Notation et processus outillé pour la description, l'analyse et la compréhension de l'automatisation dans les systèmes de commande et contrôle

Elodie Bouzekri

▶ **To cite this version:**

Elodie Bouzekri. Notation et processus outillé pour la description, l'analyse et la compréhension de l'automatisation dans les systèmes de commande et contrôle. Informatique [cs]. Université toulouse 3 Paul Sabatier, 2021. Français. NNT : 2021TOU30003 . tel-03208713v1

HAL Id: tel-03208713

https://theses.hal.science/tel-03208713v1

Submitted on 3 Dec 2023 (v1), last revised 26 Apr 2021 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par l'Université Toulouse 3 - Paul Sabatier

Présentée et soutenue par

Elodie BOUZEKRI

Le 14 janvier 2021

Notation et processus outillé pour la description, l'analyse et la compréhension de l'automatisation dans les systèmes de commande et contrôle

---

Ecole doctorale : **EDMITT - Ecole Doctorale Mathématiques, Informatique et Télécommunications de Toulouse**

Spécialité : **Informatique et Télécommunications**

Unité de recherche :
IRIT : Institut de Recherche en Informatique de Toulouse

Thèse dirigée par
Philippe PALANQUE et Célia MARTINIE

Jury

M. Jean VANDERDONCKT, Rapporteur
M. Xavier LE PALLEC, Rapporteur
M. Kris LUYTEN, Examinateur
M. Marco Antonio ALBA WINCKLER, Examinateur
Mme Christine GRIS, Examinatrice
M. Philippe PALANQUE, Directeur de thèse
Mme Célia MARTINIE, Co-directrice de thèse

Remerciements

Je voudrais commencer par remercier Philippe Palanque et Célia Martinie qui m'ont donné l'opportunité de réaliser cette thèse. Je me sens chanceuse d'avoir pu découvrir la recherche académique aux côté de chercheurs ouverts et reconnus à l'internationale. Ainsi, je vous remercie chaleureusement pour votre accompagnement, votre disponibilité et votre patience tout à long de cette thèse. Merci d'être resté une source d'inspiration et de référence durant ces années de doctorat où, à vos côtés, j'ai pu apprendre à renforcer mon esprit critique et ma rigueur.

J'adresse également tous mes remerciements à Jean Vanderdonckt et Xavier Le Pallec pour avoir accepté d'être les rapporteurs de mon manuscrit et avoir consacré de leur temps à sa relecture. Je vous remercie également pour les conseils et les avis enrichissants sur les travaux présentés dans ce manuscrit. Je remercie également M. Kris Luyten, M. Marco Antonio Alba Winckler et Mme Christine Gris d'avoir accepté de faire partie du jury de thèse.

J'adresse un merci tout particulier à Christine Gris avec qui nous avons pu travailler sur le projet IKKY pour son expertise, ses retours précieux, son aide et sa disponibilité à toutes les réunions de travail hebdomadaires. Merci aussi à Yannick Déléris avec qui nous avons débuté le projet IKKY avant que Christine Gris ne prenne le relais. De nombreux travaux de cette thèse ont été rendus possibles grâce au financement de AIRBUS, notamment par le biais du projet IKKY.

Je remercie également le Zonta pour la reconnaissance des travaux réalisés pendant cette thèse à travers le prix Amélia Earhart qui m'a encouragé à continuer dans cette voie.

Un grand merci aussi à tous les membres, ou anciens membres, de l'équipe ICS et SIERA que j'ai pu rencontrer pour leur soutien et leur bonne humeur. Je remercie en particulier Alexandre Canny et Eric Barboni, partenaires de bureau et de discussion privilégiés. Merci à Guillaume Pottier, Alexandre Battut, Dimitri Drouet, Jean Luc Hak, David Navarre et Thiago Rocha Silva pour leur bonne humeur et tous les échanges qui ont animés nos pauses café. Et merci à Racim Fahssi, Camille Fayolas et Martin Cronel pour leur gentillesse et leurs retours sur l'expérience qu'est un doctorat.

Enfin, j'adresse un merci à mes parents et à mes frères pour leur soutien et simplement pour les liens qui nous unissent et le repère qu'ils m'apportent. Je remercie mon compagnon Victor pour son soutien et sa présence durant toutes ces années. Merci également à mes amis que je n'ai pas encore remerciés Julie, Alex, Anaïs, Sébastien et Lily.

Table des matières

Table des matières *5*

Liste des figures *13*

Liste des tables *17*

Introduction *19*

1 Problématique **19**

2 Hypothèse de la thèse **20**

3 Structure de la thèse **21**

Partie I. Etat de l'art ***23***

Chapitre 1. Systèmes interactifs et systèmes interactifs critiques ***24***

1 Systèmes interactifs **24**

1.1 Définition d'un système interactif 24

1.2 Architectures des systèmes interactifs 25

1.2.1 Les modèles à couches 25

1.2.2 Les modèles à agents 26

1.3 Propriétés des systèmes interactifs 27

1.3.1 Utilisabilité 27

1.3.2 Propriétés externes et internes de (Cockton & Gram, 1996) 28

Propriétés externes 28

Propriétés internes 29

1.3.3 Expérience ou Ressenti Utilisateur 29

2 Systèmes interactifs critiques **30**

2.1 Définition d'un système critique 30

2.2 Système interactif critique : le cas des systèmes de commande et contrôle 31

2.3 Propriétés des systèmes interactifs critiques 31

2.4 Cockpit d'avion civil 32

2.4.1 DO-178C et ARP 4754 34

2.4.2 CS25 36

3 Conclusion **39**

Chapitre 2. Systèmes contenant de l'automatisation et leur analyse ***40***

1 Définitions de l'automatisation **40**

2 Propriétés de l'automatisation **41**

2.1 Trust 41

2.2 *Situation awareness* 42

3 Allocation de fonctions et de tâches **43**

3.1 MABA MABA 43

3.2 Niveaux d'automatisation 45

3.2.1 Niveaux d'automatisation de Sheridan, Verplank et Parasuraman 45
3.2.2 Niveaux d'automatisation d'Endsley et Kiris 46
3.2.3 Autres niveaux d'automatisation 48
3.2.4 Niveaux d'automatisation pour les voitures autonomes 49
3.3 Allocation dynamique des tâches et des fonctions 51
3.4 Autres approches pour l'allocation de tâches et de fonctions 52
3.4.1 IDA-S (Andrew Dearden, 2001) 52
3.4.2 Métaphore du cheval (F. Flemisch et al., 2005) 54

4 Allocation des ressources, de l'autorité, de la responsabilité et transitions de contrôle 54
4.1 Contrôle, Autorité, Responsabilité et transitions de contrôle 55
4.1.1 Contrôle 55
4.1.2 Autorité 55
4.1.3 Responsabilité 55
4.1.4 Transitions de contrôle de la tâche de conduite avec un véhicule autonome 56
4.2 Allocation de ressource : transparence 57

5 Conclusion 58

Chapitre 3. Modélisation de l'humain et de ses tâches 60

1 Modèles de l'Humain : décomposition des tâches 60
1.1 Décomposition des tâches humaines 60
1.1.1 Le modèle SRK 60
1.1.2 Le modèle du processeur humain 61
1.1.3 La théorie de l'action de Norman 62
1.2 Décomposition des tâches ou fonctions allouables à un humain ou un système contenant de l'automatisation 63
1.2.1 Perception, cognition et action (Task-Agent Control Space) 63
1.2.2 Information, décision et action (IDA Cycle model) 64
1.2.3 Monitoring, génération, sélection et implémentation 65
1.2.4 Information, décision, action et supervision (IDA-S) 65
1.2.5 Le modèle simple en quatre étapes du traitement de l'information par l'humain et par le système contenant de l'automatisation 65
1.3 Décomposition des tâches coopératives 67
1.4 Synthèse 68

2 Analyse et la modélisation des tâches utilisateur 69
2.1 Analyse de tâches 69
2.2 Modèles de tâches 69
2.3 Notations et outils de modélisation de tâches 71
2.3.1 Acteurs et rôles 71
2.3.2 Présentation des notations pour la modélisation de tâches 72
2.4 Comparaison 76

3 Notation HAMSTERS et outil HAMSTERS 78
3.1 Types de tâche 78
3.1.1 Types de tâches génériques 78
3.1.2 Tâches perceptives, cognitives, motrices et interactives 79
3.1.2.1 Tâches perceptives, cognitives et motrices 79
3.1.2.2 Tâches cognitives d'analyse et de décision et tâches interactives 80

3.1.3 Fonctions systèmes 81
3.2 Hiérarchie et Ordonnancement temporel des tâches 83
3.2.1 Opérateurs d'ordonnancement temporel 83
3.2.2 Tâches itératives et optionnelles 85
3.2.3 Relations temporelles quantitatives 85
3.2.4 Copie, Sub-routines et Composants 86
3.2.4.1 Copy task 86
3.2.4.2 Sub-routines 86
3.2.4.3 Composants 87
3.3 Description des données, périphériques et objets (DOD) 88
3.3.1 Données et objets 88
3.3.2 Périphériques 90
3.3.3 Relations entre les DODs et les taches 90
3.4 Outil HAMSTERS 92
3.4.1 Edition de modèle de tâches 92
3.4.2 Simulation de modèle de tâches 94

4 Conclusion 96

Partie II. Contribution 97

Chapitre 4. Cadre conceptuel pour l'analyse des systèmes interactifs contenant de l'automatisation et des tâches de leurs utilisateurs 98

1 Un exemple illustratif : Le jeu des 15 99
1.1 Présentation du jeu des 15 99
1.2 Concept map du jeu des 15 99

2 Fonctions et de Tâches 101
2.1 Définitions de fonctions et de tâches 101
2.2 Définition de la description de l'allocation des Fonctions et des Tâches 102
2.3 Rôles des utilisateurs et des systèmes, buts, finalités et abilités 102
2.4 Description de l'allocation des Fonctions et des Tâches sur l'exemple du jeu des 15 103
2.5 Relations de la description de l'allocation des fonctions et des tâches avec les autres éléments du cadre conceptuel RCRAFT 106

3 Ressources 106
3.1 Définition de Ressource 106
3.2 Définition de la description de l'allocation des Ressources 107
3.3 Identification des instances, des rôles et acteurs possesseurs et acquéreurs des ressources 107
3.4 Description de l'allocation des Ressources sur l'exemple du jeu des 15 108

4 Autorité 110
4.1 Définition de l'Autorité 111
4.2 Définition de la description de l'allocation de l'Autorité 111
4.3 Les contraintes posées aux acteurs par les acteurs 111
4.4 Description de l'allocation de l'Autorité sur l'exemple du jeu des 15 112

5 Transitions de contrôle et de l'initiative 113
5.1 Définition des transitions de contrôles et de l'initiative 113
5.2 Définition de l'allocation des Transitions de contrôle et de l'initiative 114
5.3 Aspects temporels et les conditions des transitions de contrôle et de l'initiative 114

5.3.1 Séquence ou interruption 114
5.3.2 Contraintes temporelles des transitions de contrôle 114
5.3.3 Transitions de contrôle optionnelles ou nécessaires 115
5.4 Description de l'allocation des Transitions de contrôle et de l'initiative sur l'exemple du jeu des 15 115

6 Responsabilité 117
6.1 Définition de la Responsabilité 117
6.2 Définition de la description de l'allocation de la Responsabilité 117
6.3 Description de l'allocation de la Responsabilité en détail 117
6.4 Description de l'allocation de la Responsabilité sur l'exemple du jeu des 15 118

7 Conclusion 120

Chapitre 5. Notation pour la représentation des éléments RCRAFT 121

1 D'une modélisation intégrée à une modélisation ségréguée 121
1.1.1 L'exemple illustratif sur un distributeur automatique de billets 123
1.1.2 La modélisation intégrée du comportement de l'utilisateur avec le système interactif 124
1.1.3 La modélisation ségréguée du comportement de l'utilisateur avec le comportement du système 126

2 Coopération entre tâches de modèles de tâches ségrégués et fonctions de modèle de fonctions 128
2.1 Tâches et fonctions coopératives 128
2.1.1 Tâche coopérative 128
2.1.2 Fonction coopérative 128
2.1.3 Exemple de tâche et de fonction coopérative sur l'exemple du distributeur de billets 129
2.2 Protocole de coopération 129
2.2.1 Type de coopération du protocole de coopération 131
2.2.2 Protocole de coopération synchrone ou asynchrone 131
2.2.3 Cardinalité du protocole de coopération 132
2.2.4 Localisation du protocole de coopération 132
2.2.5 Les ressources reçues ou émises du protocole de coopération 132

3 Eléments de représentation RCRAFT 132
3.1 Allocation des fonctions et tâches 132
3.2 Allocation de ressources 133
3.3 Allocation des transitions de contrôle et de l'initiative 133
3.4 Allocation de la responsabilité 135
3.5 Allocation de l'autorité 135

4 Conclusion 135

Chapitre 6. Processus RCRAFT d'analyse et de description systématique de RCRAFT 137

1 Description du processus RCRAFT 137
1.1 Vision haut-niveau du processus RCRAFT 137
1.2 Vue détaillée du processus RCRAFT 141
1.2.1 Phase de modélisation des tâches explicitant l'allocation de fonctions et de tâches 141
1.2.2 Phase de description de l'allocation de ressources, les transitions de contrôle et de de l'initiative, de la responsabilité et de l'autorité 142
1.2.2.1 Extraction des DODs 143

1.2.2.2 Description de l'allocation de ressources 144
1.2.2.3 Description de l'allocation des transitions de contrôle et de l'initiative 145
1.2.2.4 Description de l'allocation de la responsabilité 147
1.2.2.5 Description de l'allocation de l'autorité 149
1.2.3 Phase de Modélisation et d'Analyse RCRAFT 150

2 Application du processus RCRAFT à l'exemple du jeu des 15 151
2.1 Phase de modélisation des tâches explicitant l'allocation de fonctions et de tâches 151
2.2 Phase de description de l'allocation de ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité 157
2.2.1 Extraction des DODs 157
2.2.2 Description de l'allocation de ressources 158
2.2.3 Description de l'allocation des transitions de contrôle et de l'initiative 162
2.2.4 Description de l'allocation de la responsabilité 165
2.2.1 Description de l'allocation de l'autorité 168
2.3 Phase de modélisation et d'analyse RCRAFT 170

3 Conclusion 176

Chapitre 7. Extension de l'outil Hamsters pour l'édition des modèles tâches dans le cadre de l'analyse de l'automatisation 177

1 Extension de l'outil HAMSTERS pour la représentation de la coopération 177
1.1 Ressources du protocole de coopération 177
1.2 Protocole de coopération synchrone ou asynchrone 178
1.3 Localisation de la coopération 178
1.4 Cardinalité du protocole de coopération 179
1.5 Type de coopération 179

2 Extension de l'outil HAMSTERS pour la représentation des éléments RCRAFT 179
2.1 Exploitation des modèles pour la représentation de l'Allocation des transitions de contrôle et de l'initiative 180
2.2 Exploitation des modèles pour la représentation de l'allocation de la Responsabilité 180
2.3 Exploitation des modèles pour la représentation de l'allocation de l'Autorité 181

3 Conclusion 182

Partie III. Validation de la contribution 183

Chapitre 8. Validation de l'expressivité de RCRAFT sur des variantes du jeu des 15 184

1 Version du jeu des 15 avec un temps limité pour jouer son tour 184
1.1 Présentation du jeu des 15 avec un temps limité pour jouer son tour 184
1.2 Description de l'allocation RCRAFT pour la version du jeu des 15 avec un temps limité pour jouer son tour 185
1.2.1 Description de l'allocation des tâches et des fonctions pour la version du jeu des 15 avec un temps limité pour jouer son tour 185
1.2.2 Description de l'allocation des ressources pour la version du jeu des 15 avec un temps limité pour jouer son tour 190
1.2.3 Description de l'allocation des transitions de contrôle et de l'initiative pour la version du jeu des 15 avec un temps limité pour jouer son tour 193
1.2.4 Description de l'allocation de la responsabilité pour la version du jeu des 15 avec un temps limité pour jouer son tour 196

1.2.5 Description de l'allocation de l'autorité pour la version du jeu des 15 avec un temps limité pour jouer son tour 198
1.2.1 Description de l'allocation RCRA dans les modèles de tâches ségrégués et de fonctions..... 199
1.3 Comparaison des versions du jeu des 15 202
1.3.1 Comparaison de l'allocation des fonctions et des tâches 202
1.3.1.1 Identification des différences..... 202
1.3.1.2 Impact 203
1.3.2 Comparaison de l'allocation des ressources 203
1.3.2.1 Identification des différences..... 203
1.3.2.2 Impact 204
1.3.3 Comparaison de l'allocation des transitions de contrôle et de l'initiative 204
1.3.3.1 Identification des différences..... 204
1.3.3.1 Impact 204
1.3.4 Comparaison de l'allocation de la responsabilité 204
1.3.4.1 Identification des différences..... 204
1.3.4.1 Impact 205
1.3.5 Comparaison de l'allocation de l'autorité..... 205
1.3.5.1 Identification des différences..... 205
1.3.5.1 Impact 205
2 Version du jeu des 15 proposant des recommandations aux joueurs 205
2.1 Présentation du jeu des 15 proposant des recommandations aux joueurs..... 205
2.2 Description de l'allocation RCRAFT pour la version du jeu des 15 proposant des recommandations aux joueurs 206
2.2.1 Description de l'allocation des tâches et des fonctions pour la version du jeu des 15 proposant des recommandations aux joueurs 206
2.2.2 Description de l'allocation des ressources pour la version du jeu des 15 proposant des recommandations aux joueurs 209
2.2.3 Description de l'allocation des transitions de contrôle et de l'initiative pour la version du jeu des 15 proposant des recommandations aux joueurs 212
2.2.4 Description de l'allocation de la responsabilité pour la version du jeu des 15 proposant des recommandations aux joueurs 214
2.2.5 Description de l'allocation de l'autorité pour la version du jeu des 15 proposant des recommandations aux joueurs 216
2.2.6 Description de l'allocation RCRA dans les modèles de tâches ségrégués et de fonctions..... 218
2.3 Comparaison des versions du jeu des 15 218
2.3.1 Comparaison de l'allocation des tâches et des fonctions 219
2.3.1.1 Identification des différences..... 219
2.3.1.2 Impact 219
2.3.2 Comparaison de l'allocation des ressources 219
2.3.2.1 Identification des différences..... 219
2.3.2.2 Impact 220
2.3.3 Comparaison de l'allocation des transitions de contrôle et de l'initiative 220
2.3.3.1 Identification des différences..... 220
2.3.3.2 Impact 220
2.3.4 Comparaison de l'allocation de la responsabilité 220
2.3.4.1 Identification des différences..... 221
2.3.4.2 Impact 221
2.3.5 Comparaison de l'allocation de l'autorité..... 221
2.3.5.1 Identification des différences..... 221

2.3.5.2 Impact 221
2.4 Synthèse des comparaisons 221
3 Conclusion 222
Chapitre 9. Etude de cas : Analyse RCRAFT de systèmes de gestions des alarmes des cockpits AIRBUS 224
1 Présentation du projet de recherche IKKY 225
2 Système de gestion des alarmes actuel 226
2.1 Présentation du « Flight Warning System » 226
2.2 Procédures de résolution des alarmes 227
2.2.1 Priorités des alarmes 227
2.2.2 Supprimer une alarme intempestive 228
2.2.3 Effacer une procédure et valider les lignes de procédures 229
2.2.4 Naviguer parmi la procédure et les alarmes 229
2.2.5 Rappeler la dernière ou toutes les procédures et alarmes 229
2.2.6 Les pages systèmes 230
2.3 Panneau plafond 230
2.4 Rôles de pilote *flying* et de pilote *monitoring* 231
2.5 Analyse RCRAFT de la gestion des défaillances du pilote « monitoring » avec le « flight warning system » 231
2.5.1 Modélisation des tâches du pilote monitoring et des fonctions du FWS 232
2.5.1.1 Phase 1 du processus RCRAFT : Modélisation des fonctions du FWS 233
2.5.1.2 Phase 1 du processus RCRAFT : Modélisation des tâches du pilote Monitoring 234
2.5.2 Allocation des ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité entre le pilote monitoring et le FWS 235
2.5.2.1 Phase 2 du processus RCRAFT : Allocation des ressources 236
2.5.2.2 Phase 2 du processus RCRAFT : Allocation des transitions de contrôle et de l'initiative 237
2.5.2.3 Phase 2 du processus RCRAFT : Allocation de la responsabilité 240
2.5.2.4 Phase 2 du processus RCRAFT : Allocation de l'autorité 241
2.5.2.5 Phase 3 du processus RCRAFT : Modélisation RCRAFT 243
3 Prototype de système alternatif de gestion des alarmes 243
3.1 Interface pour donner le choix au pilote 243
3.1.1 Recommandations d'actions de récupération plutôt que des procédures 244
3.1.1.1 Recommandations d'actions de récupération immédiates 244
3.1.1.2 Recommandations d'actions de récupération non immédiates 244
3.1.2 Etats systèmes 245
3.2 Analyse RCRAFT de la gestion des défaillances du pilote « monitoring » avec le système alternatif de gestion des alarmes 246
3.2.1 Modélisation des tâches du pilote monitoring et des fonctions du recommandeur 247
3.2.1.1 Phase 1 du processus RCRAFT : Modélisation des fonctions du recommandeur 248
3.2.1.2 Phase 1 du processus RCRAFT : Modélisation des tâches du pilote monitoring 249
3.2.2 Allocation des ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité entre le pilote monitoring et le recommandeur 251
3.2.2.1 Phase 2 du processus RCRAFT : Allocation des ressources 252
3.2.2.2 Phase 2 du processus RCRAFT : Allocation des transitions de contrôle et de l'initiative 253
3.2.2.3 Phase 2 du processus RCRAFT : Allocation de la responsabilité 256
3.2.2.4 Phase 2 du processus RCRAFT : Allocation de l'autorité 258

3.2.2.5 Phase 3 du processus RCRAFT : Modélisation RCRAFT 259

4 Comparaison des automatisations 259

4.1 Comparaison de l'allocation des tâches et des fonctions 260
4.2 Comparaison de l'allocation des ressources 260
4.3 Comparaison de l'allocation des transitions de contrôle et de l'initiative 260
4.4 Comparaison de l'allocation de la responsabilité 261
4.5 Comparaison de l'allocation de l'autorité 261

5 Conclusion 261

Chapitre 10. Conclusion et Perspectives 264

Publications personnelles 267

Références 269

Résumé 289

Abstract 290

Annexes 291

1 Modèles de tâches ségrégués et de fonctions pour la gestion des alarmes avec le FWS 291

1.1 Modèles de tâches ségrégués du PM 291
1.2 Modèles de fonctions du FWS 297

2 Colonnes du tableau RCRA décrivant l'allocation des ressources pour la gestion des alarmes avec le FWS 299

3 Modèles de tâches et de fonctions pour la gestion des alarmes avec le système alternatif de gestion des alarmes 303

3.1 Modèles de tâches ségrégués du PM 303
3.2 Modèles de fonctions du Recommandeur 304

4 Colonnes du tableau RCRA décrivant l'allocation des ressources pour la gestion des alarmes avec le système alternatif de gestion des alarmes 307

5 Meta-modélisation des modèles de tâches ségrégués et des modèles de fonctions 311

5.1 Meta-modélisation des modèles de tâches ségrégués 311
5.2 Meta-modélisation des modèles de fonctions 312
5.3 Meta-modélisation du protocole de coopération 313

6 ERRATA 314

Liste des figures

FIGURE 1. BOUCLE DE CONTROLE PERCEPTION-ACTION DE (FRANK FLEMISCH ET AL., 2012) 19
FIGURE 2. REPRESENTATION FONCTIONNELLE D'UN SYSTEME INTERACTIF PROPOSEE PAR (BASTIDE & PALANQUE, 1999). A) REPRESENTATION DES ENTREES ET SORTIES ENTRELACEES DES SYSTEMES INTERACTIFS. B) REPRESENTATION DE LA BOUCLE ENTREE-SORTIE DES SYSTEMES INTERACTIFS 25
FIGURE 3. ARCHITECTURE SEEHEIM (PFAFF, 1985) 26
FIGURE 4. ARCHITECTURE MVC D'APRÈS (KRASNER & POPE, 1988) 27
FIGURE 5. PROCESSUS DE CONCEPTION CENTRE UTILISATEUR PROPOSE PAR LA NORME ISO 9241 (INTERNATIONAL STANDARD ORGANIZATION, 2019) 28
FIGURE 6. PROPRIETES EXTERNES ET INTERNES D'UN SYSTEME INTERACTIFS PROPOSEES PAR (COCKTON & GRAM, 1996). FIGURE EXTRAITE DE (MARTINIE DE ALMEIDA, 2011). 29
FIGURE 7. ARBRE DE FIABILITE ET SECURITE D'APRES LES TRAVAUX DE (AVIZIENIS ET AL., 2004) 32
FIGURE 8. COCKPIT DE L'A350. L'ECAM EST ENCADRE EN ROUGE. 33
FIGURE 9. DESCRIPTION DE L'ECP ("ECAM CONTROL PANEL") 33
FIGURE 10. PROCESSUS DE DEVELOPPEMENT LOGICIEL DE LA NORME DO-178C. 35
FIGURE 11. CS-25.1302 (EXTRAIT DE LA NORME CS-25 DE L'EASA) 36
FIGURE 12. EXTRAIT DE LA CS-25 LIVRE 1 - ANNEXE D CRITERES POUR DETERMINER L'EQUIPAGE MINIMUM 37
FIGURE 13. EXTRAIT DE LA CS-25 TRAITANT DES RECOMMANDATIONS LES CONTROLES POUR LES SYSTEMES CONTENANT DE L'AUTOMATISATION 38
FIGURE 14. EXTRAIT DE LA CS-25 TRAITANT DES RECOMMANDATIONS POUR LES AFFICHAGES DES SYSTEMES CONTENANT DE L'AUTOMATISATION 38
FIGURE 15. MODELE DE LA *SITUATION AWARENESS* DANS LA PRISE DE DECISION DE (ENDSLEY, 1995) 43
FIGURE 16. NIVEAUX D'AUTOMATISATION DE (THOMAS B. SHERIDAN & VERPLANK, 1978) POUR LA PRISE DE DECISION POUR UNE SEULE ETAPE ELEMENTAIRE DE LA DECISION. 45
FIGURE 17. 10 NIVEAUX D'AUTOMATISATION D'APRES LES TRAVAUX DE (R. PARASURAMAN ET AL., 2000) 46
FIGURE 18. LES 10 NIVEAUX D'AUTOMATISATION DE (ENDSLEY, 1999). 47
FIGURE 19. NIVEAUX D'AUTOMATISATION PROPOSES PAR (ENDSLEY, 2016) 48
FIGURE 20. NIVEAUX D'AUTOMATISATION D'APRES LES TRAVAUX DE (BILLINGS, 1991) 49
FIGURE 21. NIVEAUX D'AUTOMATISATION DE LA CONDUITE DE LA SAE J3016 51
FIGURE 22. PROCESSUS D'ALLOCATION DE FONCTIONS ET DE TACHES PROPOSE PAR (ANDY DEARDEN ET AL., 2000) 53
FIGURE 23. LE TEMPLATE IDA-S 54
FIGURE 24. ILLUSTRATIONS DE LA METAPHORE DU CHEVAL UTILISE POUR CONCEVOIR LES CONTROLES PHYSIQUES D'UNE VOITURE D'APRES LES TRAVAUX DE (F. FLEMISCH ET AL., 2005). A) SERRER LES RENES POUR CONTROLER LE CHEVAL. B) RELACHER LES RENES POUR DONNER LE CONTROLE AU CHEVAL. 54
FIGURE 25. ARBRE DE CLASSIFICATION DES TRANSITIONS DE CONTROLE D'APRES LES TRAVAUX DE (LU ET AL., 2016) 56
FIGURE 26. MODELE GMOC D'APRES LES TRAVAUX DE (JANSSON ET AL., 2014) 58
FIGURE 27. ILLUSTRATION SIMPLIFIEE DES TROIS NIVEAUX DE PERFORMANCE D'UTILISATEURS QUALIFIES D'APRES LES TRAVAUX DE (RASMUSSEN, 1983) 61
FIGURE 28. LE MODELE DU PROCESSEUR HUMAIN D'APRES LES TRAVAUX DE (S. CARD ET AL., 1986) 62
FIGURE 29. LES SEPT ETAPES DE LA THEORIE DE L'ACTION (NORMAN, 2013) 63
FIGURE 30. MODEL DU PROCESSUS DANS L'ESPACE D'ETAT TASK-AGENT-CONTROL (TAC) 64
FIGURE 31. MODELE SIMPLE EN 4 ETAPES DU TRAITEMENT DE L'INFORMATION PAR L'HUMAIN 66
FIGURE 32. MODELE SIMPLE EN 4 ETAPES DU TRAITEMENT DE L'INFORMATION PAR LE SYSTEME. 66
FIGURE 33. MODELE DE TACHES HTA POUR EFFECTUER LE CONTROLE DE SERVICE D'UNE LIGNE DE METRO TIRE DE (SHEPHERD, 2002) 71
FIGURE 34. ETAPES DU MODELE DE (R. PARASURAMAN ET AL., 2000) ET LEUR REPRESENTATION DANS LA NOTATION HAMSTERS 81

