variable ou une méthode d'instance est propre à une instance d'une classe et existe donc conceptuellement autant de fois qu'il y a d'objets dans la classe.

6. Les classes prédéfinies

Une des originalités de Smalltalk tient au fait que le code du logiciel est lui-même programmé, en grande partie, en Smalltalk. Le système utilise des classes et des méthodes prédéfinies. De nombreuses classes prédéfinies le sont non pas parce qu'elles sont vitales au

système mais parce qu'elles sont susceptibles d'être utilisées souvent par le programmeur. Ceci simplifie le travail de programmation. De plus, comme ces classes prédéfinies sont écrites en Smalltalk, on peut en lire le code, s'en servir comme exemple et donc accélérer le processus d'apprentissage de Smalltalk/V.

7. Remarques

Smalltalk/V permet, comme la plupart des langages, de procéder à des affectations de valeurs, de traiter des éléments appartenant à des tableaux, de gérer des fichiers, de mettre en oeuvre des structures algorithmiques plus complexes telles que la condition et la boucle, etc.

8. Interface sur Macintosh

Smalltalk/V n'est pas seulement un langage; c'est un environnement de programmation complet. En effet, ce système combine un compilateur, un "debugger" (permettant de corriger les erreurs d'exécution), un "browser" (permettant de circuler à travers la hiérarchie des classes), un mécanisme d'inspection, un éditeur de texte, un éditeur graphique et une interface utilisateur basée sur un système de multi-fenêtrage.

L'environnement de Smalltalk/V est adapté aux ordinateurs Macintosh, parce que ces supports hardware présentent des facilités d'interfaçage particulières (écran "bitmap", souris, multi-fenêtrage, ...).

(Extraits repris de [DEL-JOU], p. 171 à 176)

Annexe 5 : Implantation de la caméra

Le texte Smalltalk/V correspondant à l'implantation de la caméra sera disponible le jour de la défense du mémoire.