FIGURE 35. MODELE SIMPLE EN QUATRE ETAPES DU TRAITEMENT SYSTEME DE L'INFORMATION DE (R. PARASURAMAN ET AL., 2000) ET LEUR REPRESENTATION DANS LA NOTATION HAMSTERS 81
FIGURE 36. EXEMPLE DE COMPOSITION D'OPERATEURS TEMPORELS..... 85
FIGURE 37. REPRESENTATIONS DE TACHES ABSTRAITES A) OPTIONNELLE B) ITERATIVE C) OPTIONNELLE ET ITERATIVE..... 85
FIGURE 38. REPRESENTATION GRAPHIQUE DE LA *COPY TASK*. 86
FIGURE 39. REPRESENTATION D'UN COMPOSANT DANS HAMSTERS 87
FIGURE 40. COMPOSANT AVEC LE PARAMETRE GENERIQUE "JETON" 87
FIGURE 41. COMPOSANT LE PARAMETRE "JETON" INSTANCIE AVEC LA VALEUR 3. 88
FIGURE 42. REPRESENTATION DES DONNEES, OBJETS ET PERIPHERIQUES (DOD) DANS HAMSTERS..... 88
FIGURE 43. EXEMPLE DE MODELISATION D'UNE CONNAISSANCE SITUATIONNELLE DANS HAMSTERS..... 89
FIGURE 44. EXEMPLE DE MODELISATION D'UNE CONNAISSANCE STRATEGIQUE DANS HAMSTERS. 89
FIGURE 45. EXEMPLE D'UTILISATION DU DOD DUREE..... 90
FIGURE 46. VUE GLOBALE DE L'EDITEUR DE TACHES HAMSTERS..... 93
FIGURE 47. COPIE D'ECRAN D'UNE SIMULATION DE MODELES DE TACHES AVEC L'OUTIL LOGICIEL HAMSTERS (EXEMPLE DE TACHES EFFECTUEES ET ACTIVES) 95
FIGURE 48. CONCEPT MAP DU JEU DES 15..... 100
FIGURE 49. UN DISTRIBUTEUR AUTOMATIQUE DE BILLET..... 124
FIGURE 50. MODELE DE TACHES INTEGRE DU CLIENT POUR RETIRER DE L'ARGENT AU DISTRIBUTEUR DE BILLETS 125
FIGURE 51. MODELE DE TACHES SEGREGUE DU CLIENT POUR RETIRER DE L'ARGENT..... 127
FIGURE 52. MODELE DE FONCTIONS DU DISTRIBUTEUR POUR DISTRIBUER DES BILLETS. 127
FIGURE 53. REPRESENTATION DES DONNEES, OBJETS ET PERIPHERIQUES (DOD) DANS HAMSTERS..... 133
FIGURE 54. REPRESENTATION DE DODS PERSONNALISES POUR LA REPRESENTATION DE RESSOURCES EMISES OU REÇUES LORS DE LA COOPERATION..... 133
FIGURE 55. SYMBOLE REPRESENTANT UNE TRANSITION DE CONTROLE. 133
FIGURE 56. SYMBOLE REPRESENTANT UNE TRANSITION DE CONTROLE ASSOCIE A A) UNE FONCTION ET A B) UNE TACHE..... 134
FIGURE 57. A) TACHE « ATTENDRE LES BILLETS » INTERROMPUE PAR L'AFFICHAGE DU « TEXTE INDIQUANT DE RECUPERER LA CARTE DE CREDIT » EXECUTE PAR B) LA FONCTION « AFFICHER « RECUPERER VOTRE CARTE » ET RENDRE LA CARTE » DU DISTRIBUTEUR DE BILLET 134
FIGURE 58. SYMBOLE REPRESENTANT LA RESPONSABILITE. 135
FIGURE 59. SYMBOLE DE LA RESPONSABILITE ASSOCIE A A) UNE FONCTION ET A B) UNE TACHE 135
FIGURE 60. SYMBOLE REPRESENTANT L'AUTORITE. 135
FIGURE 61. SYMBOLE DE L'AUTORITE ASSOCIE A A) UNE FONCTION ET A B) UNE TACHE 135
FIGURE 62. VUE HAUT NIVEAU DU PROCESSUS RCRAFT..... 138
FIGURE 63. VUE DETAILLEE DE LA PHASE DE MODELISATION DES TACHES EXPLICITANT L'ALLOCATION DE FONCTIONS ET DE TACHES. 141
FIGURE 64. VUE DETAILLEE DE LA SOUS-PHASE D'EXTRACTION DES DODS. 143
FIGURE 65. VUE DETAILLEE DE LA SOUS-PHASE DE DESCRIPTION DE L'ALLOCATION DE RESSOURCES..... 144
FIGURE 66. VUE DETAILLEE DE LA SOUS-PHASE DE DESCRIPTION DE L'ALLOCATION DES TRANSITIONS DE CONTROLE ET DE L'INITIATIVE 145
FIGURE 67. VUE DETAILLEE DE LA SOUS-PHASE DE DESCRIPTION DE L'ALLOCATION DE LA RESPONSABILITE. 147
FIGURE 68. VUE DETAILLEE DE LA SOUS-PHASE DE DESCRIPTION DE L'ALLOCATION DE L'AUTORITE 149
FIGURE 69. VUE DETAILLEE DE LA PHASE 3 DU PROCESSUS RCRAFT. 150
FIGURE 70. VUE DETAILLEE DU SOUS-PROCESSUS DE MODELISATION DES TACHES EXPLICITANT L'ALLOCATION DE FONCTIONS ET DE TACHES AVEC LES ETAPES EXPLICITEES NUMEROTEES DE 1 A 7 DANS DES PASTILLES BLEUES. 151
FIGURE 71. MODELE DE TACHES SEGREGUE DU ROLE JOUEUR POUR JOUER AU JEU DES 15 153
FIGURE 72. MODELE DE TACHES SEGREGUE DU ROLE LEADER POUR GAGNER LE JEU 154
FIGURE 73. MODELE DE TACHES SEGREGUE DU ROLE CHALLENGER POUR GAGNER LE JEU 156
FIGURE 74. MODELE DE FONCTIONS DU ROLE ARBITRE POUR FAIRE RESPECTER LES REGLES DU JEU DES 15 156
FIGURE 75. VUE DETAILLEE DE LA SOUS-PHASE D'EXTRACTION DES DODS AVEC LES ETAPES EXPLICITEES NUMEROTEES DE 1 ET 2 DANS DES PASTILLES BLEUES..... 157
FIGURE 76. COUCHE INTERFACE "WIDGET DES JETONS DISPONIBLES"..... 158

Figure 77. Vue detaillee de la sous-phase de description de l'allocation des ressources avec les etapes explicitees numerotees de 1 a 5 dans des pastilles bleues. 158
Figure 78. Vue detaillee de la sous-phase l'allocation des transitions de controle et de l'initiative avec les etapes explicitees numerotees de 1 a 4 dans des pastilles bleues. 162
Figure 79. Vue detaillee de la sous-phase de description de la responsabilite avec les etapes explicitees numerotees de 1 a 6 dans des pastilles bleues. 165
Figure 80. Vue detaillee de la sous-phase de description de l'autorite avec les etapes explicitees numerotees de 1 a 5 dans des pastilles bleues. 168
Figure 81. Vue detaillee de la phase de la modelisation RCRAFT avec les etapes explicitees numerotees de 1 a 2 dans des pastilles bleues 171
Figure 82. Modele de taches segregue du role Joueur pour jouer une partie du jeu des 15. 172
Figure 83. Modele de taches segregue du Leader pour gagner le jeu des 15. 173
Figure 84. Modele de taches segregue du Challenger pour gagner le jeu des 15. 174
Figure 85. Modele de fonctions de l'arbitre pour soutenir la partie dans le respect des regles du jeu des 15. 175
Figure 86. Tache "Finaliser de retirer l'argent" du role Client. 178
Figure 87. Fonction "Finaliser la transaction" du role distributeur de billet. 178
Figure 88. Propriete de communication du protocole de cooperation. 178
Figure 89. Propriete de localisation du protocole de cooperation. 179
Figure 90. Edition de la transition de controle initiee par la tache motrice "Selectionner un jeton" du Leader. 180
Figure 91. Edition de la responsabilite de la tache motrice "Selectionner un jeton" du Leader. 181
Figure 92. Edition de l'autorite de la tache cognitive "Trouver des alternatives" du Leader. 181
Figure 93. Aperçu du modele de taches du Leader pour gagner le jeu des 15 avec un temps limite pour jouer son tour 187
Figure 94. Aperçu du modele de fonctions de l'Arbitre pour soutenir la partie du jeu des 15 avec un temps limite pour le tour des joueurs. 189
Figure 95. Modele de taches segregues du Leader pour gagner le jeu des 15. 200
Figure 96. Modele de fonctions de l'arbitre pour soutenir la partie dans le respect des regles du jeu des 15 201
Figure 97. Tache « Voir le jeu » du modele de taches segregue du Leader pour gagner le jeu des 15 modifiee dans cette version du jeu des 15. 207
Figure 98. Tache « Trouver un jeton a jouer » du modele de taches segregue du Leader pour gagner le jeu des 15 modifiee dans cette version du jeu des 15 207
Figure 99. Tache « Attendre la fin de tour du Challenger » du modele de taches segregue du Leader pour gagner le jeu des 15 modifiee dans cette version du jeu des 15. 208
Figure 100. Modele de fonctions du recommandeur pour recommander des jetons aux joueurs. 209
Figure 101. Modele de fonctions du recommandeur pour recommander des jetons aux joueurs. 218
Figure 102. Synthese des comparaisons RCRAFT des differentes versions du jeu des 15. 222
Figure 103. Cockpit de l'A350. L'ECAM est encadre en rouge. 226
Figure 105. b. Architecture de l'ECAM - Lien du FWS avec d'autres systemes du cockpit 227
Figure 105. a. Filtrage effectué par le *Flight Warning System* 227
Figure 106. Les boutons MASTER Warning (a gauche) et MASTER Caution (a droite) allumes. 227
Figure 107. Procedure ALL ENGs FLAME OUT affichee sur le WD. 228
Figure 108. Description de l'ECP ("ECAM Control Panel") 230
Figure 109. Le panneau plafond a gauche et son emplacement dans le cockpit de l'A350. 231
Figure 110. Concept map de la gestion des alarmes. 232
Figure 111. Vue haut-niveau du processus RCRAFT avec les artefacts modeles de taches et de fonctions et la phase 1 mis en valeur. 232
Figure 112. Modele de fonctions haut-niveau du FWS pour gerer les alarmes 234
Figure 113. Modele de taches segregue haut-niveau du pilote monitoring pour resoudre les alarmes et ramener l'avion dans un etat fonctionnel 235
Figure 114. Vue haut-niveau du processus RCRAFT avec l'artefact : tableau RCRA et la phase 2 mis en valeur 236

FIGURE 115. PROTOTYPE DE SYSTEME ALTERNATIF DE GESTION DES ALARMES. 246
FIGURE 116. CONCEPT MAP DE LA GESTION DES ALARMES AVEC LE SYSTEME ALTERNATIF DE GESTION DES ALARMES. 247
FIGURE 117. VUE HAUT-NIVEAU DU PROCESSUS RCRAFT AVEC LES ARTEFACTS MODELES DE TACHES SEGREGUES ET DE FONCTIONS ET LA PHASE 1 MIS EN VALEUR. 248
FIGURE 118. MODELE DE FONCTIONS DU RECOMMANDEUR POUR GERER LES ALARMES. 249
FIGURE 119. MODELE DE TACHES DU PM POUR RESOUDRE LES ALARMES AVEC LE PROTOTYPE DE RECOMMANDEUR. 251
FIGURE 120. VUE HAUT-NIVEAU DU PROCESSUS RCRAFT AVEC L'ARTEFACT : TABLEAU RCRA ET LA PHASE 2 MIS EN VALEUR. 252
FIGURE 121. MODELE DE TACHES SEGREGUE DU PM : ACQUITTER LES ALARMES. 292
FIGURE 122. MODELE DE TACHES SEGREGUE DU PM : CONSULTER LA PROCEDURE ET LES ALARMES 293
FIGURE 123. MODELE DE TACHES SEGREGUE DU PM : ACCEDER A LA PROCEDURE PERTINENTE 294
FIGURE 124. MODELE DE TACHES SEGREGUE DU PM : REALISER LA PREMIERE PROCEDURE. 294
FIGURE 125. MODELE DE TACHES SEGREGUE DU PM : CONSULTER LA PAGE MORE. 295
FIGURE 126. MODELE DE TACHES SEGREGUE DU PM : CONSULTER UNE PAGE SD 295
FIGURE 127. MODELE DE TACHES SEGREGUE DU PM : RAPPELER TOUTES (RECALL ALL) LES PROCEDURES EFFACEES 295
FIGURE 128. MODELE DE TACHES SEGREGUE DU PM : RAPPELER LA DERNIERE PROCEDURE EFFACEE. 296
FIGURE 129. MODELE DE TACHES SEGREGUE DU PM : EMER CANCEL (SUPPRIMER) UNE PROCEDURE 296
FIGURE 130. MODELE DE TACHES SEGREGUE DU PM : CLEAR (EFFACER) UNE PROCEDURE TERMINEE 297
FIGURE 131. MODELE DE FONCTIONS DU FWS : RECEVOIR UNE NOUVELLE ALARME. 297
FIGURE 132. MODELE DE FONCTIONS DU FWS : AFFICHER LES ALARMES. 298
FIGURE 133. MODELE DE FONCTIONS DU FWS : SOUTENIR L'ACQUITTEMENT DES ALARMES. 298
FIGURE 134. MODELE DE FONCTIONS DU FWS : SOUTENIR LA SELECTION D'UNE PROCEDURE (DIFFERENTE DE LA PREMIERE PROCEDURE) 298
FIGURE 135. MODELE DE FONCTIONS DU FWS : SOUTENIR LA REALISATION DE LA PREMIERE PROCEDURE. 299
FIGURE 136. MODELE DE TACHES SEGREGUE DU PM : ACQUITTER LES ALARMES 303
FIGURE 137. MODELE DE TACHES SEGREGUE DU PM : CHOISIR UNE RECOMMANDATION A REALISER. 303
FIGURE 138.MODELE DE TACHES SEGREGUE DU PM : REALISER LA RECOMMANDATION SELECTIONNEE. 304
FIGURE 139. MODELE DE FONCTIONS DU RECOMMANDEUR : TRAITER L'ARRIVEE D'UNE ALARME. 304
FIGURE 140. MODELE DE FONCTIONS DU RECOMMANDEUR : AFFICHER LES RECOMMANDATIONS. 305
FIGURE 141. MODELE DE FONCTION DU RECOMMANDEUR POUR SOUTENIR L'ACQUITTEMENT DES ALARMES. 305
FIGURE 142. MODELE DE FONCTIONS DU RECOMMANDEUR : SOUTENIR LA SELECTION D'UNE RECOMMANDATION. 306
FIGURE 143. MODELE DE FONCTIONS DU RECOMMANDEUR : SOUTENIR LA REALISATION D'UNE RECOMMANDATION. 306
FIGURE 144. META-MODELE D'UN MODELE DE TACHES SEGREGUE. 311
FIGURE 145. META-MODELE D'UN MODELE DE FONCTIONS. 312
FIGURE 146. META-MODELE D'UN PROTOCOLE DE COOPERATION. 313

Liste des tables

TABLEAU 1. NIVEAUX DE CRITICITE, DAL ET OBJECTIFS EN TERMES DE PROBABILITE D'OCCURRENCE..... 35
TABLEAU 2. LA LISTE MABA MABA DE (PAUL M. FITTS, 1951)..... 44
TABLEAU 3. MATRICE ESPACE-TEMPS DU COLLECTICIEL..... 67
TABLEAU 4. COMPARAISON D'UN ECHANTILLON REPRESENTATIF DE NOTATIONS DE MODELISATION DE TACHES UTILISATEURS ETENDANT LE TRAVAIL DE (MARTINIE DE ALMEIDA, 2011)..... 76
TABLEAU 5. TYPES DE TACHES ET FONCTIONS GENERIQUES DE LA NOTATION HAMSTERS..... 79
TABLEAU 6. TYPES DE TACHES UTILISATEUR..... 80
TABLEAU 7. TACHES INTERACTIVES D'ENTREE/SORTIE DE LA NOTATION HAMSTERS..... 83
TABLEAU 8. OPERATEURS D'ORDONNANCEMENT TEMPOREL UTILISES PAR LA NOTATION HAMSTERS..... 83
TABLEAU 9. TYPES DE *SUB-ROUTINES* DANS LA NOTATION HAMSTERS..... 86
TABLEAU 10. TABLEAU DES RELATIONS ENTRE TACHES ET DOD..... 90
TABLEAU 11. JEU DES 15 UTILISE A DES FINS D'ILLUSTRATION DU CADRE CONCEPTUEL RCRAFT..... 99
TABLEAU 12. DISTRIBUTION DES TACHES ET DES FONCTIONS POUR LE JEU DES 15..... 103
TABLEAU 13. DISTRIBUTION DES RESSOURCES POUR LE JEU DES 15..... 108
TABLEAU 14. DESCRIPTION DE L'ALLOCATION DE L'AUTORITE..... 112
TABLEAU 15. DISTRIBUTION DES TRANSITIONS DE CONTROLE ET DE L'INITIATIVE..... 116
TABLEAU 16. DISTRIBUTION DE LA RESPONSABILITE POUR LE JEU DES 15 CLASSIQUE #1 ET LE JEU DES 15 ANNULATION SELECTION #5. 118
TABLEAU 17. STRUCTURE D'UN PROTOCOLE DE COOPERATION..... 130
TABLEAU 18. EXEMPLES DE TACHES ET FONCTIONS COOPERATIVES AVEC LA NOTATION HAMSTERS..... 130
TABLEAU 19. OPERATEURS D'ORDONNANCEMENT TEMPOREL UTILISES PAR LA NOTATION HAMSTERS..... 134
TABLEAU 20. TABLEAU RCRA..... 140
TABLEAU 21. COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DES RESSOURCES..... 145
TABLEAU 22. COLONNES DU TABLEAU RCRA DECRIVANT LES TRANSITIONS DE CONTROLE ET L'ALLOCATION DE L'INITIATIVE..... 146
TABLEAU 23. COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DE LA RESPONSABILITE..... 148
TABLEAU 24. COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DE L'AUTORITE..... 150
TABLEAU 25. COLONNES DE LA PARTIE « ALLOCATION DES RESSOURCES » DU TABLEAU RCRA REMPLI APRES LES ETAPES DECRITES DANS CETTE SECTION..... 159
TABLEAU 26. COLONNES DE LA PARTIE « ALLOCATION DES TRANSITIONS DE CONTROLE ET DE L'INITIATIVE » DU TABLEAU RCRA COMPLETE DURANT LES ETAPES 1 A 4 DECRITES CI-DESSUS..... 163
TABLEAU 27. COLONNES DE LA PARTIE « ALLOCATION DE LA RESPONSABILITE » DU TABLEAU RCRA COMPLETEES DURANT LES ETAPES DECRITES CI-DESSUS..... 166
TABLEAU 28. COLONNES DE LA PARTIE « ALLOCATION DES RESSOURCES » DU TABLEAU RCRA COMPLETEES DURANT LES ETAPES DECRITES CI-DESSUS..... 169
TABLEAU 29. PRESENTATION DU JEU DES 15 AVEC UN TEMPS LIMITE POUR JOUER SON TOUR..... 185
TABLEAU 30. DESCRIPTION DE L'ALLOCATION DES RESSOURCES POUR LE JEU DES 15 AVEC UN TEMPS LIMITE POUR JOUER SON TOUR 191
TABLEAU 31. DESCRIPTION DE L'ALLOCATION DES TRANSITIONS DE CONTROLE ET DE L'INITIATIVE AVEC UN TEMPS LIMITE POUR JOUER SON TOUR..... 193
TABLEAU 32. DESCRIPTION DE L'ALLOCATION DE LA RESPONSABILITE POUR LE JEU DES 15 AVEC UN TEMPS LIMITE POUR JOUER SON TOUR..... 196
TABLEAU 33. DESCRIPTION DE L'ALLOCATION DE L'AUTORITE AVEC UN TEMPS LIMITE POUR JOUER SON TOUR..... 198
TABLEAU 34. DESCRIPTION DE L'ALLOCATION DES RESSOURCES POUR LE JEU DES 15 PROPOSANT DES RECOMMANDATIONS AUX JOUEURS..... 210

TABLEAU 35. DESCRIPTION DE L'ALLOCATION DES TRANSITIONS DE CONTROLE ET DE L'INITIATIVE PROPOSANT DES RECOMMANDATIONS AUX JOUEURS..... 212
TABLEAU 36. DESCRIPTION DE L'ALLOCATION DE LA RESPONSABILITE POUR LE JEU DES 15 PROPOSANT DES RECOMMANDATIONS AUX JOUEURS..... 215
TABLEAU 37. DESCRIPTION DE L'ALLOCATION DE L'AUTORITE PROPOSANT DES RECOMMANDATIONS AUX JOUEURS 216
TABLEAU 38. EXTRAIT DES COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DES RESSOURCES ENTRE LE FWS ET LE PM POUR LA RESOLUTION DES ALARMES. 236
TABLEAU 39. COLONNES DU TABLEAU DECRIVANT LES TRANSITIONS DE CONTROLE ET L'ALLOCATION DE L'INITIATIVE ENTRE LE FWS ET LE PM POUR LA RESOLUTION DES ALARMES. 238
TABLEAU 40. COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DE LA RESPONSABILITE ENTRE LE FWS ET LE PM POUR LA RESOLUTION DES ALARMES. 240
TABLEAU 41. COLONNES DU TABLEAU DECRIVANT L'ALLOCATION DE L'AUTORITE ENTRE LE FWS ET LE PM POUR LA RESOLUTION DES ALARMES. 242
TABLEAU 42. EXTRAIT DES COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DES RESSOURCES ENTRE LE RECOMMANDEUR (RECOMMANDEUR) ET LE PILOTE *MONITORING* (PM) POUR LA RESOLUTION DES ALARMES..... 252
TABLEAU 43. COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DES TRANSITIONS DE CONTROLE ET DE L'INITIATIVE ENTRE LE RECOMMANDEUR (RECOMMANDEUR) ET LE PILOTE MONITORING (PM)..... 254
TABLEAU 44. COLONNES DU TABLEAU DECRIVANT L'ALLOCATION DE LA RESPONSABILITE ENTRE LE RECOMMANDEUR (RECOMMANDEUR) ET LE PILOTE MONITORING (PM) 256
TABLEAU 45. COLONNES DU TABLEAU RCRA DECRIVANT L'ALLOCATION DE L'AUTORITE ENTRE LE RECOMMANDEUR (RECOMMANDEUR) ET LE PILOTE MONITORING (PM) 258
TABLEAU 46. TABLEAU DECRIVANT L'ALLOCATION DES RESSOURCES ENTRE LE FWS ET LE PM POUR LA RESOLUTION DES ALARMES. 299
TABLEAU 47. TABLEAU DECRIVANT L'ALLOCATION DES RESSOURCES ENTRE LE SYSTEME ALTERNATIF DE GESTION DES ALARMES (RECOMMANDEUR) ET LE PILOTE *MONITORING* (PM) POUR LA RESOLUTION DES ALARMES..... 307

Introduction

L'automatisation permet aux systèmes interactifs d'exécuter des fonctions sans intervention extérieure de la part d'un utilisateur. L'automatisation permet également d'adapter les fonctions que ces systèmes exécutent à de nouveaux contextes ou buts utilisateur. L'automatisation a été introduite dans les systèmes interactifs afin d'améliorer les performances des utilisateurs les utilisant. Nous pouvons citer des exemples de tâches quotidiennes soutenues par des systèmes interactifs contenant de l'automatisation comme par exemple, faire du café avec une cafetière, faire la vaisselle avec un lave-vaisselle ou regarder des épisodes d'une série sur une plateforme de streaming. Également, nous pouvons citer des tâches plus complexes comme les tâches permettant à un avion civil de se déplacer d'un aéroport A à un aéroport B. Ces tâches, auparavant, réparties entre quatre pilotes dans le cockpit peuvent, désormais, être réalisées par deux pilotes grâce à l'automatisation contenue dans le cockpit interactif d'avion civil et les autres systèmes avions.

1 Problématique

Cependant, cette automatisation peut parfois se révéler contre-productive et produire le paradoxe de l'automatisation (Bainbridge, 1983). Au lieu d'améliorer les performances des utilisateurs en réduisant la charge qui leur est allouée pour réaliser leurs tâches, les performances sont réduites par une charge supplémentaire revenant à l'utilisateur pour gérer cette automatisation. Ce problème peut se déclarer si l'utilisateur est tenu en dehors de la boucle de contrôle perception-action (illustrée dans la Figure 1) avec le système contenant de l'automatisation et doit soudainement reprendre le contrôle. Dans ce cas, les utilisateurs sont, à la fois, lents pour détecter qu'un problème avec l'automatisation existe et pour arriver à une compréhension suffisante du problème pour intervenir efficacement (Endsley & Kiris, 1995).

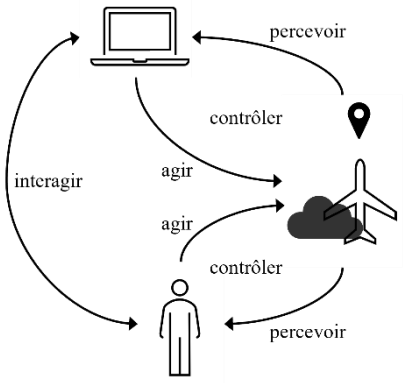


Figure 1. Boucle de contrôle perception-action de (Frank Flemisch et al., 2012)

De plus, certains systèmes contenant de l'automatisation sont capables d'adapter leur comportement à de nouvelles situations. Par exemple, lorsqu'une voiture passe du mode de conduite autonome au mode de conduite manuelle car le marquage au sol n'est pas assez entretenu, le conducteur devra prendre en charge la tâche de conduite. Si le système ne donne pas à l'utilisateur la capacité de suivre et d'anticiper les changements de comportement dus au

passage au mode de conduite manuelle, l'utilisateur devra faire face à ce que (Sarter et al., 1997) appellent des surprises d'automatisation. Afin d'éviter ces problèmes propres à l'introduction de l'automatisation dans les systèmes interactifs, des méthodes, techniques et outils sont utilisés pour comprendre et concevoir l'automatisation dans les systèmes interactifs.

En amont du cycle de conception de l'automatisation, l'allocation des fonctions et des tâches s'est d'abord appuyée sur des recommandations permettant d'identifier qui de l'utilisateur ou du système est le meilleur pour réaliser une tâche d'un certain type. Un premier type d'approche pour ces recommandations, appelée MABA MABA (Machine Are Better At – Man Are Better At), a été proposée par (Paul M. Fitts, 1951). Le MABA MABA indique, par exemple, que le système sera plus approprié pour faire de longs calculs alors que l'utilisateur sera plus approprié pour prendre des décisions. Par la suite, (R. Parasuraman et al., 2000) ont proposé un autre type d'approche : les 10 niveaux d'automatisation. Chacun de ces niveaux décrit une allocation de tâches et de fonctions différente entre l'utilisateur et le système : plus le niveau d'automatisation augmente, plus le nombre de tâches migrées de l'utilisateur vers le système est important. Ces types d'approche ont fait l'objet de critiques, notamment dans les travaux de (Dekker & Woods, 2002) qui rappellent que l'introduction de l'automatisation provoque l'allocation de nouvelles tâches à l'utilisateur pour gérer cette automatisation. Ils rappellent aussi que ces approches masquent l'aspect coopératif du couple utilisateur-système contenant de l'automatisation. De nouvelles approches, comme celle proposée par (Frank Flemisch et al., 2012), mettent en valeur le fait que la conception de l'automatisation doit aussi prendre en compte l'allocation de l'autorité et de la responsabilité. L'analyse de l'allocation de l'autorité doit permettre de déterminer qui exerce le contrôle pour une tâche ou une fonction mais aussi qui peut redistribuer le contrôle. L'autorité de redistribuer le contrôle permet à l'utilisateur ou au système contenant de l'automatisation qui la détient d'initier des transitions de contrôle. Enfin, l'analyse de l'allocation de ressources entre utilisateur et système doit aussi être prise en compte afin de pouvoir déterminer dans quelle mesure les utilisateurs pourront comprendre le comportement, les performances, et les futurs plans d'action du système. Ce concept appelé transparence est un élément du modèle HAT (*Human-Automation Teaming*) (Battiste et al., 2018).

Les approches existantes de conception des systèmes contenant de l'automatisation considèrent, à haut niveau et en amont dans le cycle de conception, certains ou l'ensemble de ces concepts d'allocation de fonctions, de tâches, d'autorité, de transition de contrôle, de responsabilité et de ressources qui sont nécessaires pour caractériser un système contenant de l'automatisation. Cependant, les approches existantes ne permettent pas de prendre en compte systématiquement, uniformément et de manière détaillée par rapport aux tâches utilisateur et au comportement du système interactif, l'analyse et la description de ces concepts.

2 Hypothèse de la thèse

Dans cette thèse notre objectif est de démontrer qu'il est possible de décrire de façon systématique l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité par rapport à allocation des fonctions et des tâches entre les utilisateurs et les systèmes contenant de l'automatisation.

Le contenu de cette thèse a pour objectif de démontrer qu'une approche **à base de modèles de tâches associée à un processus outillé de description de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité, de l'autorité, des fonctions et des tâches** permet d'atteindre ces objectifs.

3 Structure de la thèse

Cette thèse s'organise en 11 chapitres, chacun correspondant à une étape de l'argumentaire.

Le **Chapitre 1** présente les systèmes interactifs et les systèmes interactifs critiques faisant l'objet de cette thèse. Leurs particularités architecturales, les propriétés cibles pour ces systèmes et des processus pour garantir ces propriétés sont présentées. Ce chapitre démontre qu'il est nécessaire de prendre en compte les particularités architecturales et les propriétés propres à ces systèmes en utilisant des techniques et des processus les garantissant.

Le **Chapitre 2** présente l'automatisation contenue dans les systèmes interactifs. Les propriétés cibles pour ces systèmes et les approches pour les garantir sont également présentées. Ce chapitre démontre que pour prendre en compte les propriétés cibles de ces systèmes lors de leur conception, il est nécessaire d'analyser l'allocation des tâches et des fonctions, des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. En particulier, ce chapitre démontre que la description des tâches utilisateur et fonctions système est un prérequis à cette analyse.

Le **Chapitre 3** présente différents modèles de l'humain et de ses tâches. Également, un ensemble de notations pour la modélisation de tâches est présenté. Ce chapitre démontre que les notations et outils existants pour l'analyse et la description des tâches utilisateur ne permettent pas de décrire explicitement les allocations des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Ce chapitre présente de manière détaillée la notation et l'outil associé HAMSTERS qui ont été étendus dans le cadre des travaux de cette thèse.

Le **Chapitre 4** présente le cadre conceptuel RCRAFT pour la description de l'allocation des ressources, de transitions de contrôles, de la responsabilité et de l'autorité par rapport à l'allocation des fonctions et des tâches entre le ou les utilisateurs et le ou les systèmes contenant de l'automatisation. Chaque élément du cadre conceptuel est illustré sur l'exemple du jeu des 15 présenté dans ce chapitre. Cette contribution propose de décrire à travers un unique cadre conceptuel les concepts d'allocation des ressources, des transitions de contrôle, de la responsabilité et de l'autorité par rapport à l'allocation des tâches et des fonctions.

Le **Chapitre 5** présente les éléments de notation permettant la représentation des éléments du cadre conceptuel RCRAFT à base de modèles de tâches. Cette contribution propose de décrire explicitement les éléments RCRAFT dans une notation à base de modèles de tâches.

Le **Chapitre 6** présente un processus d'analyse et de description systématique des éléments RCRAFT en s'appuyant sur les modèles de tâches et de fonctions présentés au chapitre précédent.

Le **Chapitre 7** présente les extensions apportées à l'outil HAMSTERS pour la représentation des éléments du cadre conceptuel RCRAFT. Cette contribution propose des éléments de représentation explicite des éléments RCRAFT afin de soutenir le processus présenté au chapitre précédent.

Le **Chapitre 8** présente l'application du processus RCRAFT à différentes versions du jeu des 15. Ce chapitre a pour but de démontrer la faisabilité de l'approche proposée ainsi que ses bénéfices pour l'analyse de l'automatisation à travers la comparaison de trois versions du jeu des 15.

Le **Chapitre 9** présente l'application du processus RCRAFT à une étude de cas industrielle dans le domaine d'application de l'aéronautique. Le processus RCRAFT est appliqué à la gestion des alarmes avec le système actuel du cockpit d'avion A350 et à un système proposé proposant des recommandations. Ce chapitre a pour but de démontrer la faisabilité de l'approche ainsi que sa pertinence sur une étude de cas industrielle concrète.

Les **Chapitre 10** présente la conclusion de cette thèse et les perspectives de futurs travaux d'utilisation et d'extensions de l'approche proposée.

Partie I. Etat de l'art

Chapitre 1. Systèmes interactifs et systèmes interactifs critiques

Ce chapitre présente les caractéristiques et propriétés des systèmes interactifs et des systèmes critiques afin de :

- Caractériser les systèmes faisant l'objet des travaux de cette thèse.
- Présenter leurs propriétés et les problématiques liées à leur conception et à leur développement.

La première section décrit les caractéristiques principales des systèmes interactifs et, particulièrement, les propriétés d'utilisabilité et d'expérience utilisateur.

La seconde section décrit les caractéristiques principales des systèmes critiques et, particulièrement, les propriétés de sureté de fonctionnement et de sécurité. Nous présentons également un exemple de système interactif critique : le cockpit d'avion civil. Cette section introduit le prochain chapitre en présentant des recommandations pour la conception des systèmes interactifs contenant de l'automatisation dans les cockpits d'avion civil.

1 Systèmes interactifs

Dans cette sous-section, afin de comprendre ce qu'est un système interactif, nous présentons une définition des systèmes interactifs, puis nous présentons leurs caractéristiques architecturales, leurs propriétés et des processus permettant de garantir ces propriétés.

1.1 Définition d'un système interactif

Selon la norme ISO 9241-210 (International Standard Organization, 2019) :

> *Un système interactif est combinaison de matériels et/ou de logiciels et/ou de services et/ou de personnes avec laquelle interagissent des utilisateurs afin d'atteindre des objectifs spécifiques.*

En d'autres termes, un système interactif est un système informatisé réactif destiné à être utilisé par des utilisateurs. Un système réactif est un système en interaction permanente avec son environnement tout en produisant des sorties sur celui-ci (Pnueli, 1986). Les systèmes transformationnels, au contraire, traitent toutes les entrées au début de leur exécution puis fournissent les sorties à la fin de leur exécution. Un système interactif traite les entrées du ou des utilisateurs, et produit en sortie une représentation perceptible de son nouvel état interne au cours de son exécution. Par conséquent, les entrées et les sorties des systèmes interactifs sont entrelacées.

La Figure 2 présente deux représentations d'un système interactif selon (Bastide & Palanque, 1999). La Figure 2 a) représente l'entrelacement des entrées et des sorties des systèmes interactifs au cours de son exécution. La Figure 2 b) décrit qu'un système interactif prend en entrée les données de l'environnement et les évènements déclenchés par les utilisateurs par

l'intermédiaire de périphériques d'entrée. Ces entrées ne sont pas totalement prévisibles car nous considérons un utilisateur humain. Les entrées réalisées par le ou les utilisateurs dépendent des sorties produites par le système et de leur traitement par l'utilisateur. La Figure 2 b) met également en valeur que chaque entrée utilisateur doit déclencher une sortie du système interactif. Les sorties fournies par le système interactif sont exécutées selon le traitement des entrées fournies par l'utilisateur.

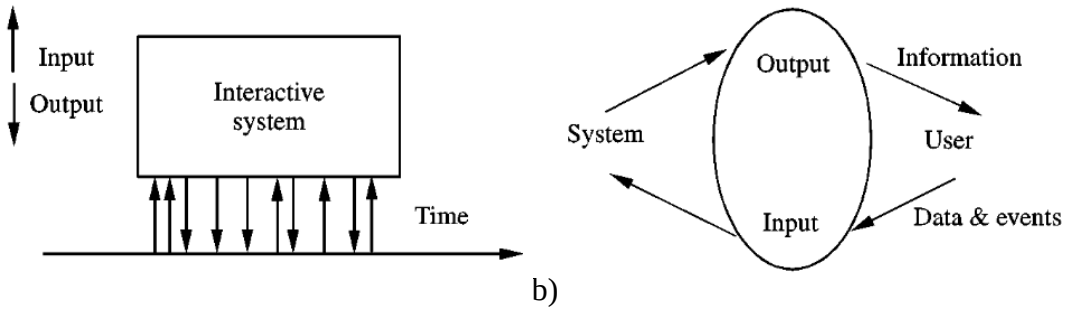


Figure 2. Représentation fonctionnelle d'un système interactif proposée par (Bastide & Palanque, 1999). a) représentation des entrées et sorties entrelacées des systèmes interactifs. b) représentation de la boucle entrée-sortie des systèmes interactifs.

1.2 Architectures des systèmes interactifs

Les architectures des systèmes interactifs peuvent se baser sur des modèles à couches (ou linguistiques) ou à agents.

1.2.1 Les modèles à couches[1]

Les modèles à couches définissent trois composantes logiques d'un système interactif :

- **Lexicale** : le vocabulaire d'entrée (par exemple, un clic) et de sortie (par exemple, des icônes).
- **Syntaxique** : la grammaire d'entrée correspondant aux séquences d'actions et de retours valides.
- **Sémantique** : la partie fonctionnelle du système.

Le modèle de Seeheim (Pfaff, 1985) présenté en Figure 3 est une architecture logicielle haut niveau pour les systèmes interactifs. Cette architecture structure les systèmes interactifs en trois composantes logiques correspondants aux couches logiques présentées ci-avant :

- La **présentation** : Ce composant permet de gérer l'interaction avec l'utilisateur. Les entrées de l'utilisateur sont interprétées par la couche présentation. La couche présentation génère les sorties perceptibles par l'utilisateur. La couche présentation est la composante lexicale du système interactif.
- Le **dialogue** : Ce composant se charge des états de l'application interactive et de l'impact des actions des utilisateurs sur l'interface utilisateur sur les changements de ces états. La couche dialogue est la composante syntaxique du système interactif.
- L'**interface du noyau fonctionnel** : Ce composant convertit les interactions d'entrée de l'utilisateur en appels de fonctions au noyau fonctionnel. Également, ce composant

[1] Modifications apportées post-publication

converti les retours du noyau fonctionnel en appel de fonction au dialogue. La couche interface du noyau fonctionnel est la composante sémantique du système interactif.

Le noyau fonctionnel gère toutes les fonctions non interactives de l'application comme l'accès aux bases de données par exemple. L'utilisateur perçoit les retours du système interactif et interagit avec le système interactif par le biais de la couche présentation.

Le modèle de Seeheim introduit un composant supplémentaire permettant de représenter le retour sémantique rapide rendu à l'utilisateur. Ce composant représente les modifications immédiates du rendu graphique de la couche présentation telles que la mise à jour de la position du curseur graphique.

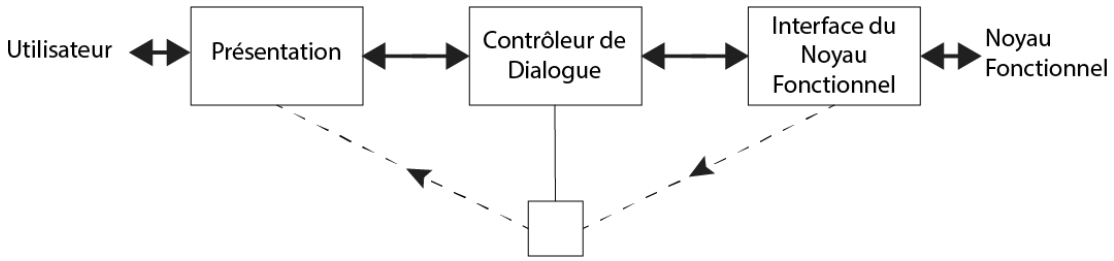


Figure 3. Architecture Seeheim (Pfaff, 1985)

D'autres architectures décrites par des modèles à couches ont été proposées. Parmi celles-ci, nous pouvons citer : l'architecture ARCH et le méta-modèle Slinky de (Bass et al., 1992), le modèle PIE et red-PIE de (Dix, 1991) ou MIODMITT de (Cronel et al., 2019).

1.2.2 Les modèles à agents

Les modèles à agent proposent une décomposition modulaire des systèmes interactifs.

L'architecture Modèle Vue Contrôleur (MVC) de (Krasner & Pope, 1988) présenté en Figure 4 décompose les systèmes interactifs en trois composants :

- Le modèle : Ce composant est la simulation logicielle spécifique au domaine ou l'implémentation de la structure de l'application. Le modèle modifie la vue associée à son état à chaque fois que celui-ci est modifié par le noyau fonctionnel de l'application ou par le contrôleur.
- La vue : Ce composant représente l'état du modèle perceptible par l'utilisateur. La vue est mise à jour lors des changements d'état du modèle.
- Le contrôleur : Ce composant reçoit et interprète les événements correspondants aux interactions de l'utilisateur. Le contrôleur modifie l'état du modèle en fonction des interactions utilisateur. Le contrôleur peut également demander des modifications immédiates du rendu graphique à la vue telle que la mise à jour de la position du curseur graphique.

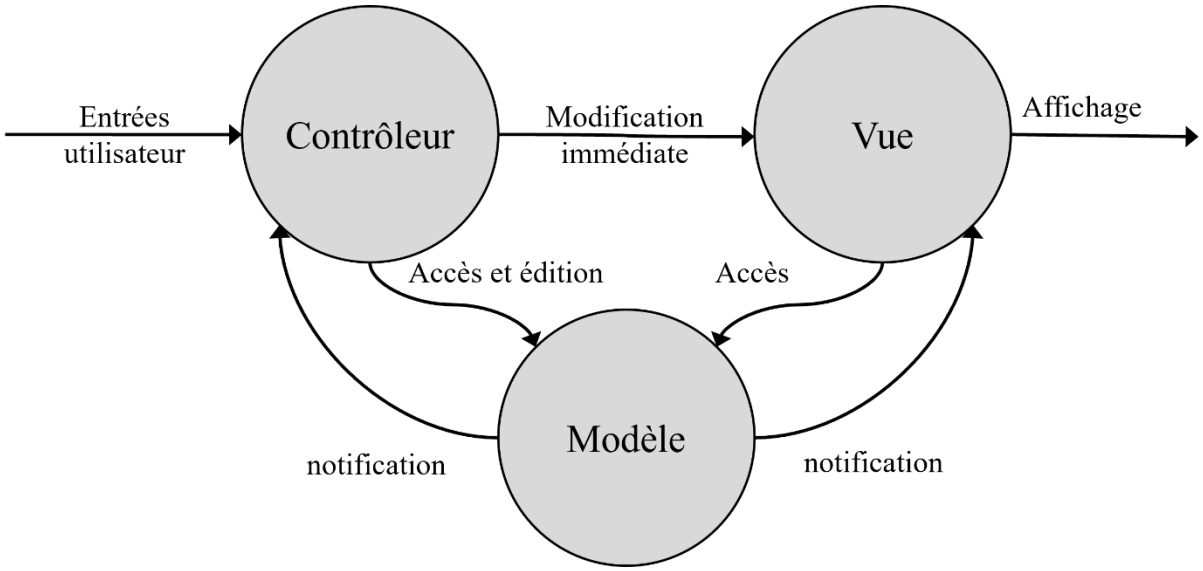


Figure 4. Architecture MVC d'après (Krasner & Pope, 1988)

De nombreuses variantes de l'architecture MVC ont été proposées comme celle proposée par (Buschmann et al., 2007), la version modifiée de MVC pour Java SE (Eckstein, 2007), l'architecture MVP (Modèle Vue Présentation) de (Potel, 1996), l'architecture MVVN (Modèle Vue VueModèle) de Fowler présentée dans (Sørensen & Mihailesc, 2010) ou l'architecture MDPC (Model Display view Picking view Controller) de (Conversy et al., 2008).

D'autres architectures des systèmes interactifs décrites par des modèles à agents ont été proposées. Parmi celles-ci, nous pouvons citer l'architecture PAC et PAC-Amodeus (Coutaz, 1987).

1.3 Propriétés des systèmes interactifs

Les propriétés propres aux systèmes interactifs permettent de mesurer la qualité de l'interaction entre l'utilisateur et le système interactif. Ces propriétés font l'objet de plusieurs travaux, standards et recommandations. Nous présentons dans les sous-sections suivantes les principales propriétés des systèmes interactifs : les propriétés externes et internes définies par (Cockton & Gram, 1996), l'utilisabilité et enfin l'expérience ou le ressenti utilisateur (User eXperience ou UX).

1.3.1 Utilisabilité

L'utilisabilité d'un système interactif est défini par la norme ISO 9241-210 (International Standard Organization, 2019) :

L'utilisabilité est le degré selon lequel un système, un produit ou un service peut être utilisé, par des utilisateurs spécifiés, pour atteindre des buts définis avec efficacité, efficience et satisfaction, dans un contexte d'utilisation spécifié.

- *L'efficacité est la précision et degré d'achèvement avec lesquels l'utilisateur atteint des objectifs spécifiés.*

- *L'efficience est le rapport entre les ressources utilisées (effort, coûts et matériaux) et les résultats obtenus.*
- *La satisfaction est le degré selon lequel les réactions physiques, cognitives et émotionnelles de l'utilisateur qui résultent de l'utilisation d'un système, produit ou service répondent aux besoins et attentes de l'utilisateur.*

L'utilisabilité est une propriété fondamentale car elle détermine la capacité du système interactif à remplir le service désiré par l'utilisateur. Les évaluations de l'utilisabilité appliquant une approche empirique font intervenir les utilisateurs à travers diverses techniques (questionnaires, observations, méthode de la pensée à voix haute…). Les évaluations de l'utilisabilité appliquant une approche analytique ne font intervenir que des experts qui utilisent leurs connaissances et des heuristiques. Les méthodes d'évaluation de l'utilisabilité basées sur les modèles sont un exemple d'une telle approche (Ladry et al., 2010). Ces évaluations sont intégrées dans les processus de conception et de développement centrés utilisateur. Les processus de conception centrée utilisateur ont pour objectif d'améliorer l'utilisabilité des systèmes interactifs. Figure 5 présente le processus de conception centré utilisateur proposé par la norme ISO 9241 (International Standard Organization, 2019). Les processus de conception centrée utilisateur sont itératifs. Ils proposent d'analyser les besoins et le comportement des utilisateurs, d'utiliser le prototypage et d'évaluer l'utilisabilité du système interactif conçu et développé.

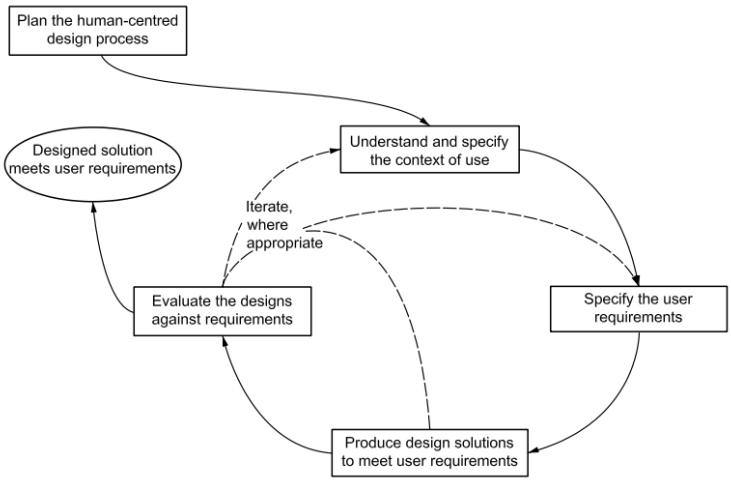


Figure 5. Processus de conception centré utilisateur proposé par la norme ISO 9241 (International Standard Organization, 2019)

1.3.2 Propriétés externes et internes de (Cockton & Gram, 1996)

Les travaux de (Cockton & Gram, 1996) définissent deux catégories de propriétés pour garantir la qualité des systèmes interactifs. La Figure 6 présente ces deux catégories de propriétés : les propriétés externes et internes du système interactif.

Propriétés externes

Cette première catégorie regroupe les propriétés externes caractérisant la qualité du système interactif selon l'utilisateur en termes de flexibilité et de robustesse de l'interaction. La flexibilité de l'interaction permet de qualifier la manière dont le système échange des informations avec l'utilisateur en termes d'entrées-sorties, de techniques d'interaction, ou de

son aspect dynamique par exemple. La robustesse de l'interaction permet de qualifier la manière dont le système permet à l'utilisateur d'accomplir ses tâches sans commettre d'erreurs irréversibles.

Propriétés internes

La seconde catégorie regroupe les propriétés internes d'un système interactifs. Les propriétés internes caractérisent la qualité du système interactif du point de vue de sa conception et de son développement. Ces propriétés concernent, par exemple, la capacité d'évolution, de maintenabilité ou de performance du système interactif. Ce type de propriétés doivent être prise en compte dès les phases conception et de développement du système interactif. Ainsi l'utilisation d'architectures propres aux systèmes interactifs (présentées en section 1.2) peut permettre de garantir leur prise en compte dès la phase de conception.

Le niveau de prise en considération de ces propriétés est décidé pendant la conception et le développement du système interactif.

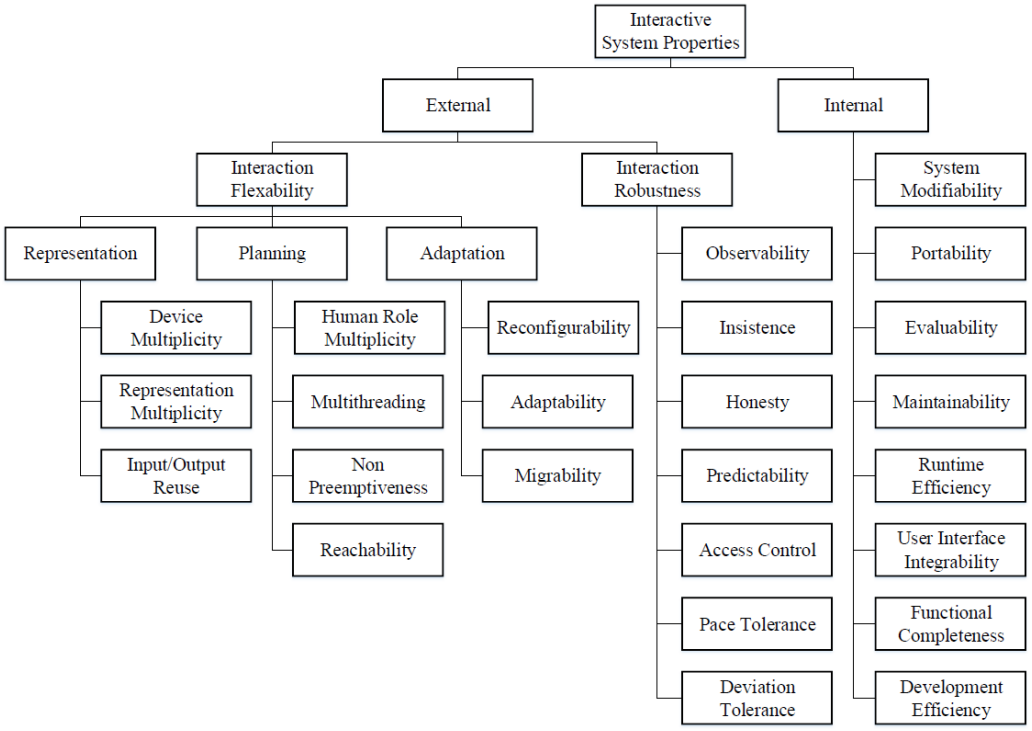


Figure 6. Propriétés externes et internes d'un système interactifs proposées par (Cockton & Gram, 1996). Figure extraite de (Martinie De Almeida, 2011).

1.3.3 Expérience ou Ressenti Utilisateur

L'expérience ou le ressenti utilisateur est défini par la norme ISO 9241-210 (International Standard Organization, 2019) :

> *L'expérience utilisateur sont les perceptions et réactions d'un utilisateur qui résultent de l'utilisation effective et/ou anticipée d'un système, produit ou service.*

Cette définition de l'expérience utilisateur n'identifie pas de critère mesurable de l'expérience utilisateur permettant son évaluation. Les travaux de (Winckler et al., 2013) identifie six dimensions pour mesurer l'expérience utilisateur d'une application mobile :

- Esthétique : La dimension esthétique implique la capacité d'un objet ou système à ravir une ou plusieurs de nos modalités sensorielles
- Emotion : Le niveau émotionnel implique les expériences généralement prises en compte dans la psychologie des émotions et dans le langage quotidien des émotions (par exemple, l'amour et la colère) qui sont suscitées par la signification relationnelle avec l'objet ou le système.
- Stimulation : La dimension de stimulation décrit dans quelle mesure un objet ou un système peut répondre au besoin humain de fonctions, d'interaction et de contenu innovants et intéressants.
- Identification : La dimension identification décrit la capacité d'un objet ou d'un système à communiquer des valeurs personnelles importantes aux personnes concernées.
- Signification et valeur : La dimension de signification implique la capacité de l'humain à attribuer une personnalité et à évaluer la signification personnelle ou symbolique des objets ou des systèmes. La valeur fait référence au concept selon lequel les objets ou les systèmes sont parfois choisis parce qu'ils reflètent ou représentent des valeurs importantes pour la personne.
- Relation sociale : La dimension relation sociale décrit l'interaction avec les autres. Les objets ou les systèmes facilitant la communication contribuent au plaisir social.

De la même manière que l'utilisabilité, les travaux de (Bernhaupt, 2015) montre que l'expérience utilisateur doit être prise en compte tout au long du processus de conception et de développement centré utilisateur.

2 Systèmes interactifs critiques[2]

Dans cette sous-section, afin de comprendre ce qu'est un système interactif critique, nous présentons une définition des systèmes interactifs critiques, puis nous présentons leurs propriétés et un exemple de processus pour garantir ces propriétés. La dernière sous-section présente un exemple de système interactif critique : le cockpit d'avion civil.

2.1 Définition d'un système critique

Un système critique peut être défini comme un système pour lequel une défaillance peut avoir un coût supérieur à celui du développement du système (Laprie, 1996). Ici, la notion de coût est général et diffère selon le type de système critique considéré (Sommerville, 2011) :

- L'intégrité d'une ou plusieurs personnes (vie humaine) ou environnementale (*safety-critical)* : une défaillance du système critique présente un risque pour des vies humaines ou de créer des dommages importants. Les systèmes tels que les systèmes de commandes et contrôle des centrales nucléaires dont la défaillance peut d'importants dommages environnementaux durables et la mort de personnes sont un exemple de ce type de système critique.
- L'économie d'une entreprise ou d'une organisation (*business-critical*) : une défaillance du système critique peut résulter en un coût économique important : autant budgétaire

[2] Modifications post-publication

que sur la notoriété. Les systèmes bancaires dont une défaillance peut résulter en une perte d'argent pour l'entreprise et ses clients ainsi qu'en une perte de confiance de ceux-ci envers l'entreprise sont un exemple de ce type de système critique.

- La mission (*mission-critical*) : une défaillance du système critique peut résulter en un échec la mission : le but supporté par le système n'est pas atteint. Les robots déployés pour des missions lunaires ou martiennes dont la défaillance en l'absence de possibilité d'intervention humaine peut résulter en la non atteinte du but de la mission sont un exemple de ce type de système critique.

2.2 Système interactif critique : le cas des systèmes de commande et contrôle

Un système interactif critique peut être défini comme un système interactif pour lequel une défaillance a un coût supérieur à celui du développement du système (Palanque & Bastide, 1994). De la même manière que les systèmes critiques, les systèmes interactifs critiques peuvent être classifiés selon les trois catégories présentées ci-avant selon le coût considéré :

- L'intégrité d'une ou plusieurs personnes (vie humaine) ou environnementale. Par exemple, les systèmes du cockpit d'avion civil dont la défaillance peut entrainer la perte de l'appareil, des dommages environnementaux et la mort des passagers.
- L'économie d'une entreprise ou d'une organisation. Par exemple, une voiture autonome dont la défaillance peut entrainer une perte de confiance des conducteurs envers l'entreprise qui la commercialise.
- La mission. Par exemple, les systèmes de commande et contrôle de lancement spatial dont la défaillance peut provoquer un échec du lancement d'un satellite.

2.3 Propriétés des systèmes interactifs critiques

Les propriétés de sureté de fonctionnement et de sécurité doivent être garanties pour les systèmes critiques (Avizienis et al., 2004) et (Laprie, 1996).

La sûreté de fonctionnement d'un système est définie comme la capacité à délivrer un service pour lequel les utilisateurs peuvent placer une confiance justifiée, ou bien, la capacité d'éviter les défaillances plus fréquentes et sévères qu'acceptable (Avizienis et al., 2004).

La sécurité d'un système est définie comme la capacité du système à garantir l'absence d'accès ou de manipulations non-autorisées à celui-ci (Avizienis et al., 2004). En d'autres termes, il s'agit de la capacité du système à se protéger des attaques visant à nuire.

La Figure 7 présente les attributs de ces propriétés ainsi que les entraves possibles de ces propriétés et les moyens (méthodes et techniques) permettant de garantir ces propriétés.

Les attributs de la sureté de fonctionnement et de la sécurité sont :

- La disponibilité (*Availability*) : la capacité du système à être prêt à assurer un service correct lors de son utilisation.

- La fiabilité (*Reliability*) : la capacité du système à assurer la continuité d'un service correct.
- La sécurité-innocuité (*Safety*) : la capacité du système à garantir l'absence de conséquences catastrophiques pour l'environnement et les utilisateurs.
- La confidentialité (*Confidentiality*) : la capacité du système à garantir l'absence de divulgation non autorisée d'informations (cet attribut ne concerne que la propriété de sécurité).
- L'intégrité (*Integrity*) : la capacité du système à garantir l'absence d'altération de son information et de son fonctionnement.
- La maintenabilité (*Maintainability*) : la capacité du système à évoluer en étant modifiable et réparable.

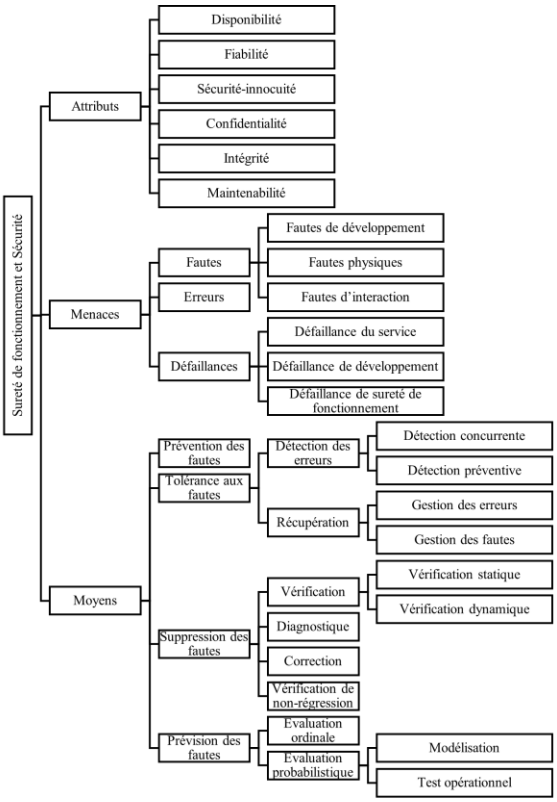


Figure 7. Arbre de Fiabilité et Sécurité d'après les travaux de (Avizienis et al., 2004)

2.4 Cockpit d'avion civil

Le système interactif dans les cockpits d'avion civil est appelé CDS (Control and Display System). Le CDS a pour but de fournir aux pilotes le soutien nécessaire pour leurs tâches de vol (FLY), de communication (COMMUNICATE), de navigation (NAVIGATE) et de maintien des systèmes avions (MANAGE SYSTEMS). Le cockpit de l'A350 est présenté en Figure 8.

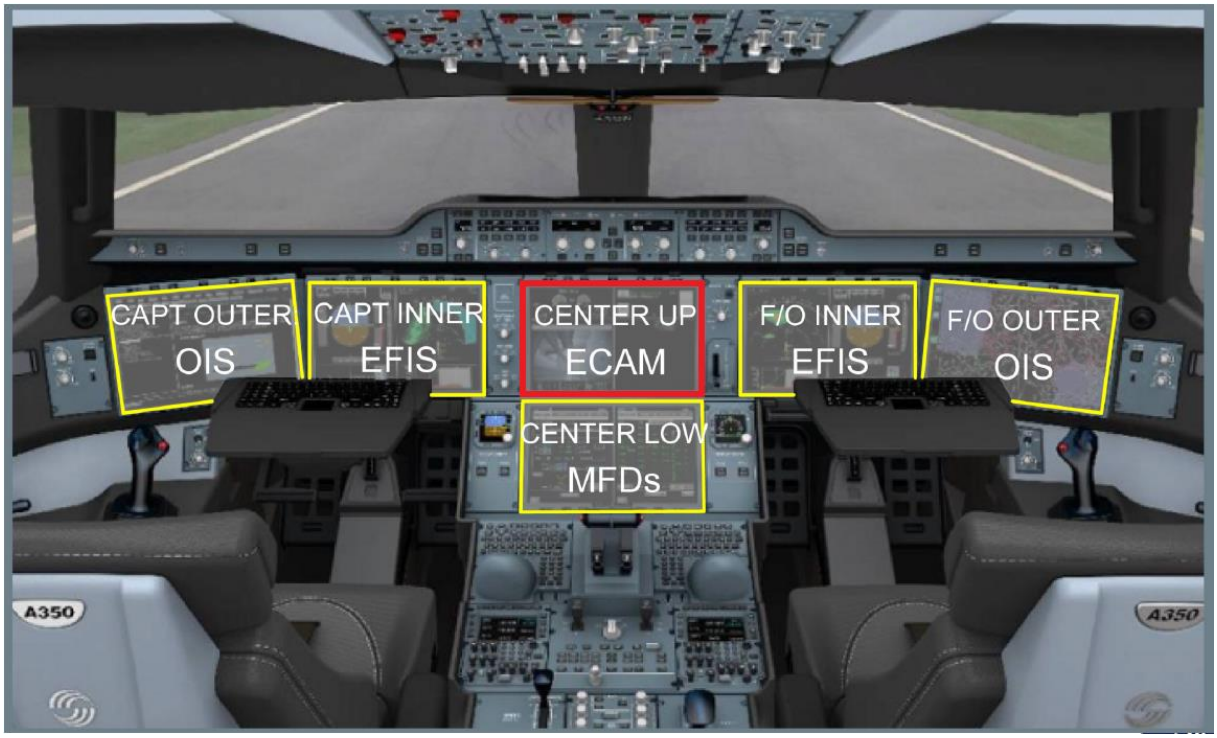


Figure 8. Cockpit de l'A350. L'ECAM est encadré en rouge.

Lorsqu'une ou plusieurs alarmes sont actives à la suite de la défaillance d'un système, les informations sur le système impacté, les limitations du système, l'alarme et la procédure associée à suivre sont affichées à l'ECAM (*Electronic Centralized Aircraft Monitor* encadré en rouge sur la Figure 8). Les pilotes interagissent avec l'ECAM grâce aux périphériques d'entrée de l'ECP (ECAM Control Panel présenté en Figure 9).

Ces systèmes font l'objet d'une présentation détaillée au Chapitre 9 où nous présentons l'étude de cas industrielle à laquelle les travaux de cette thèse sont appliquées.

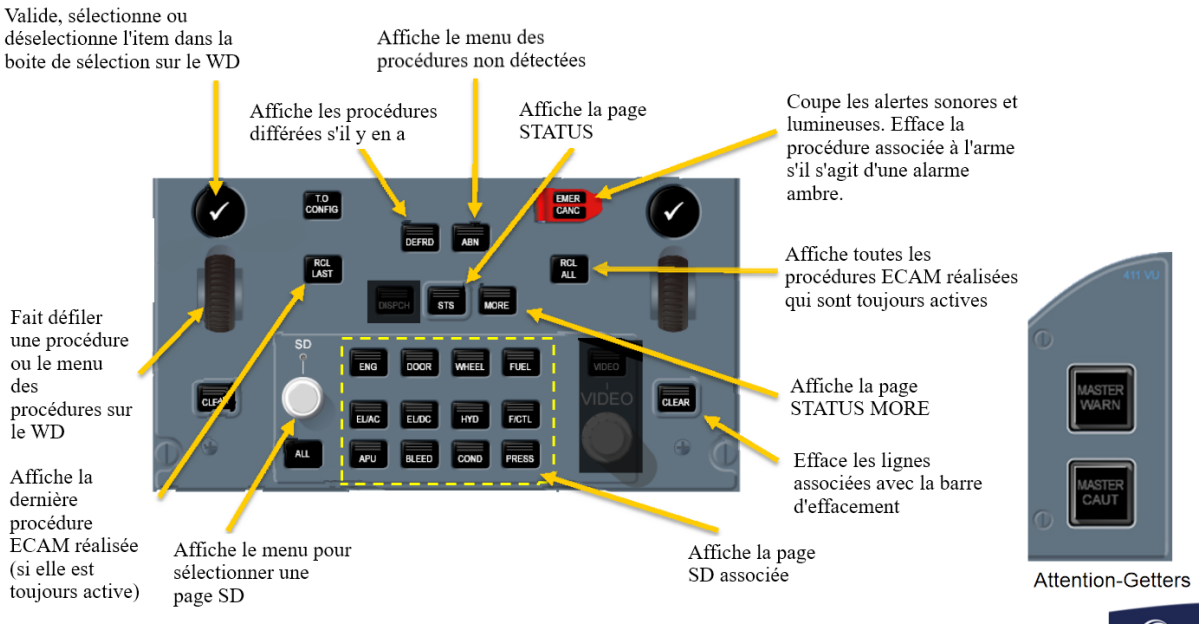


Figure 9. Description de l'ECP ("ECAM Control Panel")

Le cockpit d'avion civil est un exemple de système de commande et contrôle critique. Afin d'être autorisé à opérer avec des passagers, tous les avions doivent obtenir un certificat de navigabilité. Celui-ci est délivré par les autorités de navigabilité aérienne (ex. la Direction Générale de l'Aviation Civile) et garantit la conformité de l'avion aux normes de sécurité internationales définies par ces mêmes autorités de navigabilité aérienne. Les principales agences traitant de la sécurité aérienne sont : l'agence européenne EASA (European Aviation Safety Agency) et l'agence américaine FAA (Federal Aviation Agency). Ces agences délivrent un certificat de type lorsque l'avion répond aux spécifications de sécurité-innocuité (*safety)* européenne (EASA) ou américaine (FAA).

En conséquence, le développement des systèmes du cockpit d'avion civil doit respecter les exigences de l'EASA définies dans la norme CS-25 (EASA, 2014) pour pouvoir être certifiés. Pour cela, il est nécessaire de suivre les processus de développement définis par la norme DO-178C (RTCA & EUROCAE, 2011) pour la conception et le développement logiciel et la norme ARP 4754 pour la conception et le développement des systèmes de l'avion.

2.4.1 DO-178C et ARP 4754[3]

La norme DO-178C (RTCA & EUROCAE, 2011) est un standard aéronautique pour le développement logiciel. Ce standard défini des directives afin de garantir les propriétés de sureté de fonctionnement et de sécurité des équipements avions. Ainsi, les objectifs et les moyens pour prouver qu'elles sont atteintes couvrent tout le cycle de vie du logiciel. La Figure 10 présente le processus de développement défini par la norme DO-178C.

Ce processus est composé de trois sous-processus :

Le processus d'évaluation de la sureté de fonctionnement produit les fonctions, les défaillances possibles et des informations sur la sécurité et la sureté de fonctionnement à prendre en compte pendant le processus de développement.

Le processus de développement de l'avion et des systèmes qui produit les exigences systèmes pour le logiciel. Ce processus est couvert par l'ARP 4754. L'ARP 4754 définit des recommandations pour le développement des avions civils et de leurs systèmes. Cela inclut les activités de validation des exigences et de vérification de la conception des systèmes pour leur certification et leur assurance. L'ARP 4754 proposent des pratiques et un processus de développement pour l'avion et ses systèmes pour garantir la sécurité-innocuité (*Safety*) des systèmes.

Le processus de développement logiciel qui comprend quatre phases :

- **La phase d'analyse des exigences (*SW Requirements Process*)** produit les exigences fonctionnelles et non-fonctionnelles de haut niveau.
- **La phase de design (*SW Design Process*)** raffine les exigences de haut niveau en exigences bas niveau et permet de définir une architecture logicielle.

[3] Modifications apportées post-publication.

- **La phase de codage (*SW Coding Process*)** produit le code source à partir de l'architecture logicielle et des exigences de bas niveau définies dans la phase précédente.
- **La phase d'intégration (*Integration Process*)** réunie les différents composants du logiciel afin de produire un exécutable.

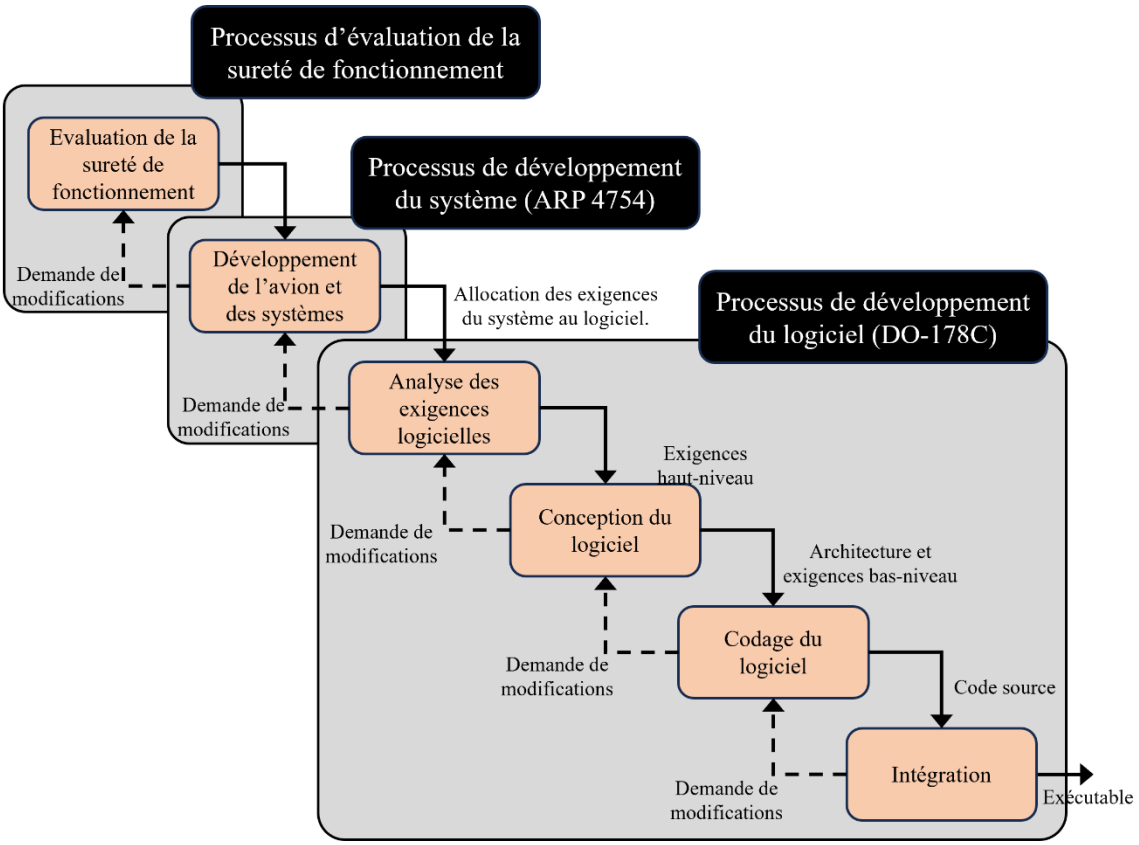


Figure 10. Processus de développement logiciel de la norme DO-178C.

A ce processus s'ajoute une phase de tests lors de laquelle sont produites des exigences de tests.

Pour pouvoir garantir la sûreté de fonctionnement du logiciel en fonction de son niveau de criticité, la norme DO-178C définit cinq niveaux d'assurance classés sur une échelle appelée DAL (*Development Assurance Level*), allant de A à E. Le Tableau 1 présente les niveaux de DAL associés à la sévérité des conditions de défaillance et l'exigence de probabilité d'occurrence des conditions correspondantes.

Tableau 1. Niveaux de criticité, DAL et objectifs en termes de probabilité d'occurrence.

Sévérité des conditions de défaillance	Niveau de DAL	Probabilité des conditions de défaillance (par heure de vol)
Catastrophique	A + *Fail Safe*	< 10-9
Dangereux (*Hazardous*)	B	< 10-7
Majeur	C	< 10-5
Mineur	D	< 10-3

Pas d'effet sur la sécurité et la sureté de fonctionnement	E	Pas d'exigence de probabilité d'occurrence

Ces niveaux de DAL s'alignent sur les niveaux de sévérité des défaillances définis par la norme CS25.

2.4.2 CS25

La norme CS-25 (EASA, 2014) décrit les spécifications des avions gros porteurs (par exemple, l'Airbus A350) afin d'obtenir un certificat de navigabilité de la part de la DGAC (Direction Générale de l'Aviation Civile). Cette norme fournit des moyens de prouver la conformité aux spécifications demandées : les AMC (*Acceptable Means of Compliance*).

Cette thèse traitant de l'automatisation des systèmes interactifs, nous nous intéressons particulièrement aux parties :

- CS-25.1302 définissant les exigences pour la conception des systèmes avec lesquels l'équipage interagit.
- Traitant de l'automatisation des systèmes avions.

CS 25.1302 Installed systems and equipment for use by the flight crew
(See AMC 25.1302)

This paragraph applies to installed equipment intended for flight-crew members' use in the operation of the aeroplane from their normally seated positions on the flight deck. This installed equipment must be shown, individually and in combination with other such equipment, to be designed so that qualified flight-crew members trained in its use can safely perform their tasks associated with its intended function by meeting the following requirements:

(a) Flight deck controls must be installed to allow accomplishment of these tasks and information necessary to accomplish these tasks must be provided.

(b) Flight deck controls and information intended for flight crew use must:

(1) Be presented in a clear and unambiguous form, at resolution and precision appropriate to the task.

(2) Be accessible and usable by the flight crew in a manner consistent with the urgency, frequency, and duration of their tasks, and

(3) Enable flight crew awareness, if awareness is required for safe operation, of the effects on the aeroplane or systems resulting from flight crew actions.

(c) Operationally-relevant behaviour of the installed equipment must be:

(1) Predictable and unambiguous, and

(2) Designed to enable the flight crew to intervene in a manner appropriate to the task.

(d) To the extent practicable, installed equipment must enable the flight crew to manage errors resulting from the kinds of flight crew interactions with the equipment that can be reasonably expected in service, assuming the flight crew is acting in good faith. This subparagraph (d) does not apply to skill-related errors associated with manual control of the aeroplane.

[Amdt. No.: 25/3]

Figure 11. CS-25.1302 (extrait de la norme CS-25 de l'EASA)

La CS-25.1302 présenté en Figure 11 impose que les systèmes de commande et contrôle permettent aux pilotes d'accomplir toutes leurs tâches et présentent les informations nécessaires à la réalisation de ces tâches. Les informations doivent « être présentées de manière claire et non ambigüe ». En concordance avec « l'urgence, la fréquence et la durée des tâches », les contrôles et les informations présentées doivent être « accessibles et utilisables ». Le comportement des systèmes de commande et contrôle doit être « prédictible, non ambigu » et « permettre aux pilotes d'intervenir » lorsqu'approprié. Enfin, les systèmes de commande et contrôle doivent permettre aux pilotes de « gérer les erreurs » pouvant survenir durant l'utilisation de tels systèmes.

La norme CS25 décrit également des spécifications pour l'automatisation des systèmes de commandes et contrôle. L'automatisation y est définie comme l'exécution **autonome** d'une tâche (ou de plusieurs tâches) par les systèmes aéronautiques déclenchée par une action de **contrôle** de haut niveau par l'équipage. Par conséquent, il est considéré dans la norme que

l'équipage a toujours **l'initiative** du déclenchement d'une fonction traduisant une automatisation.

> ***Automation*** *– The autonomous execution of a task (or tasks) by aeroplane systems started by a highlevel control action of the flight crew.*

Dans la norme CS25 Livre 1 – Annexe D définissant les critères pour déterminer l'équipage minimum, il est considéré comme significatif pour la charge cognitive de l'équipage de déterminer : le **degré d'automatisation** fournit par les systèmes avions (après une défaillance ou un disfonctionnement) pour permettre le croisement automatique ou une isolation des difficultés afin de minimiser la nécessité d'une action de l'équipage pour se prémunir contre la perte de puissance hydraulique ou électrique des commandes de vol ou d'autres systèmes essentiels. L'extrait correspondant est présenté en Figure 12.

CS 25 Book 1 - Appendix D - Criteria for determining minimum flight crew 1–App D–1
B) Workload factors. The following workload factors are considered significant when analyzing and demonstrating workload for minimum flight crew determination:
(7) The degree of automation provided in the aircraft systems to afford (after failures or malfunctions) automatic crossover or isolation of difficulties to minimise the need for flight crew action to guard against loss of hydraulic or electrical power to flight controls or other essential systems.

Figure 12. Extrait de la CS-25 Livre 1 - Annexe D Critères pour déterminer l'équipage minimum

Dans la norme CS25 Livre 2 une partie est dédiée aux recommandations pour la conception des systèmes contenant de l'automatisation. Nous en présentons un extrait dédié aux **contrôles** pour les systèmes contenant de l'automatisation. L'extrait correspondant est présenté en Figure 13. Il est spécifié que les systèmes contenant de l'automatisation peuvent exécuter des fonctions sélectionnées par l'équipage et **supervisées** par l'équipage. Les contrôles des systèmes contenant de l'automatisation doivent permettre à l'équipage de préparer le système à une fonction en toute sécurité. Une nouvelle tâche ne doit pas avoir d'interférence avec (ou être confondue avec) une tâche exécutée par le système. Ces contrôles doivent permettre à l'équipage d'activer la fonction voulue sans confusion possible. Ces contrôles doivent permettre à l'équipage **d'intervenir manuellement** dans n'importe quelle fonction du système contenant de l'automatisation comme requis par les conditions opérationnelles ou bien de revenir à un **contrôle manuel**.

Controls for Automated Systems
Automated systems can perform various tasks selected by and under supervision of the flight crew. Controls should be provided for managing functionalities of such a system or set of systems. The design of such "automation specific" controls should enable the crew to:

- Safely prepare the system for the task to be executed or the subsequent task to be executed. Preparation of a new task (for example, new flight trajectory) should not interfere with, or be confused with, the task being executed by the automated system.
- Activate the appropriate system function without confusion about what is being controlled, in accordance with crew expectations. For example, the flight crew should have no confusion when using a vertical speed selector which could set either vertical speed or flight path angle.

- Manually intervene in any system function, as required by operational conditions, or to revert to manual control. For example, manual intervention might be needed during loss of system functionality, system abnormalities, or failure conditions.

Figure 13. Extrait de la CS-25 traitant des recommandations les contrôles pour les systèmes contenant de l'automatisation

Dans cette même partie de la norme CS-25 Livre 2, nous présentons un autre extrait dédié aux affichages pour les systèmes contenant de l'automatisation. L'extrait correspondant est présenté en Figure 14. Il est spécifié que les systèmes contenant de l'automatisation peuvent exécuter des fonctions avec une intervention minimale de la part de l'équipage mais sous la supervision de l'équipage. Pour assurer cette supervision et maintenir la **conscience de la situation** de l'équipage (*crew awareness*), les affichages des systèmes doivent fournir un retour reconnaissable sur : les entrées de l'équipage (permettre la détection et la correction d'erreurs), l'état courant du système contenant de l'automatisation (qu'est-il en train de faire ?), les actions prises par le système pour atteindre ou maintenir un état désiré (qu'essaie-t-il de faire ?), les futurs états prévus par l'automatisation (que va-t-il faire ensuite ?), les **transitions** entre les états systèmes.

Display for Automated Systems

Automated systems can perform various tasks with minimal crew interventions, but under the supervision of the flight crew. To ensure effective supervision and maintain crew awareness of system state and system "intention" (future states), displays should provide recognisable feedback on:

- Entries made by the crew into the system so that the crew can detect and correct errors
- Present state of the automated system or mode of operation. (What is it doing?)
- Actions taken by the system to achieve or maintain a desired state. (What is it trying to do?)
- Future states scheduled by the automation. (What is it going to do next?)
- Transitions between system states.

Figure 14. Extrait de la CS-25 traitant des recommandations pour les affichages des systèmes contenant de l'automatisation

La norme CS-25 précise dans cette même partie que, lors de la conception des systèmes contenant de l'automatisation, les aspects suivants doivent être pris en compte :

- Les affichages suites aux commandes de l'équipage et les affichages décrivant l'état courant du système doivent permettre à l'équipage de déterminer si les systèmes automatisés fonctionneront conformément à leurs attentes
- Si le système contenant de l'automatisation se rapproche de son **autorité** opérationnelle ou fonctionne de manière anormale, ou est incapable de fonctionner au niveau sélectionné, il doit en informer l'équipage, de manière appropriée par rapport à la tâche courante
- Le système contenant de l'automatisation doit soutenir la **coordination** et la **coopération** en assurant une **conscience partagée** de l'état du système et des entrées de l'équipage dans le système.
- Le système contenant de l'automatisation doit permettre à l'équipage d'examiner et de confirmer l'exactitude des commandes avant de les activer.

Ces spécifications et recommandations s’appuient sur les travaux visant à définir l’automatisation et à concevoir des systèmes contenant de l’automatisation pour améliorer les performances et réduire l’effort des utilisateurs.

3 Conclusion

Dans ce chapitre, nous avons présenté les systèmes interactifs et les systèmes interactifs critiques faisant l’objet de cette thèse. Nous avons présenté les particularités architecturales et les propriétés à garantir pour ces types de systèmes. Également, nous avons présentés des processus permettant de garantir ces propriétés. Dans le cas des systèmes critiques, la garantie des propriétés de sureté de fonctionnement et de sécurité est d’une importance particulière. L’utilisation de méthodes formelles est citée comme étant plus efficace pour trouver certaines erreurs que le test et fournissant des preuves de vérification objectives telles que la cohérence et l'exactitude d'une représentation formellement spécifiée par le DO-178C pour la conception et le développement logiciel dans le domaine de l’aéronautique. De manière similaire, l’ARP 4754 recommande de pratiques et des méthodes pour la conception et le développement système dans le domaine de l’aéronautique. Nous présenterons dans cette thèse une approche à base de modèles pour la description des tâches et des fonctions et de leurs caractéristiques lors de l’interaction avec un système interactif contenant de l’automatisation. La norme CS-25 définit des spécifications pour la conception de l’automatisation dans les systèmes de commande et contrôle. En effet, lorsque le système interactif (critique ou non) contient de l’automatisation, d’autres propriétés sont à garantir et d’autres techniques existent afin de les prendre en compte. Ceci fait l’objet du chapitre suivant.

Chapitre 2. Systèmes contenant de l'automatisation et leur analyse

Différents constituants de l'automatisation sont étudiés selon le domaine de recherche considéré. Nous pouvons citer le domaine de la conception de l'automatisation avec le groupe de recherche ACM SIGDA (*Special Interest Group on Design Automation*) qui étudie la conception électronique de l'automatisation. Également, nous pouvons citer le domaine des facteurs humains pour la conception de l'automatisation qui étudie la conception de l'automatisation avec la prise en compte de l'humain, de ses capacités et limitations. Nous pouvons encore citer le domaine de recherche de l'IHM où l'interaction entre l'humain et le système interactif est étudié. L'automatisation étant présente dans un nombre de plus en plus important de systèmes interactifs (cockpit, voiture, téléphone), l'IHM inclue l'étude de l'interaction entre humain et système contenant de l'automatisation.

Cette thèse se situant dans le domaine de l'IHM, nous présentons l'automatisation dans le but de concevoir des systèmes interactifs contenant de l'automatisation soutenant les buts des utilisateurs et répondant à leurs besoins. A cela, s'ajoute la prise en considération de propriétés propres aux systèmes interactifs contenant de l'automatisation que nous présentons dans ce chapitre. Pour cela, nous nous intéresserons plus spécifiquement à l'analyse des tâches et fonctions de l'humain et du système pour l'atteinte d'un but commun.

Dans ce chapitre nous définissons les différent concepts et méthodes pour un utilisateur et un système interactif contenant de l'automatisation. Cependant, ces définitions sont applicables à un ou plusieurs utilisateurs interagissant avec un ou plusieurs systèmes contenant de l'automatisation.

La première section présente les définitions de l'automatisation.

La seconde section présente des exemples de problèmes causés par une mauvaise conception de l'automatisant dans les systèmes interactifs et présente deux propriétés à garantir pour éviter ces problèmes.

La troisième section présente différentes méthodes d'allocation de fonctions et de tâches entre l'utilisateur et le système interactif contenant de l'automatisation.

La quatrième section présente différentes méthodes de conception des systèmes interactifs contenant de l'automatisation prenant en compte l'allocation des ressources, de l'autorité, de la responsabilité, les transitions de contrôle et l'initiative.

La cinquième section conclut ce chapitre.

1 Définitions de l'automatisation

Les termes « automatisation » et « autonomie » ont souvent été utilisés indifféremment ou bien dans le but de mieux définir un type de système particulier (Vagia et al., 2016). Le terme

automatisation a d'abord pu être défini comme l'exécution par le système d'une tâche réalisée précédemment par un humain (Raja Parasuraman, 2000). L'automatisation définie comme telle désigne la migration de tâche de l'humain vers le système. Cette définition ne capture pas les systèmes contenant de l'automatisation capables de réaliser des tâches que les humains n'avaient pas la capacité de réaliser auparavant. Nous pouvons citer par exemple, les systèmes utilisant le radar afin de sonder les fonds marins.

Dans cette thèse nous considérons que l'automatisation permet aux systèmes d'exécuter des fonctions sans contrôle extérieur et d'adapter les fonctions qu'ils exécutent à de nouveaux contextes ou buts. Cette définition considère que les systèmes interactif contenant de l'automatisation sont des systèmes capables d'un certain degré d'autonomie. L'autonomie est définie par (Bradshaw et al., 2013) comme étant bidirectionnelle :

- *Self-directedness* (autogestion) : cette notion exprime la liberté du système du contrôle extérieur, son habilité à exécuter ses fonctions sans ordre émanant de l'utilisateur.
- *Self-sufficiency* (auto-suffisance) : cette notion est l'équivalent du terme français « autonome » et exprime être capable de s'occuper de soi-même, l'habilité du système à auto générer des buts.

L'autosuffisance du système interactif contenant de l'automatisation permet de réduire la charge revenant aux utilisateurs. Le système interactif contenant de l'automatisation est capable de réaliser des fonctions pour soutenir un but utilisateur et est capable de s'adapter à de nouvelles situations. L'autogestion du système interactif contenant de l'automatisation est sa capacité à se passer d'un contrôle de l'utilisateur pour l'exécution de ses fonctions.

2 Propriétés de l'automatisation

Une automatisation avec des défauts de conception peut provoquer des problèmes de surprises d'automatisation ou de *out-of-loop*. Les surprises d'automatisation sont des situations où le système contenant de l'automatisation a un comportement en dehors des attentes des utilisateurs qui le supervise (Woods, 1996). Le système contenant de l'automatisation change de comportement en se basant sur des facteurs situationnels et systèmes sans donner à l'utilisateur la capacité de suivre et d'anticiper ces changements de comportement (Sarter et al., 1997). Le problème *out-of-loop* exprime la perte de performance que peuvent expérimenter les utilisateurs lorsqu'ils sont retirés de la boucle perception-action et qu'ils doivent réagir à une nouvelle situation ou reprendre le contrôle d'une tâche précédemment allouée au système (Endsley & Kiris, 1995).

Tout comme les systèmes interactifs et les systèmes critiques, les systèmes contenant de l'automatisation possède des propriétés qui leurs sont propres.

Nous présentons dans cette section deux propriétés de ces systèmes : *trust* et *situation awareness*.

2.1 Trust

(J. D. Lee & See, 2004) définit la confiance (trust) comme :

> *La manière dont un système [un agent] aidera à atteindre les objectifs d'un individu dans une situation caractérisée par l'incertitude et la vulnérabilité.*

La confiance utilisateur-système contenant de l'automatisation dépend de la performance, du processus ou le but du système contenant de l'automatisation (J. Lee & Moray, 1992).

(T. B. Sheridan, 1988) propose des attributs de la confiance utilisateur-système contenant de l'automatisation afin de la mesurer et de la modéliser :

- La fiabilité du système,
- La robustesse du système,
- La familiarité : le système utilise des procédures, des termes ou des normes familières à l'utilisateur,
- La compréhensibilité : l'utilisateur peut former un modèle mental du système et prédire le comportement du système
- L'explication des intentions : le système communique explicitement qu'il agira d'une certaine façon
- L'utilité du système pour l'utilisateur
- La dépendance de l'utilisateur au système

Dans la revue réalisée par (Hoff & Bashir, 2015), les auteurs identifient trois couches de variabilité de la confiance : confiance dispositionnelle (tendance durable de l'utilisateur à faire confiance au système), confiance situationnelle (dépend de la situation actuelle) et confiance acquise (dépend de l'expérience de l'utilisateur avec le système). Également, (Hoff & Bashir, 2015) identifie que les concepteurs peuvent fournir à l'utilisateur un retour d'information permanent sur la fiabilité de l'automatisation et les facteurs situationnels qui affectent les performances afin de favoriser une confiance appropriée de l'utilisateur au système contenant de l'automatisation.

2.2 *Situation awareness*

(Endsley, 1995) propose une définition de *situation awareness* (conscience de la situation) :

> *La situation awareness est la perception des éléments de l'environnement dans un volume de temps et d'espace, la compréhension de leur signification et la projection de leur état dans un futur proche.*

(Endsley, 1995) définit également trois niveaux de *situation awareness* :

- Niveau 1 : Perception des éléments de l'environnement. La première étape de pour atteindre la *situation awareness* est de percevoir l'état et les attributs des éléments pertinents de l'environnement.
- Niveau 2 : Compréhension de la situation actuelle. La deuxième étape pour atteindre la *situation awareness* est de comprendre la signification des éléments pertinents perçus en niveau 1.

- Niveau 3 : Projection des futurs états. La troisième et dernière étape pour atteindre la *situation awareness* est la capacité de prévoir les futures actions des éléments de l'environnement.

Le modèle de la *situation awareness* proposé par (Endsley, 1995) est présenté en Figure 15. Ce modèle intègre les trois étapes pour atteindre la *situation awareness* dans le processus de prise de décision des utilisateurs.

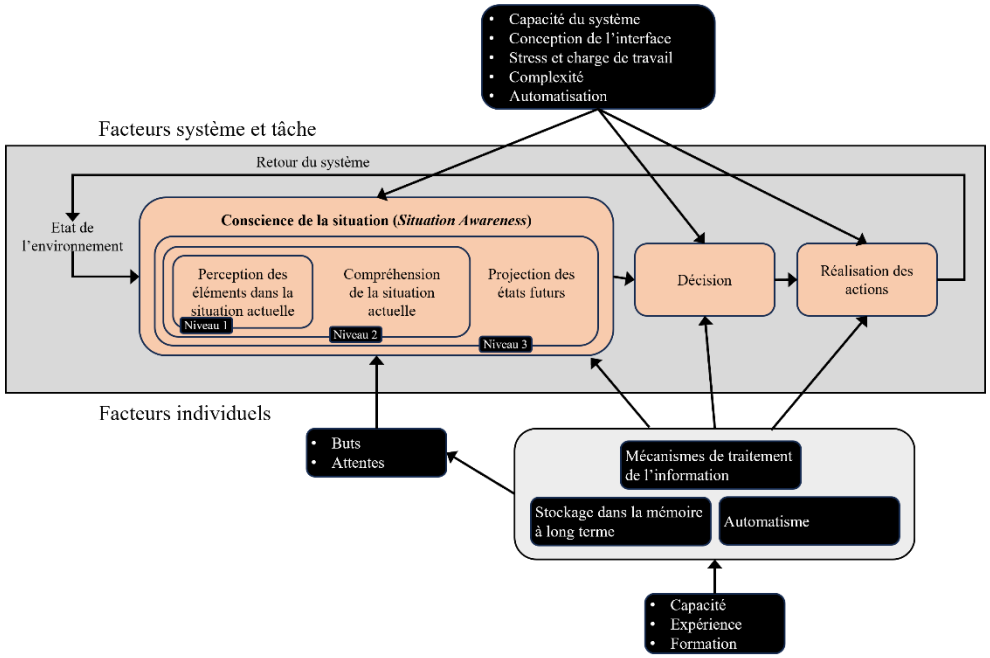


Figure 15. Modèle de la *situation awareness* dans la prise de décision de (Endsley, 1995)

Une perte de la propriété de *situation awareness* provoque le problème *out-of-loop* comme démontré par (Endsley & Kiris, 1995) puis confirmé par de nombreuses autres études.

Ces méthodes de conception où les buts, les capacités, les tâches et le contexte d'utilisation du système par l'utilisateur sont pris en compte sont appelées des méthodes de conception de l'automatisation centrée utilisateur. (Billings, 1991) défini l'automatisation centrée utilisateur comme l'automatisation conçue pour travailler en coopération avec des opérateurs humains à la poursuite d'objectifs définis.

3 Allocation de fonctions et de tâches

L'allocation de fonctions et de tâches est l'identification des fonctions exécutées par le ou les systèmes et des tâches réalisées par l'utilisateur ou les utilisateurs.

3.1 MABA MABA

Une première approche de l'allocation des fonctions et des tâches a été de migrer vers le système le plus de tâches possibles dans le but d'augmenter les performances en réduisant la charge mentale et physique des utilisateurs. (Dekker & Woods, 2002) rappelle que cette approche encouragée dans le domaine aéronautique dans les années 70 a conduit à une

diminution du *mode* (ou *situation*) *awareness* provoquant des surprises d'automatisation (*automation surprise*) comme décrit par (Sarter & Woods, 1997).

Afin de déterminer comment allouer les tâches et les fonctions entre l'humain et le système, Fitts (Paul M. Fitts, 1951) a proposé une première approche pour l'allocation des tâches et des fonctions : la liste MABA MABA (Men Are Better At, Machine Are Better At) présenté dans le Tableau 2. Cette approche suppose que les humains sont, de par leurs capacités humaines, meilleurs pour les tâches consistant à détecter des variations dans l'environnement, percevoir des patterns, improviser, apprendre des informations et s'en rappeler sur le long terme, raisonner de manière inductive et exercer un jugement. De la même manière, les systèmes seraient, de par leurs capacités, plus rapides pour répondre aux signaux de commande, meilleurs pour appliquer une grande force avec précision, meilleurs pour stocker brièvement des informations et les effacer et pour raisonner par déduction. Plusieurs autres travaux ont proposé des listes similaires désignées sous le nom de « Fitts lists ». Par exemple, nous pouvons citer les travaux de (Mertes & Jenney, 1974) pour l'allocation de tâches et de fonctions pour le contrôle aérien et les travaux de (Swain, 1983) pour l'allocation de tâches et de fonctions dans les centrales nucléaires. Également, nous pouvons citer les travaux de (Chapanis, 1965) qui précise que cette approche à base de liste n'est pertinente que pour un stade de développement technologique et qu'il n'est pas possible de prédire précisément les performances humaines ou du système en amont dans le cycle de conception et de développement (Price et al., 1982). Ces travaux précisent que l'allocation de fonctions et de tâches peut varier au cours du cycle de conception et de développement.

Tableau 2. La liste MABA MABA de (Paul M. Fitts, 1951)

Les humains sont meilleurs à	Les machines sont meilleures à
Détecter de petites quantités d'énergie visuelle, auditive ou chimique Percevoir des patterns de lumière ou de son Improviser et utiliser des procédures flexibles Stocker des informations pendant de longues périodes et se rappeler les informations appropriées Raisonner de manière inductive Exercer un jugement	Répondre rapidement aux signaux de commande Appliquer une grande force en douceur et avec précision Stocker brièvement les informations et les effacer complètement Raisonner par déduction

Les limitations de la liste MABA MABA de Fitts ont fait l'objet de plusieurs contributions. Nous pouvons citer (Dekker & Woods, 2002) qui soulignent que le MABA MABA néglige la conception de la coopération utilisateur-automatisation. Également, (T.B. Sheridan, 2000) énumère une liste de problèmes soulevés par le MABA MABA. Parmi ces problèmes nous pouvons citer que, bien que l'allocation des tâches et des fonctions peut être dynamique comme le souligne (P. A. Hancock & Scallen, 1996), le MABA MABA ne propose qu'une allocation statique des tâches et des fonctions. De plus le MABA MABA n'est pas adapté aux avancées technologiques et l'allocation des tâches et des fonctions est réalisée a trop haut niveau (T.B. Sheridan, 2000).

3.2 Niveaux d'automatisation

Dans le but de fournir une automatisation appropriée aux besoins utilisateurs, plusieurs descriptions de niveaux d'automatisation ont été proposées. Une revue de ces différents niveaux d'automatisations a été réalisée par (Vagia et al., 2016). Nous les présentons et complétons cette revue ci-après.

3.2.1 Niveaux d'automatisation de Sheridan, Verplank et Parasuraman

Les travaux de (Thomas B. Sheridan & Verplank, 1978) proposent une première description de différents niveaux d'automatisation possibles des tâches de télé-opération sous-marine. Ces niveaux d'automatisation sont présentés dans la Figure 16.

Level	Description of Interaction
1	human does the whole job up to the point of turning it over to the computer to implement
2	computer helps by determining the options
3	computer helps determine options and suggests one, which human need not follow.
4	computer selects action and human may or may not do it.
5	computer selects action and implements it if human approves
6	computer selects action, informs human in plenty of time to stop it
7	computer does whole job and necessarily tells human what it did
8	computer does whole job and tells human what it did only if human explicitly asks

Figure 16. Niveaux d'automatisation de (Thomas B. Sheridan & Verplank, 1978) pour la prise de décision pour une seule étape élémentaire de la décision.

Les travaux de (Thomas B. Sheridan, 1992) présentent 10 niveaux d'automatisation basés sur ceux de (Thomas B. Sheridan & Verplank, 1978). Ces 10 niveaux d'automatisation sont présentés en Figure 17. Pour chacun des 10 niveaux, l'interaction entre l'utilisateur et le système pour le choix de l'action à réaliser est décrite lorsqu'elle existe.

Elevé	10	L'ordinateur décide tout, agit de manière autonome en ignorant l'humain.
	9	Informe l'humain seulement si l'ordinateur le décide
	8	Informe l'humain seulement s'il le demande, ou
	7	S'exécute automatiquement, puis informe obligatoirement l'humain, et
	6	Autorise l'humain un temps limité pour annuler (veto) avant l'exécution automatique, ou
	5	Exécute cette suggestion si l'humain approuve, ou
	4	Suggère une alternative
	3	Diminue la sélection à quelques-uns, ou
	2	L'ordinateur offre un ensemble complet d'alternatives de décision/action, ou
Bas	1	L'ordinateur n'offre aucune assistance : l'humain doit prendre toutes les décisions et réaliser toutes les actions.

Figure 17. 10 niveaux d'automatisation d'après les travaux de (R. Parasuraman et al., 2000)

(R. Parasuraman et al., 2000) précise que les différentes étapes d'une tâche peuvent être d'un niveau d'automatisation différent. Ces travaux proposent également un processus pour déterminer le niveau d'automatisation appropriée pour chaque étape des tâches à réaliser.

3.2.2 Niveaux d'automatisation d'Endsley et Kiris

(Endsley, 1987) présente un modèle à 4 niveaux d'automatisation pour le cockpit interactif. Dans cette première version, elle n'inclut pas de niveau d'automatisation décrivant un contrôle totalement manuel et elle propose moins de variations de niveaux d'automatisation que (Thomas B. Sheridan & Verplank, 1978). (Endsley & Kiris, 1995) ont étudié chacun des niveaux d'automatisation proposés dans (Endsley, 1987), avec l'addition du niveau manuel d'automatisation, dans l'objectif d'identifier les niveaux d'automatisation où l'utilisateur est maintenu dans la boucle afin d'éviter qu'il soit *out-of-loop* et d'avoir un impact négatif sur le *situation awareness*. En se basant sur les travaux de (Draper, 1995) et sur ses précédents travaux, (Endsley, 1999) propose 10 niveaux d'automatisation présentés dans la Figure 18. Ces niveaux d'automatisation présentent l'avantage de fournir une applicabilité à une large gamme de tâches cognitives et psychomotrices nécessitant un contrôle en temps réel des systèmes dans de nombreux domaines, y compris le contrôle du trafic aérien, le pilotage d'aéronefs, la fabrication avancée et les télé-opérations.

10	Automatisation complète	L'ordinateur réalise toutes les tâches. L'humain ne peut pas intervenir.
9	Contrôle de supervision	Le système génère des options recommandées, sélectionne l'option à implémenter et la réalise. L'humain surveille (*monitoring*) le système et intervient si nécessaire.
8	Prise de décision automatisée	Le système se base sur une liste d'options qu'il génère en même temps que l'humain pour sélectionner la meilleure et la réaliser.
7	Système rigide	Le système présente seulement un ensemble limité d'options recommandées à l'humain et les options possible sont restreintes à cet ensemble. Le système réalise l'option choisie.
6	Prise de décision mélangée	L'ordinateur génère une liste d'options recommandées, en sélectionne et réalise une si l'humain y consent. L'humain peut approuver l'option choisi ou en sélectionner une autre proposé par l'ordinateur ou l'humain.
5	Aide à la décision	L'ordinateur génère une liste d'options recommandées parmi lesquelles l'humain peut choisir. L'humain peut générer ses propres options. L'humain choisit une option et l'ordinateur l'implémente.
4	Contrôle partagé	L'humain et l'ordinateur génèrent et sélectionnent des options possibles. L'humain décide de l'option à implémenter. L'humain réalise l'option choisie avec l'aide du système système.
3	Traitement par lots	L'humain génère et sélectionne les options à réaliser, l'ensemble d'options choisi est ensuite réalisé automatiquement par le système.
2	Aide à l'action	L'humain réalise la tâche avec l'aide du système.
1	Contrôle manuel	L'humain réalise toutes les tâches.

Figure 18. Les 10 niveaux d'automatisation de (Endsley, 1999).

Les travaux de (Endsley, 2017) présentent une nouvelle version des niveaux d'automatisation. Ces niveaux d'automatisation sont présentés dans la Figure 19. Les niveaux « *information cueing* » et « *situation awareness support* » complètent les niveaux d'automatisation proposés dans (Endsley, 1999). Chaque étape d'une tâche est allouée soit à l'humain, soit au système, soit au deux. La décomposition d'une tâche par Endsley diffère de celle proposée par (R. Parasuraman et al., 2000). (R. Parasuraman et al., 2000) propose de décomposer la phase de perception de l'information en 2 étapes alors qu'Endsley traite cette phase en une seule étape (*monitoring information*) mais décompose la phase de prise de décision en 2 étapes (génération d'options et sélection d'une action).

Niveau d'automatisation	Description	Allocation			
		Monitoring et Présentation de l'information	Génération d'options	Prise de décision/Sélection du déroulement des actions	Implémentation des actions
Automatisation complète	L'ordinateur réalise toutes les tâches. L'humain ne peut pas intervenir.	Le système collecte les informations.	Le système	Le système	Le système
Contrôle de supervision	Le système génère des options recommandées, sélectionne l'option à implémenter et la réalise. L'humain surveille (*monitoring*) le système et intervient si nécessaire.	L'humain monitore le système et peut intervenir.	Le système	Le système	Le système
Prise de décision automatisée	Le système se base sur une liste d'options qu'il génère en même temps que l'humain pour sélectionner la meilleure et la réaliser.	L'humain et le système collectent les informations.	L'humain et le système	Le système décide	Le système
Système rigide	Le système présente seulement un ensemble limité d'options recommandées à l'humain et les options possible sont restreintes à cet ensemble. Le système réalise l'option choisie.	L'humain et le système collectent les informations.	Le système	L'humain décide	Le système
Prise de décision mélangée	L'ordinateur génère une liste d'options recommandées, en sélectionne et réalise une si l'humain y consent. L'humain peut approuver l'option choisi ou en sélectionner une autre proposé par l'ordinateur ou l'humain.	L'humain et le système collectent les informations.	L'humain et le système	Le système décide, l'humain consent	Le système
Aide à la décision	L'ordinateur génère une liste d'options recommandées parmi lesquelles l'humain peut choisir. L'humain peut générer ses propres options. L'humain choisit une option et l'ordinateur l'implémente.	L'humain et le système collectent les informations.	L'humain et le système	L'humain décide	Le système
Contrôle partagé	L'humain et l'ordinateur génèrent et sélectionnent des options possibles. L'humain décide de l'option à implémenter. L'humain réalise l'option choisie avec l'aide du système système.	L'humain et le système collectent les informations.	L'humain et le système	L'humain décide	L'humain et le système réalisent la seule tâche choisie
Traitement par lots	L'humain génère et sélectionne les options à réaliser, les ensembles d'options choisis sont ensuite réalisés automatiquement par le système.	L'humain et le système collectent les informations.	L'humain	L'humain	Le système réalise des ensembles de tâches
Aide à l'action / Télé-opération	L'humain réalise la tâche avec l'aide du système.	L'humain et le système collectent les informations.	L'humain	L'humain	L'humain et le système réalisent la seule tâche choisie
Aide à la conscience de la situation	L'ordinateur collecte les informations clefs et les intègre pour les niveaux 2 et 3 de conscience de la situation.	L'humain et le système collectent les informations. Le système intègre les informations.	L'humain	L'humain	L'humain
Signal/Repère d'information	L'ordinateur met en valeur les informations clefs.	L'humain et le système collectent les informations. Le système met en valeur des informations.	L'humain	L'humain	L'humain
Contrôle manuel	L'humain réalise toutes les tâches.	L'humain collecte les information.	L'humain	L'humain	L'humain

Figure 19. Niveaux d'automatisation proposés par (Endsley, 2016)

3.2.3 Autres niveaux d'automatisation

D'autres niveaux d'automatisation ont été proposés. Par exemple, (Ntuen & Park, 1988) propose d'ajouter le contrôle manuel aux niveaux d'automatisation de (Endsley, 1987). (Riley, 1989) propose de classifier les niveaux d'automatisation selon le niveau « d'intelligence » du système. Ce niveau d'intelligence décrit si le système est capable de fournir une réponse personnalisée ou seulement les données brutes à l'utilisateur par exemple. (Milgram et al., 1995) propose des niveaux d'automatisation pour le contrôle de robot téléopérés prenant en compte trois dimensions : le degré d'autonomie du système, le niveau de structure de l'environnement distant (objets, géométrie, localisation et orientation des objets connus ou non) et le niveau de connaissance sur le monde distant par le système. (Lorenz et al., 2001) propose 3 niveaux d'automatisation selon le niveau d'aide apportée par le système à l'utilisateur pour la gestion des fautes. (Clough, 2002) propose 4 niveaux d'automatisation pour les véhicules aériens sans pilotes (UAV) : piloter à distance, opérer à distance, superviser à distance, autonomie totale. (Proud et al., 2003) propose 8 niveaux d'automatisation basés sur l'allocation entre l'humain et le système de 4 étapes proposées pour décrire une tâche : observer, orienter (analyse des données), décider et agir. (Fereidunian et al., 2007) propose de rajouter un niveau d'automatisation aux niveaux de (R. Parasuraman et al., 2000) où le système acquière les données et les stocke sans les analyser. (Vagia et al., 2016) proposent leurs propres 8 niveaux d'automatisation afin de montrer comment des niveaux d'automatisation peuvent être créés. Ces 8 niveaux d'automatisation combinent des niveaux similaires des différents niveaux d'automatisation présentés, sélectionnés parmi des niveaux de d'automatisation récurrents et des niveaux d'automatisation proposés plus rarement. Également, (Kortenkamp et al., 1997) proposent 4 niveaux d'automatisation pour la coopération humain-robot : faire équipe, supervision, partage de tâches, guidage par l'humain et télé-opération. (Billings, 1991) propose 7 niveaux d'automatisation pour les tâches des pilotes dans le cockpit d'avion. Ces niveaux d'automatisation sont présentés en Figure 20.

		Fonction de l'automatisation	Tâches de l'humain	
Très élevé	Opération autonome	L'opération est totalement autonome Le pilote n'est généralement pas informé Le système peut être capable d'être désactivé ou pas.	Le pilote n'a pas de rôle dans l'opération. Le monitoring est limité à la détection des fautes. Les buts sont auto-définis. Le pilote n'a normalement pas de raison d'intervenir.	Très élevé
Niveau d'automatisation	Gestion par exception	L'opération est essentiellement autonome La reconfiguration est automatique Le système informe le pilote et monitore ses réponses.	Le pilote est informé des intentions du système. Le pilote doit consentir aux décisions critiques. Le pilote peut intervenir pour revenir à un niveau plus bas de gestion.	Niveau d'implication humaine
	Gestion par consentement	Les contrôle de l'avion et du vol est totalement automatisé. Les intentions et les diagnostiques du système sont fournis au pilote.	Le pilote doit consentir aux changements d'états, à l'exécution des checklists, la résolution des situations anormales. Le pilote doit manuellement exécuter les actions critiques.	
	Gestion par délégation	Le système possède des fonctions de contrôle du plan de vol par l'auto-pilote et l'automanette. La communication automatique et suivi de la navigation sont gérés par le système.	Le pilote commande le cap, l'altitude et la vitesse. La navigation est manuelle ou partagée avec le système. Le pilote commande les operation systèmes, les checklists et les communications.	
	Contrôle partagé	Le contrôle et le guidage sont améliorés par le système. Le système intelligent fournis des recommandations; Le système affiche des plans de vols potentiels et d'autres prédicteurs.	Le pilote a le contrôle à travers le CWS ou le système protégeant de toute sortie de l'enveloppe. Le pilote peut utiliser le système de recommandation. Le pilote gère manuellement les systèmes.	
	Contrôle manuel assisté	Le système poss-de des fonctions de directeur de vol, FMS et des modules de navigation; Le système gère la liaison de données avec les messages manuels; Le système monitore le contrôle du plan de vol et les systèmes avions	Le pilote a une autorité directe sur tous les systèmes; Le contrôle manuel du pilote se fait à partir de données du F/D et des affichages de navigation améliorés; Le FMS est disponible; Le pilote peut avoir des informations sur demande.	
Très bas	Contrôle manuel direct	Le système gère les avertissements et les alertes; Le système permet la communication vocale avec l'ATC et la communication ACARS de routine réalisée automatiquement.	Le pilote a une autorité directe sur tous les systèmes; Le contrôle manuel du pilote se fait à partir de données brutes; La prise de décision du pilote n'est pas aidée par le système. La communication est réalisée manuellement.	Très bas

Figure 20. Niveaux d'automatisation d'après les travaux de (Billings, 1991)

3.2.4 Niveaux d'automatisation pour les voitures autonomes

(C. D. Johnson et al., 2017) soulignent que les niveaux d'automatisation, par leur définition haut niveau de l'allocation des fonctions et des tâches, n'adressent pas le niveau de détail des décisions allouées au système contenant de l'automatisation ou quelles décisions restent allouées à l'utilisateur. Leurs travaux proposent des niveaux d'abstraction de contrôle humain (LHCA) pour décrire le niveau de détail des entrées de contrôle de l'utilisateur dans un véhicule ou un télé-robot. Ces niveaux de contrôle sont : le contrôle direct (l'utilisateur contrôle tous les aspects du système), le contrôle augmenté (l'utilisateur entre les actions que le système exécute), le contrôle paramétré (l'utilisateur entre les paramètres désirés), le contrôle orienté but (l'utilisateur entre les buts un à un au fur et à mesure de la mission), le contrôle mission-capable (l'utilisateur entre tous les buts de la mission).

La (SAE J3016, 2019) définie 5 niveaux d'automatisation de la conduite pour les voitures contenant de l'automatisation. Ces niveaux d'automatisation de la conduite sont présentés dans la Figure 21. Ces niveaux d'automatisation décrivent différentes allocations de tâches et de fonctions entre l'utilisateur et la voiture pour la tâche dynamique de conduite (Dynamic Driving Task : DDT) :

- Contrôle de mouvement latéral du véhicule,
- Contrôle du mouvement longitudinal du véhicule (accélération et décélération),

- Surveillance de l'environnement de conduite (détection d'objets et d'événements, reconnaissance, classification et préparation de la réponse),
- Exécution de la réponse,
- Planification des manœuvres,
- Amélioration de la visibilité via l'éclairage, la signalisation, les gestes, etc.

De plus, la SAE J3016 spécifie un ensemble de tâche sous la dénomination de détection d'objet et d'évènement et réponse (*Object and Event Detection and Response* : OEDR). Également, la SAE J3016 spécifie qu'une tâche peut être limitée à un domaine de conception opérationnel (*Operational Design Domain* : ODD) ne pouvant être dépassé. Par exemple, il peut être spécifier qu'une partie de la tâche de conduite ne peut être réalisée par le véhicule seulement si le marquage au sol est net et la météo bonne.

Les niveaux d'automatisations de la conduite définis par la SAE J3016 sont :

- Niveau 0 : Le conducteur humain réalise toute la tâche dynamique de conduite.
- Niveau 1 : Un système avancé d'assistance à la conduite (ADAS) sur le véhicule peut parfois aider le conducteur humain à diriger ou à freiner / accélérer, mais pas les deux simultanément.
- Niveau 2 : Un système avancé d'assistance à la conduite (ADAS) sur le véhicule peut lui-même contrôler à la fois la direction et le freinage / accélération simultanément dans certaines circonstances. Le conducteur humain doit continuer à porter toute son attention (« surveiller l'environnement de conduite ») et effectuer le reste de la tâche dynamique de conduite.
- Niveau 3 Un système de conduite automatisée (ADS) sur le véhicule peut lui-même exécuter tous les aspects de la tâche de conduite dans certaines circonstances. Dans ces circonstances, le conducteur humain doit être prêt à reprendre le contrôle à tout moment lorsque l'ADS demande au conducteur humain de le faire. Dans toutes les autres circonstances, le conducteur humain exécute la tâche dynamique de conduite.
- Niveau 4 : Un système de conduite automatisée (ADS) sur le véhicule peut lui-même effectuer toutes les tâches de la tâche dynamique de conduite et surveiller l'environnement de conduite dans certaines circonstances. L'humain n'a pas besoin de prêter attention dans ces circonstances.
- Niveau 5 : Un système de conduite automatisée (ADS) sur le véhicule peut effectuer toute la tâche dynamique de conduite en toutes circonstances. Les occupants humains ne sont que des passagers et n'ont jamais besoin d'être impliqués dans la tâche dynamique de conduite.

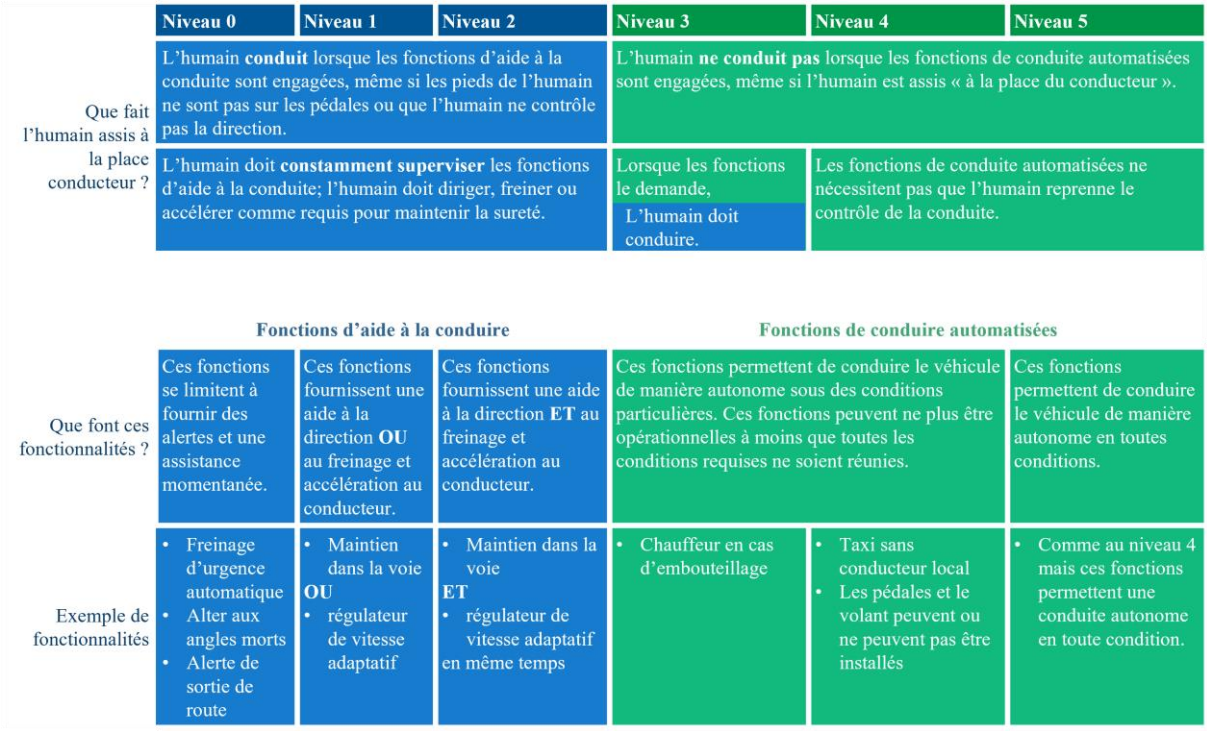


Figure 21. Niveaux d'automatisation de la conduite de la SAE J3016

Les niveaux d'automatisation de la conduite de la (SAE J3016, 2019) diffèrent de ceux présentés dans les sections précédentes par le niveaux d'abstraction employé pour la description des tâches et des fonctions. En effet, différents niveaux d'une tâche particulière (la tâche dynamique de conduite) sont décrits contrairement aux autres niveaux d'automatisation présentés applicables à différentes tâches à réaliser dans le domaine d'application concerné.

Également, ces niveaux d'automatisation décrivent des transitions possibles de niveaux d'automatisation lors de l'exécution du système. Une partie du trajet peut être réalisée en conduite partiellement autonome (par exemple en niveau 3) et une autre partie du trajet peut être réalisée avec l'utilisateur réalisant une partie de la tâche dynamique de conduite (par exemple en niveau 2).

3.3 Allocation dynamique des tâches et des fonctions

Avec le progrès technologique, il a été possible de concevoir des systèmes dont l'allocation des fonctions pouvait changer au cours de leur exécution pour prendre en compte des modifications du contexte par exemple. (Rouse, 1988) désigne ce type de systèmes comme des systèmes dont l'automatisation s'adapte aux besoins des utilisateurs laissant tout le contrôle aux l'utilisateurs lorsque ceux-ci n'ont pas besoin d'aide. Ces systèmes appelés « systèmes adaptatifs » sont des systèmes pour lesquels l'allocation des fonctions et des tâches entre l'humain et le système est flexible et réactive aux demandes des tâches ou des utilisateurs. Les travaux de (Raja Parasuraman et al., 1996) montre que l'allocation de tâches et de fonctions dynamique permet une meilleure performance des utilisateurs pour des tâches de supervision en simulation de vol. Conformément à cette étude, (Miller & Parasuraman, 2007) proposent que les niveaux d'automatisation soit ajustables durant l'exécution du système. (P. A. Hancock & Chignell, 1987) proposent un cadre conceptuel afin de concevoir de tels systèmes adaptatifs. Les « Fitts

lists » qui définissent une allocation statique des fonctions et des tâches pour toute l'exécution du système ne sont pas applicables à ce type de système.

Par la suite, d'autres méthodes de conception incrémentales et itératives de l'automatisation ont été proposées. Parmi ces méthodes, nous pouvons citer les travaux de (Boy, 1998) qui proposent une méthodologie d'analyse et d'allocation des tâches et fonctions dans un processus de conception centrée utilisateur de l'automatisation des systèmes interactifs critiques.

3.4 Autres approches pour l'allocation de tâches et de fonctions

Nous présentons ci-après, deux autres approches qui, plutôt de décrire des allocations de tâches et de fonctions pour différent niveaux d'automatisation, proposent :

- De décrire l'allocation de tâche et de fonctions en décrivant quel rôle réalise quel aspect d'une tâche ou fonction
- D'utiliser une métaphore pour guider les choix d'allocation de tâches et de fonctions.

3.4.1 IDA-S (Andrew Dearden, 2001)

Les travaux de (Andy Dearden et al., 2000) proposent une approche d'allocation des fonctions et des tâches utilisant des scénarios afin de prendre en compte le contexte d'usage. La méthode proposée est présentée en Figure 22. Les fonctions candidates à une automatisation partielle sont étudiées selon le coût en recherche et développement et les bénéfices (ou désavantages) qu'elles offrent en termes de charge cognitive, performance et de *situation awareness* pour les utilisateurs.

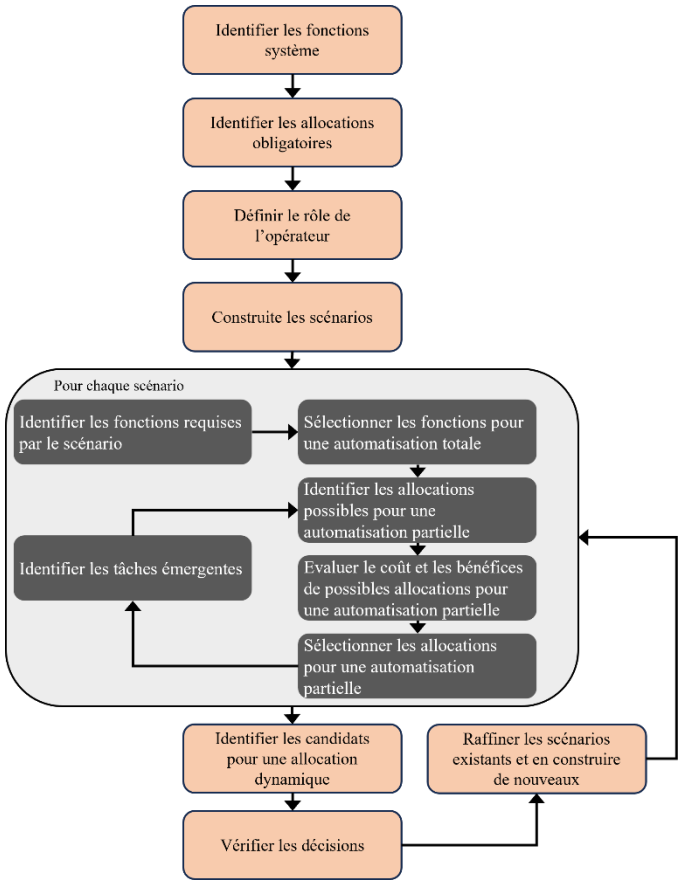


Figure 22. Processus d'allocation de fonctions et de tâches proposé par (Andy Dearden et al., 2000)

L'approche IDA-S proposée par (Andrew Dearden, 2001) propose de décomposer chaque tâche en Information, Décision, Action et Supervision. Chacun de ces éléments sont ensuite également décomposés pour décrire chacun des aspects de la tâche. Par exemple, Information est décomposé en collecte, intégration, configuration et initiation de la réponse pour planifier la réponse. Le cadre conceptuel IDA-S est présenté en Figure 23. Le concepteur doit spécifier quel rôle réalise quel aspect de chaque tâche. C'est-à-dire, le concepteur doit remplir les cases vacantes de la Figure 23 avec le rôle réalisant l'aspect de la tâche. Lorsqu'aucun rôle ne peut être alloué, le concepteur note « n/a ».

(Harrison et al., 2003) propose une extension de IDA-S utilisant une description UML afin de rendre l'approche IDA-S utilisable par des ingénieurs système.

	Information		Décision		Action	
Planification de la réponse	Collecter		Proposer		Approuver	
	Intégrer		Evaluer			
	Configurer		Modifier			
	Initier une réponse		Sélectionner			
Superviser l'exécution en cours	Monitorer/Surveiller le progrès		Identifier les exceptions		Révoquer l'autorité	
Superviser la terminaison	Déterminer le contenu en sortie		Identifier l'achèvement		Stopper le processus	
Action	Exécuter les actions					

Figure 23. Le template IDA-S

3.4.2 Métaphore du cheval (F. Flemisch et al., 2005)

Une autre approche proposée afin de concevoir l'automatisation des systèmes est la métaphore du cheval (H-Metaphor) proposée par (F. Flemisch et al., 2005). Cette métaphore a été proposée pour concevoir les voitures autonomes soutenant les tâches de l'utilisateur comme un cheval bien dressé, proposant une interface multimodale, avec un niveau d'autonomie similaire à celle d'un cheval.

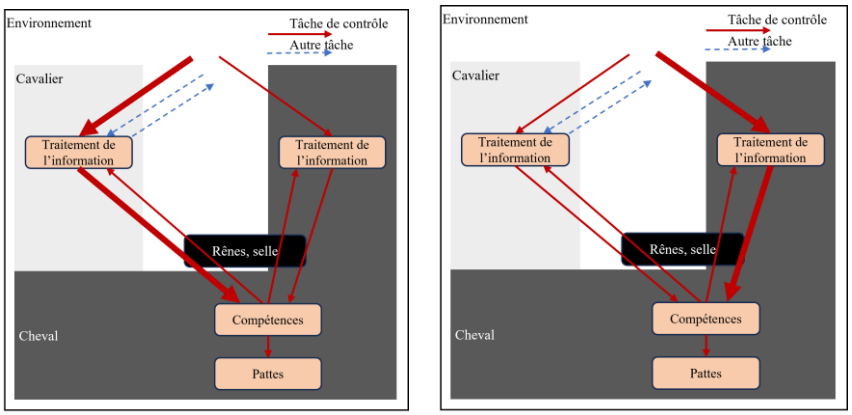


Figure 24. Illustrations de la métaphore du cheval utilisé pour concevoir les contrôles physiques d'une voiture d'après les travaux de (F. Flemisch et al., 2005). a) Serrer les rênes pour contrôler le cheval. b) Relâcher les rênes pour donner le contrôle au cheval.

4 Allocation des ressources, de l'autorité, de la responsabilité et transitions de contrôle

Les tâches allouées à l'utilisateur et les fonctions allouées au système contenant de l'automatisation peuvent décrire : un échange de ressources, une autorité, une responsabilité ou initier une transition de contrôle.

4.1 Contrôle, Autorité, Responsabilité et transitions de contrôle

L'allocation du contrôle entre l'utilisateur et le système peut être définit comme l'allocation des tâches et des fonctions qui contribuent à l'atteinte du but de l'utilisateur ou du système en modifiant l'état du système, d'un objet ou de l'environnement.

4.1.1 Contrôle

Selon (Frank Flemisch et al., 2012), le contrôle est le fait d'influencer la situation pour la faire évoluer ou la maintenir dans un sens voulu par l'entité possédant le contrôle. Le contrôle de la situation est la boucle perception-action pour maintenir ou faire évoluer la situation pour atteindre un but de l'acteur possédant le contrôle. Le contrôle est donc l'ensemble des tâches et sous tâches allouées à l'utilisateur ou au système pour atteindre son ou ses buts.

4.1.2 Autorité

(Frank Flemisch et al., 2012) définit l'autorité d'un utilisateur ou d'un système comme ce que l'utilisateur ou le système est autorisé à faire ou non. (Frank Flemisch et al., 2012) définit également deux niveaux d'autorité :

- Autorité de contrôle (*Control Autority*) : exercer une certaine distribution du contrôle.
- Autorité de modification du contrôle (*Control Change Authority*) : modifier la distribution de l'autorité entre l'utilisateur et le système pour une autre en donnant plus ou moins d'autorité à l'utilisateur ou au système.

(D. W. Lee & Bass, 2015) propose une définition de l'autorité similaire au premier niveau d'autorité définit par (Frank Flemisch et al., 2012). (D. W. Lee & Bass, 2015) définissent l'autorité comme le travail (les tâches ou les fonctions) que le système ou l'utilisateur est chargé de réaliser. (Sarter & Woods, 1997) définissent l'autorité du système contenant de l'automatisation comme sa capacité à moduler et remplacer les entrées de l'utilisateur dans le domaine de l'aviation.

De nombreux autres travaux, dont (T. B. Sheridan, 2011), (Jansson et al., 2014) et (Young et al., 2007), soulignent l'importance de la prise en compte de l'allocation de l'autorité lors de la conception de l'allocation de l'autorité sans pour autant donner une définition explicite de ce terme.

4.1.3 Responsabilité

Les travaux de (Boy, 2013) définissent l'autorité comme le contrôle (qui réalise et qui est compétent pour une tâche ou une fonction donnée) et la responsabilité. La responsabilité y est définie au sens légal, c'est-à-dire être responsable (*accountable*) du résultat d'une tâche ou d'une fonction. (Frank Flemisch et al., 2012) identifie aussi la responsabilité comme caractérisant les tâches et les fonctions du système contenant de l'automatisation et de l'utilisateur. (Frank Flemisch et al., 2012) considère que la responsabilité est allouée à l'avance pour motiver certaines actions. L'allocation de la responsabilité permet d'évaluer à la fin du travail qui de l'utilisateur ou du système contenant de l'automatisation est responsable (*accountable*) pour un état, une action et les conséquences de celle-ci. De manière similaire, (D. W. Lee & Bass, 2015) définissent la responsabilité comme le travail (tâches ou fonctions) dont le système contenant de l'automatisation ou l'utilisateur est responsable (*accountable*) et

qui n'est pas nécessairement réalisé par ce système ou cet utilisateur. Au-delà de la responsabilité allouées lors de l'allocation des tâches et des fonctions, les travaux de (Awad et al., 2018) questionnent l'allocation de la responsabilité perçue entre le conducteur et le véhicule autonome en cas d'accident par les participants de l'étude. Les résultats montrent, par exemple, que la responsabilité perçue pour le véhicule est moins importante que celle perçue pour le conducteur lorsqu'une erreur a été commise, à la fois, par le conducteur et le véhicule.

4.1.4 Transitions de contrôle de la tâche de conduite avec un véhicule autonome

Le deuxième niveau d'autorité définit par (Frank Flemisch et al., 2012) : autorité de modification du contrôle, est semblable à la notion de transitions de contrôle pour la tâche de conduite avec une voiture autonome .

Dans le domaine de l'automatisation dans l'automobile, une transition de contrôle est définie par (Lu et al., 2016) comme un processus pendant lequel le couple conducteur-véhicule passe d'un état de conduite à un autre. Par exemple, après une transition de contrôle, le conducteur peut avoir le contrôle longitudinal (vitesse) et latéral (direction) du véhicule alors que ce contrôle était alloué au véhicule avant la transition de contrôle. Nous pouvons donc définir une transition de contrôle comme le passage d'une allocation de tâches et de fonctions à une autre durant l'exécution du système.

(Lu et al., 2016) propose un arbre de classification des transitions de contrôle présenté en Figure 25. Une transition de contrôle peut être classifiée selon :

- Qui du conducteur ou du véhicule initie la transition de contrôle,
- Qui du conducteur ou du véhicule a le contrôle après la transition de contrôle et
- Si la transition de contrôle est obligatoire ou optionnelle.

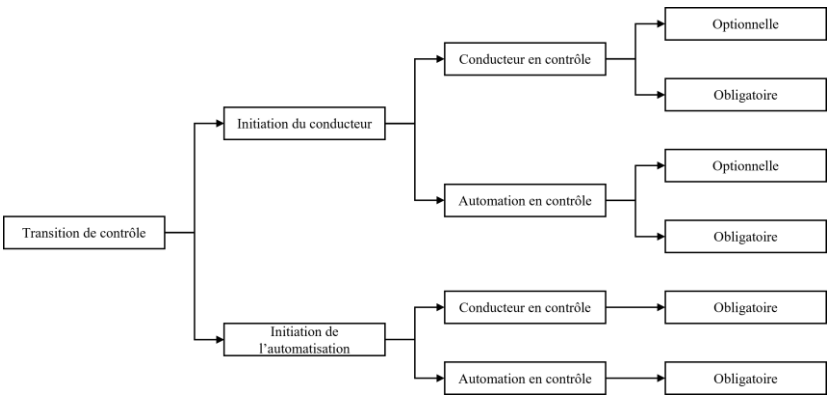


Figure 25. Arbre de classification des transitions de contrôle d'après les travaux de (Lu et al., 2016)

Il existe d'autres taxonomies centrées sur une transition de contrôle particulière : le *handover*. (Walch et al., 2015) définit un *handover* comme une transition où une tâche de conduite précédemment totalement allouée au véhicule autonome est allouée au conducteur (conduite manuelle). La transition inverse est appelée *takeover*. (Walch et al., 2015) distinguent différents *handovers* :

- Les *handovers* immédiats : par exemple, lorsque le conducteur saisit le volant

- Le transfert de contrôle par étapes : par exemple, le contrôle longitudinal est remis au conducteur, puis ensuite, le contrôle latéral
- La transition monitorée par le conducteur : par exemple, le conducteur saisit le volant et après un compte à rebours, le contrôle est remis au conducteur
- La transition monitorée par le système : par exemple, le système monitore les entrées du conducteur après la transition de contrôle

(Roderick McCall et al., 2016) et (Rod McCall et al., 2019) distinguent les *handover* suivants :

- Planifiée : la transition de contrôle est planifiée pour le conducteur et le véhicule à un moment définit
- Non planifiée : la transition de contrôle n'est pas planifiée à l'avance pour le conducteur et le véhicule. Une transition de contrôle non planifiée peut être :
    - Initiée par le système
        - D'urgence (éviter un danger) ou non
    - Initiée par le conducteur
        - D'urgence (éviter un danger) ou non

(Wintersberger et al., 2017) considèrent qu'il existe trois types de transitions de contrôle :

- Les transitions de contrôle planifiées,
- Les transitions de contrôle imminentes et
- Les transitions de contrôle initiées par le conducteur.

(Wintersberger et al., 2017) considère que les transitions programmées et imminentes sont initiées par l'automatisation. Les transitions planifiées ont l'avantage que le système les connaît à l'avance, ce qui permet une meilleure planification de la transition. (Wintersberger et al., 2017) propose une taxonomie des *handovers* décomposant la séquence de transfert du contrôle de la tâche de conduite du véhicule vers le conducteur. De la même manière, (Wintersberger et al., 2017) propose une taxonomie des *takeovers* décomposant la séquence de transfert du contrôle de la tâche de conduite du conducteur vers le véhicule.

Ces différentes taxonomies des transitions de contrôle nous apprennent qu'une transition de contrôle peut être caractérisée par :

- Quel rôle initie la transition de contrôle
- Les contraintes temporelles de la transition de contrôle (immédiate, planifiée)
- Le caractère obligatoire ou optionnel de la transition de contrôle
- La transition de l'allocation d'une tâche complète ou par étapes du système vers l'utilisateur ou de l'utilisateur vers le système

4.2 Allocation de ressource : transparence

Les mécanismes de retour du système contenant de l'automatisation influencent ce qui est observable à propos de ce système par les utilisateurs (Woods, 1996). Ce qui est observable par les utilisateurs influence leur anticipation du comportement du système contenant de l'automatisation. L'observabilité du système est défini par (Sarter & Woods, 1997) comme la

capacité du retour du système à soutenir activement les utilisateurs qui monitorent ce système et à anticiper les tâches et les transitions de ce système. Similairement au concept d'observabilité, (Klien et al., 2004) identifient, parmi leur dix challenges pour faire du système contenant de l'automatisation un membre d'une équipe, la nécessité, autant pour le système que les utilisateurs, de correctement modéliser les intentions des autres membres, d'être mutuellement prédictibles et de pouvoir observer et interpréter les états et les intentions des autres membres.

(Jansson et al., 2014) propose le modèle GMOC (*Goals, (mental) Models, Observability, Controllability*) pour la conception de systèmes contenant de l'automatisation. Ce modèle est présenté en Figure 26. L'observabilité désigne ce que l'utilisateur perçoit de l'état du système et de l'environnement. L'observabilité du système influence la construction du modèle mental du système par l'utilisateur.

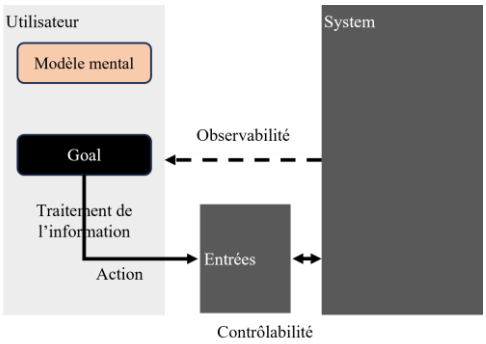


Figure 26. Modèle GMOC d'après les travaux de (Jansson et al., 2014)

L'observabilité est désignée dans les travaux plus récents sous le terme de **transparence** du système. La transparence est défini par (Mercado et al., 2016) comme la qualité d'une interface d'un système contenant de l'automatisation relative à sa capacité à permettre à un utilisateur de comprendre l'intention, les performances, les futurs plans d'action et le processus de raisonnement du système contenant de l'automatisation.

La transparence est un élément du modèle HAT (*Human-Automation Teaming*) proposé par (Battiste et al., 2018). Les trois éléments du modèle HAT sont : la transparence, la communication bidirectionnelle et une exécution dirigée par l'utilisateur. Ce modèle est implémenté par les auteurs pour concevoir un système soutenant les opérations mono-pilote pour des avions de transport commercial en coopération avec un agent d'opération au sol.

5 Conclusion

Dans ce chapitre, nous avons définit l'automatisation pour les systèmes contenant de l'automatisation. Nous avons présenté deux propriétés propres à ces systèmes : *trust* et *situation awareness*. Ensuite, nous avons présenté les techniques et méthodes proposées pour concevoir des systèmes contenant de l'automatisation garantissant ces propriétés. Bien que les différents éléments à prendre en compte pour concevoir l'automatisation aient été identifiés, il n'existe pas de méthode permettant de prendre en compte à la fois l'allocation des tâches et des fonctions, des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Peu d'outils sont fournis aux ingénieurs et développeurs afin de prendre en compte

ces différents éléments lors de la conception du système. A l'exception, des travaux (Harrison et al., 2003) qui proposent une extension de IDA-S utilisant une description UML. Cependant, cette approche ne permet pas de modéliser explicitement l'allocation de l'autorité, de la responsabilité, des transitions de contrôle et de l'initiative et les ressources partagées. De plus, il est nécessaire d'identifier, d'analyser et de décrire les tâches pour réaliser l'allocation de fonctions et de tâches entre les utilisateurs et les systèmes contenant de l'automatisation. Nous présentons au chapitre suivant les modèles, notations et outils de description des tâches utilisateur.

Chapitre 3. Modélisation de l'humain et de ses tâches

Notre but est de fournir un soutien à la conception de systèmes interactifs contenant de l'automatisation soutenant les buts des utilisateurs et répondant à leurs besoins. A cela, s'ajoute la prise en considération de propriétés propres aux systèmes interactifs contenant de l'automatisation que nous avons présenté dans le chapitre précédent. Pour cela, nous nous intéresserons plus spécifiquement à l'analyse des tâches et fonctions de l'humain et du système pour l'atteinte d'un but commun.

Dans ce chapitre nous présentons différentes approches de la modélisation de l'humain et de ses tâches en tant qu'utilisateur d'un système interactif.

Dans la première section, nous présentons les modèles de l'humain décomposant ses tâches. Nous présentons d'abord des approches présentant spécifiquement les tâches humaines. Puis, nous présentons des décompositions de tâches dont les différentes étapes peuvent être allouées soit à un utilisateur, soit à un système contenant de l'automatisation.

Dans la seconde section, nous présentons différentes techniques d'analyse, de notation et de modélisation des tâches utilisateurs.

Dans la troisième section, nous présentons la notation HAMSTERS et l'outil associé en détail que nous utilisons dans la suite de cette thèse.

La quatrième section conclut ce chapitre.

1 Modèles de l'Humain : décomposition des tâches

Dans cette section, nous présentons des modèles de décomposition des tâches humaines. Dans la première sous-section, nous présentons des décompositions de tâches propres à l'humain. Dans la seconde sous-section, les décompositions présentées permettent d'allouer les différentes étapes d'une tâche à l'humain ou au système.

1.1 Décomposition des tâches humaines

Différents modèles de l'humain décomposant les tâches humaines ont été proposés pour soutenir la conception de systèmes interactifs. Nous présentons un ensemble de ces modèles dans cette section.

1.1.1 Le modèle SRK

La Figure 27 présente le modèle SRK de (Rasmussen, 1983). Ce modèle a été proposé afin de prédire les performances humaines lors de l'utilisation de systèmes.

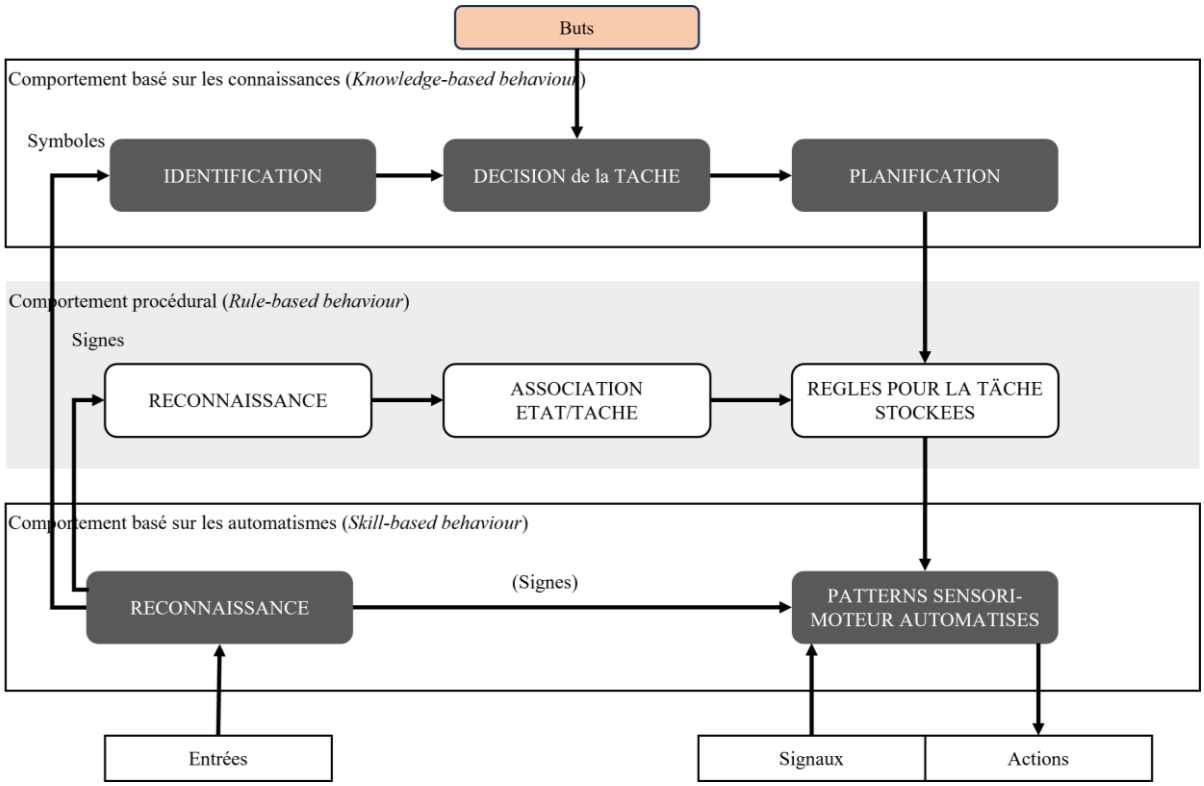


Figure 27. Illustration simplifiée des trois niveaux de performance d'utilisateurs qualifiés d'après les travaux de (Rasmussen, 1983)

Ce modèle définit trois types de comportement correspondant au niveau de performance d'un utilisateur :

- Comportement basé sur les automatismes : Ce comportement représente la performance sensori-motrice lors d'actes qui se déroulent sans contrôle conscient de l'utilisateur. Les patterns de comportement sont lisses, automatisés et hautement intégrés. Il s'agit de configurations mémorisées, d'instructions préprogrammées. Faire du vélo est un exemple de comportement basé sur les automatismes.
- Comportement basé sur les règles : Ce comportement représente la performance pour des actes d'un travail familier, contrôlé par une règle mémorisée ou une procédure qui ont pu être appliquées à des situations antérieures. La réalisation d'une recette de cuisine est un exemple de comportement basé sur les règles.
- Comportement basé sur les connaissances : Ce comportement représente la performance pour des actes dans les situations nouvelles pour lesquelles les actions doivent être planifiées en temps réel avec un contrôle conscient, en s'appuyant sur des processus analytiques et des connaissances mémorisées. L'utilisateur construit un modèle mental du comportement du système.

1.1.2 Le modèle du processeur humain

La Figure 28 présente le modèle du processeur humain de (S. Card et al., 1986). Le modèle du processeur humain offre un support pour le calcul de la performance de l'humain lorsqu'il utilise un système interactif.

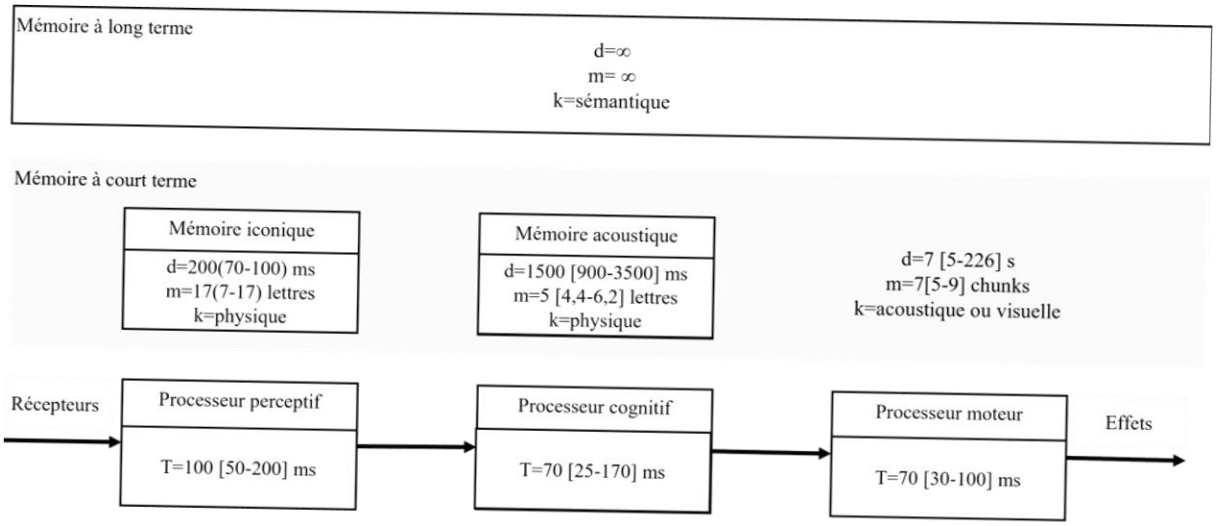


Figure 28. Le modèle du processeur humain d'après les travaux de (S. Card et al., 1986)

Le modèle du processeur humain est composé d'un ensemble de processeurs, de mémoires, d'interconnexions et d'un ensemble de principes (Loi de Fitts (P. M. Fitts, 1954) , le principe de discrimination, etc.). Les trois sous-processeurs du processeur humain sont :

- Le processeur sensoriel : Chacun des processeurs est spécialisé dans le traitement d'une classe de stimuli. L'information perçue est stockée en mémoire de travail.
- Le processeur cognitif : L'information stockée en mémoire de travail est traitée par le processeur cognitif selon le cycle de reconnaissance-action. Le contenu de la mémoire de travail est associé avec des éléments stockés dans la mémoire à long terme. La décision d'agir est prise dans le processeur cognitif.
- Le processeur moteur : L'action est contrôlée par le processeur moteur. L'action motrice n'est pas continue mais est composée d'une série de micromouvements.

Une opération parallèle de ces trois sous-systèmes est possible comme, par exemple, des tâches consistant à taper et à lire du texte en même temps.

1.1.3 La théorie de l'action de Norman

Selon la théorie de l'action, une action utilisateur peut être décomposée en 7 étapes : la formalisation d'un but, l'établissement d'une intention, la spécification d'une séquence d'action, l'exécution de ces actions, la perception du changement d'état, l'interprétation de la perception et l'évaluation par rapport au but.

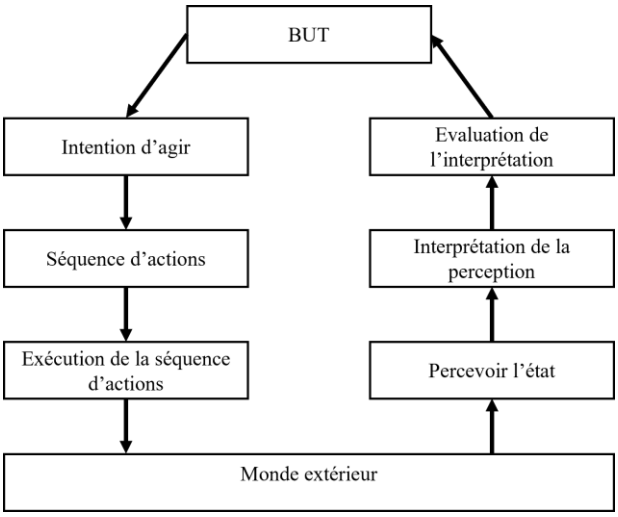


Figure 29. Les sept étapes de la théorie de l'action (Norman, 2013)

La Figure 29 décrit les 7 étapes de la théorie de l'action présentée dans (Norman, 2013).

La théorie de l'action met en valeur deux types de distance :

- La distance d'évaluation entre l'état du système et l'évaluation de l'état du système par l'utilisateur en fonction de ses buts.
- La distance d'exécution entre les buts de l'utilisateur et l'exécution d'une séquence d'action par l'utilisateur sur le système.

La description des tâches utilisateur de différents sous-types est donc nécessaire pour pouvoir : évaluer ces deux types de distances, essayer de les réduire, ou trouver des solutions aux étapes problématiques.

1.2 Décomposition des tâches ou fonctions allouables à un humain ou un système contenant de l'automatisation

Différentes décompositions des tâches et des fonctions ont été proposées pour soutenir la conception de systèmes contenant de l'automatisation centrée humaine. Nous en présentons un sous-ensemble dans cette sous-section.

1.2.1 Perception, cognition et action (Task-Agent Control Space)

(Shoval et al., 1993) propose une méthode pour l'allocation de tâches appelée processus TASK-AGENT-CONTROL (TAC). Ce processus est présenté dans la Figure 30 dans l'espace d'état TAC. Ce processus divise sur l'axe TASK les tâches en sous tâches :

- Perception
- Cognition
- Actions

Une flèche indique le flux de l'information dans l'ordre chronologique. En général, une tâche perceptive acquière l'information, cette information est traitée par une tâche cognitive qui résulte sur une ou des actions à réaliser.

L'axe AGENT indique à quel agent est allouée la tâche (un système ou un humain). L'axe CONTROL défini qui est responsable de l'allocation de cette tâche. Cet axe ne définit pas seulement que l'allocation peut être réalisée par un agent spécifique mais aussi qui a l'autorité de modifier cette allocation.

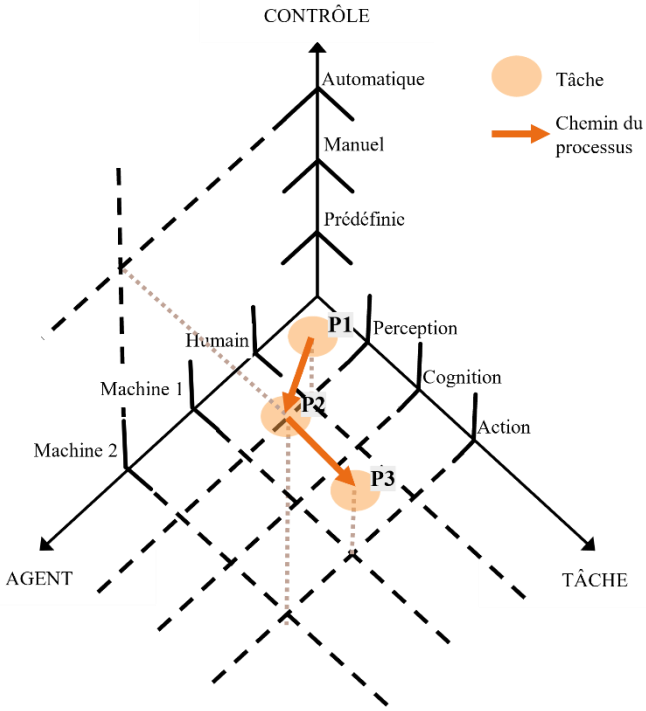


Figure 30. Model du processus dans l'espace d'état TASK-AGENT-CONTROL (TAC)

1.2.2 Information, décision et action (IDA Cycle model)

(Semple, 1998) propose de décrire les opérations de missions réalisées par des pilotes avec un outil d'assistance à l'équipage (*crew assisting*) en : acquisition d'information, décision du plan d'actions correct et réalisation les actions :

- Information : Cette étape représente l'acquisition de l'information. Cette étape requiert parfois la réalisation des étapes suivantes de IDA pour savoir quelles sont les informations requises et comment les acquérir.
- Décision : Cette étape représente la prise de décision. Certaines décisions peuvent demander plus d'effort de la part de l'utilisateur et de temps que d'autres mais il peut s'avérer que cet effort et ce temps supplémentaire soit la cause du traitement de l'information avant la prise de décision.
- Action : Cette étape peut décrire une action simple comme lever une alerte comme une action complexe et prolongée comme la construction d'une base ou l'atterrissage d'un avion. Plus la tâche est complexe, plus elle est décomposée selon les étapes du cycle IDA.

Le modèle cyclique IDA propose de décrire toutes les tâches militaires comme des cycles enchainés, imbriqués et se chevauchant de collecte d'information, de prise de décisions sur les informations acquises et de mise en œuvre de ces décisions.

1.2.3 Monitoring, génération, sélection et implémentation

Les 10 niveaux d'automatisation de (Kaber & Endsley, 2004) sont applicables à des tâches requérant de réaliser un contrôle en temps réel du ou des systèmes comme : le contrôle de trafic aérien, le pilotage d'avion, la télé-opération et la manufacture. Selon (Kaber & Endsley, 2004), ces domaines ont en commun la réalisation de buts multiples en compétition, de tâches en compétition pour l'attention de l'utilisateur, chacune avec une pertinence différente selon le système et une demande importante de tâches avec une ressource de temps limitée.

Quatre fonctions ou tâches génériques intrinsèques à ces domaines sont identifiées par (Kaber & Endsley, 2004) :

- Monitoring *(Monitoring)* qui inclus l'acquisition de toutes les informations pertinentes pour percevoir l'état du système (ex. scanner les affichages visuels).
- Génération *(Generating)* est la formulation d'options ou de stratégies pour les tâches pour accomplir des buts.
- Sélection *(Selecting)* est la décision d'une option ou d'une stratégie particulière.
- Implémentation *(Implementing)* est la réalisation de l'option choisie via des actions de contrôle sur l'interface.

1.2.4 Information, décision, action et supervision (IDA-S)

(Andrew Dearden, 2001) propose de caractériser une fonction (ou tâche) selon les trois étapes de Semple (Semple, 1998) auxquelles le composant « Supervision » a été ajouté. Ces quatre composants sont :

- Information : Ce composant contient les tâches réalisant la collecte et l'intégration des informations, la configuration de la résolution de problème (ex. changer de page sur un affichage particulier) et l'initiation d'une action.
- Décision : Ce composant contient les tâches réalisant la proposition d'évaluation (proposition d'options et l'évaluation des options), la modification d'une sélection, la sélection ou l'approbation d'une sélection.
- Action : Ce composant décrit une action décomposable selon les composants de IDA-S.
- Supervision : Ce composant contient les tâches réalisant le monitoring des opérations, l'identification des exceptions, l'actualisation l'allocation de l'autorité, la détermination le contenu produit, la détection la fin du travail et la terminaison du processus de travail.

1.2.5 Le modèle simple en quatre étapes du traitement de l'information par l'humain et par le système contenant de l'automatisation

(R. Parasuraman et al., 2000) propose de décomposer une tâche selon un modèle simple en quatre étapes du traitement de l'information présenté en Figure 31 :

- Traitement sensoriel (*Sensory Processing*) : Cette étape décrit les tâches réalisant l'acquisition et l'enregistrement de multiple sources d'information. Les tâches de positionnement et d'orientation des capteurs, de traitement sensoriel et de traitement de l'information avant la perception et l'attention sélective font partie de cette étape.

- Perception/Mémoire de travail (*Perception/Working Memory*) : Cette étape décrit les tâches réalisant la perception consciente de l'information et la manipulation de l'information traitée dans la mémoire de travail. Les tâches de répétition, d'intégration et d'inférence font partie de cette étape mais ont lieu avant le point de prise de décision.
- Prise de décision (*Decision Making*) : Cette étape décrit la décision réalisée après le traitement cognitif de l'information de l'étape précédente.
- Sélection de la réponse (*Response Selection*) : Cette étape décrit l'implémentation de la réponse ou de l'action cohérente avec la décision de l'étape précédente.

Ces différentes étapes sont coordonnées ensemble dans les cycles de perception-action lorsque l'utilisateur interagit avec le système.

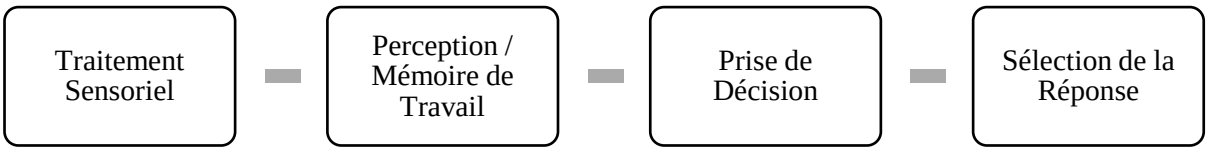


Figure 31. Modèle simple en 4 étapes du traitement de l'information par l'humain

Le modèle simple en quatre étapes du traitement de l'information par l'humain possède son équivalent en fonctions systèmes automatisant les différentes étapes. Cette automatisation peut être appliquée en quatre classes de fonctions présentées en Figure 32 :

- Acquisition de l'information (*Information acquisition*) : Cette étape décrit les fonctions qui acquièrent et enregistrent des données. Les fonctions consistant à modifier le positionnement des capteurs ou à organiser les données en liste, en mettant en valeur ou en filtrant certaines données font partie de cette étape.
- Analyse de l'information (*information analysis*) : Cette étape décrit les fonctions cognitives comme les fonctions de mémoire de travail ou d'inférence. Les fonctions d'extrapolation de données, de prédictions ou d'intégration font partie de cette étape.
- Sélection de la décision (*decision and action selection*) : Cette étape décrit les fonctions de sélection à partir de plusieurs alternatives de décisions et d'actions. Les fonctions recommandant certaines décisions ou sélectionnant une décision font partie de cette étape.
- Implémentation de l'action (*action implementation*) : Cette étape décrit les fonctions exécutant l'action choisie. Les fonctions remplaçant la main ou la voix de l'utilisateur ou des tâches utilisateurs plus complexe comme l'atterrissage d'un avion font partie de cette étape.

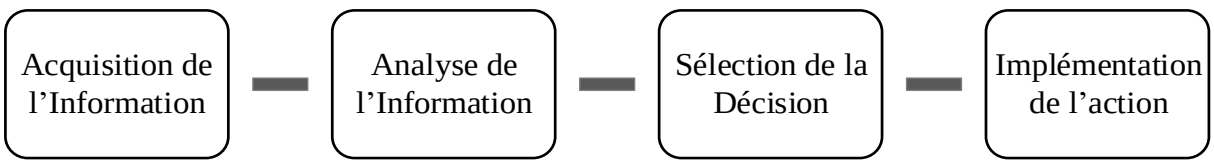


Figure 32. Modèle simple en 4 étapes du traitement de l'information par le système.

Chacune des étapes des modèles simples en 4 étapes du traitement de l'information peut être caractérisée par son niveau d'automatisation (voir Chapitre 2.3.2).

1.3 Décomposition des tâches coopératives

Il existe des tâches réalisées en coopération avec un ou plusieurs utilisateurs et un ou plusieurs systèmes, entre utilisateurs ou entre systèmes. (Battiste et al., 2018) soulignent que la coopération et la coordination entre le système et l'utilisateur est importante pour l'atteinte du but commun. La conception des tâches et fonctions isolées de l'utilisateur et du système comme le propose les « Fitts Lists » a rapidement été identifiée comme un des principaux défauts de ces listes (Dekker & Woods, 2002). La conception de la coopération utilisateur-système contenant de l'automatisation est recommandée par de nombreuses contributions. Parmi ces contributions, nous pouvons citer (Woods, 1996), (Klien et al., 2004) ou (Bradshaw et al., 2013).

Dans le domaine de recherche des collecticiels, les tâches ou fonctions coopératives et collaboratives sont définies comme telles :

- Tâche coopérative :
    - Une tâche coopérative est accomplie en divisant le labeur entre les acteurs, comme un travail où chaque acteur est responsable d'une portion de la résolution du problème (Roschelle & Teasley, 1995).
    - Une tâche coopérative est réalisée par des petits groupes d'utilisateurs partageant des objectifs clefs (Grudin, 1994).
- Tâche collaborative :
    - Une collaboration est un travail coordonné, synchrone qui est le résultat d'une tentative continue de construire et de maintenir une conception partagée d'un problème (Roschelle & Teasley, 1995).
    - Une tâche collaborative est réalisée par de grandes organisations, où les objectifs et/ou les buts généralement diffèrent ou même sont en conflits (Grudin, 1994).

Également, les travaux de (Victor M. R. Penichet, Lozano, & Gallud, 2009) définissent une interaction coopérative comme une relation qui prend place entre des groupes, des rôles ou des acteurs et exprime une interaction entre eux.

Les travaux de (Ellis et al., 1991) définissent qu'une tâche coopérative ou collaborative est réalisée avec les contraintes spatio-temporelles présentées dans le Tableau 3. Ces contraintes spatio-temporelles décrivent si les acteurs se trouvent au même endroit ou non, s'ils interagissent ensemble dans la même fourchette temporelle ou non.

Tableau 3. Matrice espace-temps du collecticiel

	Au même moment (synchrone)	A différents moments (asynchrone)
Même localisation	Interaction face-à-face	Interaction asynchrone
Localisation différente	Interaction distribuée synchrone	Interaction distribuée asynchrone

En se basant sur les travaux de (Ellis et al., 1991), (Salber et al., 1995) définit les espaces du collecticiel : le trèfle production, communication et coordination.

- Communication : La communication est l'échange de contenu sémantique entre les acteurs d'une coopération. Le contenu sémantique n'est pas connu du système qui n'est qu'un messager entre les utilisateurs.
- Production : La production est la création, la manipulation et le partage de ressources (texte, livre, plan, etc.).
- Coordination : La coordination est la définition des acteurs, des fonctions et des tâches (leurs relations temporelles), et enfin l'allocation des tâches et des fonctions aux acteurs.

(Salber et al., 1995) précise que l'acteur peut être un système contenant de l'automatisation (dit intelligent).

Les travaux de (V. M. R. Penichet et al., 2007) proposent une autre classification où la relation entre les groupes, rôles et acteurs coopérant est classifiée en terme de : contraintes spatio-temporelles, d'échange d'information, de communication et de coordination au lieu du système.

La coopération ou la collaboration entre un système contenant de l'automatisation et un utilisateur demande de définir des moyens de coordination. La coordination est définie par (Malone & Crowston, 1990) comme l'acte de gérer les interdépendances entre les tâches et les fonctions réalisées lors de la coopération pour atteindre un but. Ces interdépendances peuvent être définies comme des objets de coordination présentés par (Sire, 2000). Un objet de coordination est défini par (Sire, 2000) comme un objet qui sert à un moment donné à coordonner les tâches des individus. Ces objets de coopération traduisent sous forme matérielle les protocoles de coordination définis par (Sire, 2000) comme ce qui définit les actions à réaliser et le sens donné à ces actions pour mener à bien une coordination.

1.4 Synthèse

Dans la suite de cette thèse, nous décomposons les tâches utilisateurs et les fonctions systèmes selon les modèles en quatre étapes du traitement de l'information de (R. Parasuraman et al., 2000). Nous sélectionnons cette décomposition car elle nous permet d'adopter un langage commun avec les descriptions de tâches utilisées en IHM. En effet, le modèle en quatre étapes du traitement de l'information est similaire aux décompositions utilisées en IHM présentées dans la première sous-section. Elle décompose une tâche en tâche perceptive, cognitive (analyse et décision) et motrice. Également, elle permet de décrire les tâches et les fonctions des utilisateurs et des systèmes a des niveaux d'abstraction bas, similairement aux approches IHM et contrairement aux autres approches présentées. Dans la suite de cette thèse, nous utiliserons les termes coopération et tâches et fonctions coopératives. Nous traiterons indifféremment les tâches et fonctions permettant de réaliser une coopération ou une collaboration telles que définis par (Roschelle & Teasley, 1995) et (Grudin, 1994). Nous décrirons les tâches et les fonctions qui permettent aux rôles des acteurs de communiquer, de se coordonner ou de produire un artefact ensemble par des tâches et fonctions que nous appelons coopératives. Dans la section suivante, nous présentons comment nous adaptons ce modèle à nos besoins de modélisation. Dans la suite de cette thèse, nous utiliserons le terme « utilisateur » plutôt que « humain » ou

« opérateur » afin de désigner une personne coopérant avec un système contenant de l'automatisation pour atteindre un ou plusieurs buts.

2 Analyse et la modélisation des tâches utilisateur

La modélisation des tâches permet à travers différentes méthodes et notations de décrire, structurer et stocker le comportement des utilisateurs recueilli lors de l'analyse des tâches. Ce support fournit des éléments de représentation tels que les arbres hiérarchiques, les diagrammes de flux ou les organigrammes. Les modèles produits lors de l'analyse et de la modélisation des tâches permettent de représenter et de raisonner sur les caractéristiques particulières du comportement des utilisateurs selon le but de l'analyste (ex. mesure de l'effort physique, de la charge mentale de l'utilisateur). L'analyste sélectionne la technique de modélisation des tâches la plus adaptée à son but de modélisation.

Afin de réaliser des tâches pour atteindre un but, les utilisateurs perçoivent et manipulent des éléments de leur environnement pouvant être des objets physiques, des informations contextuelles sur la situation actuelle, ou l'état du système interactif. Ainsi, pour la conception de systèmes interactifs, identifier les buts utilisateurs, regrouper et raffiner les tâches à accomplir, décrire leur ordre d'exécution et les objets ou données perçues ou manipulées font partie de l'activité d'analyse de tâches.

2.1 Analyse de tâches

Définition de l'analyse tâches adaptée de (Kirwan & Ainsworth, 1992) :

> *On appelle analyse de tâche l'étude de ce qu'un utilisateur (ou plusieurs utilisateur) doit faire en termes d'actions et de processus cognitifs, ainsi de ce que l'utilisateur manipule en termes de données, périphériques et objets pour atteindre un but dans des contextes d'utilisation identifiés.*

L'analyse de tâches sert différents objectifs : l'organisation du travail, l'allocation des tâches et des fonctions entre le système et l'utilisateur, la représentation de l'activité mentale des utilisateurs, etc. Les travaux de (Hollnagel, 2006) et (Caffiau et al., 2010) dressent un récapitulatif de l'analyse de tâches et des différentes approches de modélisation de tâches. L'analyse de tâches a d'abord été utilisée en ergonomie (J. Annett & Duncan, 1967) en réponse à la complexité croissante des tâches des utilisateurs dans les industries dites lourdes (métallurgie, pétrochimie, …). Cette analyse de tâches identifie les tâches des utilisateurs mais ne prend pas en compte les informations, connaissances et les processus de traitement de l'information des utilisateurs (tâches cognitives). L'analyse cognitive des tâches utilisateur (*cognitive task analysis*) (Clark & Estes, 1996) a étendu l'analyse de tâches afin de décrire ces tâches cognitives et les informations manipulées par les utilisateurs. Nous utilisons le terme « analyse de tâches » pour désigner l'analyse de tâches incluant la description des tâches cognitives, des informations et connaissances manipulées par l'utilisateur.

2.2 Modèles de tâches

Définition d'un modèle de tâches adaptée de (Paterno et al., 1997) et (Limbourg & Vanderdonckt, 2003) :

*On appelle **modèle de tâches** la description hiérarchique et ordonnancée temporellement d'un but utilisateur en tâches et fonctions en indiquant leurs types.*

Définition d'un but utilisateur adaptée de (Paterno et al., 1997) :

*On appelle **but** utilisateur une modification de l'état du système ou d'objets de l'environnement que les utilisateurs veulent atteindre.*

Définition d'une tâche adaptée de (Paterno et al., 1997) :

*On appelle **tâche** ce qui est réalisé par un utilisateur pour atteindre un but dans un domaine d'application spécifié.*

Définition d'une fonction :

*On appelle **fonction** ce qui est exécuté par un système pour soutenir un but utilisateur dans un domaine d'application spécifié.*

La modélisation des tâches utilisateurs est une approche systématique et complémentaire à l'analyse de tâches. La modélisation des tâches utilisateur consiste à décrire le comportement des utilisateurs identifié lors de l'analyse de tâches. Les tâches des utilisateurs sont décrites de manière structurée afin de faciliter leur compréhension dans le cas de systèmes possédant un grand nombre de fonctionnalités et permettant de gérer des tâches complexes. Les caractéristiques des notations et outils de modélisation de tâches vont ainsi dépendre : du domaine d'application, du but de la modélisation et des tâches à modéliser.

La modélisation de tâches privilégie une structure hiérarchique (but, sous-buts, tâches élémentaires) sous forme arborescente comme introduit par HTA (John Annett, 2004) dont un exemple est présenté dans la Figure 33. Les modèles de tâches permettent de faciliter la compréhension des utilisateurs et des tâches qu'ils effectuent. Dans le domaine des systèmes critiques, l'analyse des tâches doit permettre de décrire les procédures pour les situations normales, les procédures de maintenance du système et les procédures pour des situations anormales (Redmill & Rajan, 1996). Dans le processus de conception, les modèles de tâches permettent l'identification et l'analyse de nouveaux besoins (P. Johnson et al., 1991). Les modèles de tâches permettent également d'identifier les connaissances nécessaires à l'utilisateur pour utiliser un système et donc pour coopérer avec un système contenant de l'automatisation pour atteindre un but dans le cadre de l'analyse de l'automatisation.

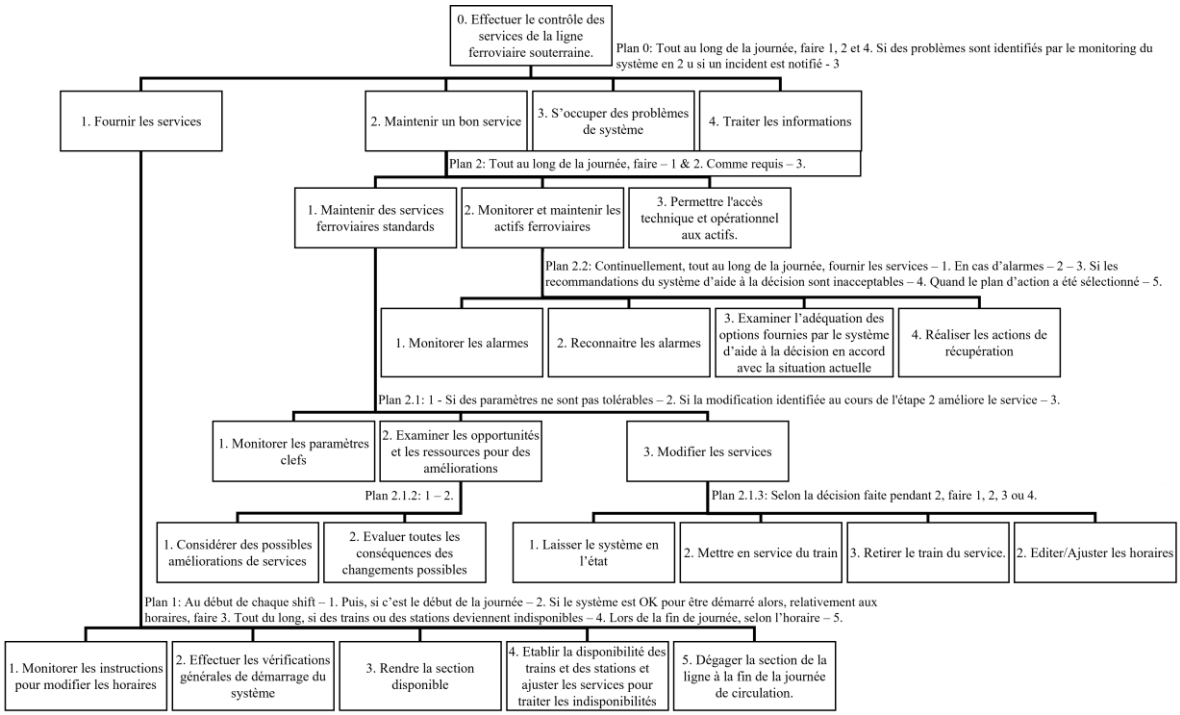


Figure 33. Modèle de tâches HTA pour effectuer le contrôle de service d'une ligne de métro tiré de (Shepherd, 2002)

Différentes notations de modélisation de tâches ont été proposées avec, notamment, l'introduction d'opérateurs temporels (ISO 8807, 1989) permettant de décrire l'ordonnancement des tâches (ex. parallélisme, répétition ou choix).

La sous-section suivante décrit les notations et outils de modélisation de tâches existantes. Ensuite, la notation et l'outil de modélisation de tâches HAMSTERS étendu et utilisé dans cette thèse est décrit en détail.

2.3 Notations et outils de modélisation de tâches

Pour modéliser les tâches des utilisateurs, la plupart des notations proposent de modéliser les tâches pour atteindre les buts des acteurs utilisateurs selon leurs rôles.

2.3.1 Acteurs et rôles

Définition d'un acteur adaptée de (van der Veer et al., 1996):

> *Les acteurs font le plus souvent référence à des entités individuelles. Les acteurs peuvent être des entités non-humaines ou des systèmes avec lesquels les entités humaines peuvent collaborer.*

Bien que la définition d'acteur de (van der Veer et al., 1996) identifie les systèmes comme pouvant être des acteurs, la plupart des notations ne prennent en compte que des acteurs utilisateurs. Le système n'est pas un acteur mais est considéré comme un outil permettant à l'utilisateur d'atteindre son but. Selon (van der Veer et al., 1996), pour la modélisation de tâche, il est important d'identifier les différents types d'acteurs. Ces différents types d'acteurs sont identifiés par des caractéristiques psychologiques (habilités spatiales et cognitives) leurs

caractéristiques liées aux tâches (expertise ou connaissances). Dans la suite de cette thèse, nous parlerons des habilités des acteurs pour désigner les caractéristiques des acteurs.

Définition de rôle adaptée de (van der Veer et al., 1996) :

> *On appelle rôle une classe d'acteurs auxquels certains sous-ensembles de tâches sont alloués. Par définition, les rôles sont génériques. Plus d'un acteur peut réaliser un même rôle. Un seul acteur peut réaliser plusieurs rôles.*

(van der Veer et al., 1996) précise que :

- Les rôles peuvent être réalisés temporairement, être négociés entre acteurs, acceptés ou refusés.
- Les acteurs peuvent avoir une représentation interne (mentale) de leur propre rôle et des autres rôles.
- Les rôles peuvent être représenté extérieurement par un comportement instrumental ou symbolique et par des objets (blouse blanche, stéthoscope ou perruque).

2.3.2 Présentation des notations pour la modélisation de tâches[4]

Les travaux de (Limbourg & Vanderdonckt, 2003) et (Caffiau et al., 2010) présentent une analyse systématique des outils graphiques, documentés et gratuits soutenant l'édition de modèles de tâches.
Ci-après, nous présentons un échantillon représentatif en termes de besoins couverts pour l'analyse de tâches des notations existantes pour la modélisation de tâches. Nous incluons dans cet échantillon des méthodes descriptives (ce qui est fait), normatives (ce qui doit être fait), et formatrice (ce qui pourrait être fait), car nous sommes davantage intéressés par les concepts inclus dans ces méthodes et notations.

HTA (*Hierarchical Task Analysis*) (J. Annett & Duncan, 1967) a d'abord été une notation textuelle. Puis Shepherd a défini une version graphique qui permet une décomposition hiérarchique visuelle (Shepherd, 1985) des buts en sous-tâches pour le *training*. Également, HTA définit le séquencement des tâches par leur numérotation ainsi qu'un plan pour chaque niveau de décomposition.
MAD (Méthode Analytique de Description des tâches) (Scapin & Pierret-Golbreich, 1989) se base sur une méthode d'interview structurée afin de décrire hiérarchiquement les tâches des utilisateurs ainsi que les relations temporelles. Cafiau et al. (Caffiau et al., 2010) proposent une extension de cette notation (K-MAD) qui définit des évènements, des préconditions aux tâches ainsi que des objets utiles à l'utilisateur ou influençant l'atteinte du but.
UAN (*User Action Notation*) (Hix & Hartson, 1993) est une notation permettant de décrire les actions interactives de l'utilisateur avec l'interface système. Le résultat se présente sous la forme d'un tableau décrivant les actions utilisateurs, le retour sur l'interface, l'état de l'interface, et ce qui est effectivement réalisé en résultat.
Diane+ (Tarby & Barthet, 1996) est une méthodologie comprenant une activité de représentation du résultat de l'analyse de tâches sous la forme d'un modèle de tâches hiérarchique. Les relations temporelles sont également spécifiées.
CTT (*Concur Task Tree*) (Paterno et al., 1997) est une notation graphique permettant de modéliser sous forme arborescente les relations hiérarchiques entre tâches et sous-tâches et de définir leur ordonnancement temporel à l'aide des opérateurs LOTOS (ISO 8807, 1989). Les opérateurs

[4] Modifications apportées post-publication.

d'ordonnancement temporels s'appliquent entre deux tâches seulement. Cette notation fournit un support à la description de tâches coopératives et de rôle utilisateur. Il est possible de décrire la cardinalité (point à point, broadcast) de la coopération et le nombre d'acteurs impliqué dans la coopération.

VTML (*Visual Task Modeling Language*) (Brown & Leveson, 1998) est une notation graphique conçue pour l'analyse formelle de la sûreté de fonctionnement à l'aide d'un modèle de tâches. Les évènement et tâches permettant les transitions d'états du système sont représentés. Les tâches de communication sont mises en valeur.

TAKD (*Task Analysis for Knowledge Descriptions*) (Diaper, 2001) est une méthode créée pour identifier les exigences pour le *training* des utilisateurs. Elle permet d'identifier les connaissances requises pour la réalisation des tâches à travers des interviews ou d'observations. Ces connaissances sont ensuite analysées pour identifier les actions et objets requis. Les descriptions fournies se veulent haut-niveau avec des descriptions génériques d'objets et d'actions pour informer la conception de nouveaux systèmes.

AMBOSS (Giese et al., 2008) est un environnement dédié à la modélisation des tâches utilisateur de systèmes critiques. Malgré le manque de description de la notation associée (Caffiau et al., 2010), AMBOSS permet de décrire les tâches des rôles des utilisateurs de manière hiérarchique et ordonnée temporellement. Également, AMBOSS prend en compte le concept de « barrière » exprimant une condition pour une exécution valide d'une tâche.

ANSI/CEA-2018 (Rich, 2009) est un standard décrivant un langage basé XML pour la description des modèles de tâches. Le but de ce standard est une utilisation à l'exécution des modèles de tâches pour guider les utilisateurs.

COMM (*COllaborative and MultiModal*) (Jourde et al., 2010) est une notation permettant de décrire de manière hiérarchique et sous forme d'arbre les tâches avec des systèmes multimodaux et multi-utilisateurs. Cette notation étend la notation CTT (Paterno et al., 1997) en définissant les rôles interactifs (rôle assigné de manière dynamique jusqu'à la fin d'une tâche) et les tâches multimodales. Également, la notation utilise les relations de Allen décrivant les interdépendances temporelles entre deux tâches. Avec CTT, la coopération de deux acteurs est représentée par deux modèles de tâche individuels et un modèle de tâche coopératif. COMM propose de décrire la coopération dans un modèle de tâche unique décrivant les tâches de plusieurs rôles. Un outil d'édition de modèle de tâches appelé e-COMM est disponible.

La recommandation du W3C (Paternò et al., 2014) a pour but de définir un standard basé sur la notation CTT (Paterno et al., 1997). Elle définit un méta-modèle d'une tâche, des opérateurs temporels et une description XML des modèles de tâches.

SAMANTA (*Situation Awareness Modeling and ANalysis for Transition Amelioration*) (Villaren et al., 2012) est une méthodologie faisant usage de modèles de tâches (CTT) et de modèles décrivant les éléments de la propriété *situation awareness* (voir Chapitre 2.2.2) afin de permettre l'analyse des transitions de contrôle et d'autres situations pouvant conduire les opérateurs à une « rupture » de la *situation awareness*.

CUA (*Collaboration Usability Analysis*) (Pinelle et al., 2003) est une technique de modélisation se concentrant sur le travail en équipe pour le développement et l'évaluation de collecticiels. Les tâches décrites sont des tâches individuelles réalisées par un acteur ou coopératives réalisées par plusieurs acteurs lors de scénarios. Les actions décrivent les différentes manières de réaliser une tâche collaborative.

CTML (*Collaborative Task Modeling Language*) (Wurdel et al., 2008) est un cadre conceptuel pour l'analyse et la description des tâches collaboratives des rôles des acteurs. CTML inclut la modélisation du domaine, du comportement des utilisateurs par la description des tâches de chaque rôle, des états des tâches réalisées (actives, en cours, inactive, annulée, suspendue, passée ou complétée). CTML utilise les modèles CTT (Paterno et al., 1997) pour modéliser les tâches des utilisateurs durant un scénario.

MABTA (*Multiple Aspect Task Analysis*) (Lim, 2004) est un mécanisme permettant d'intégrer divers aspects de la situation de travail d'un groupe d'utilisateurs collaborant entre eux dans

l'analyse de tâches. Ces aspects peuvent concerner les relations entre les utilisateurs, l'environnement, ou les artefacts. Les modèles de tâches représentent les tâches coopératives de plusieurs utilisateurs ainsi que les tâches individuelles des utilisateurs. MABTA inclut également des modèles d'aspect représentant des facteurs contextuels qui affectent les tâches des utilisateurs.
FlowiXML (Guerrero Garcia et al., 2008) est une méthode proposant de décrire les tâches, les processus, le flux de travail, le domaine et le contexte d'utilisation pour des interfaces utilisateurs pour les systèmes de gestion de flux de travail. FlowiXML utilise la notation CTT pour décrire les tâches utilisateurs et les fonctions systèmes. Chacun des changements apportés dans un modèle est propagé dans les autres.
CoTaL et CoTaSE (*Cooperative Task Specification Environment*) (Buchholz & Forbrig, 2017) est une extension aux modèles de tâches pour la description de la coopération. CoTaSE propose de décrire les tâches utilisateur mais aussi les tâches coopératives dans un modèle d'équipe (le team model). Ce modèle décrit de manière plus niveau la coopération et les contraintes pour réaliser les tâches coopératives.
TOUCHE (*Task-Oriented and User-Centred Process Model for Developing Interfaces for Human–Computer–Human Environments*) (Victor M. R. Penichet, Lozano, Gallud, et al., 2009) est un processus et une méthodologie pour le développement des interfaces utilisateurs des collecticiels. La description des tâches se base sur les modèles CTT (Paterno et al., 1997). TOUCHE propose de décrire des diagrammes de collaboration décrivant la collaboration des utilisateurs. Une tâche collaborative ou coopérative peut être décrite selon son type : coordination, communication, coopération. Également, la notation permet de spécifier si une tâche coopérative est synchrone ou asynchrone, locale ou distante.

CWA (*Cognitive Work Analysis*) (Vicente, 1999) est un cadre conceptuel guidant l'analyse et la conception de systèmes interactifs pour le travail. Cinq phases sont définies dans ce cadre conceptuel. La première propose d'analyser le domaine de travail (*Work Domain Analysis*) en identifiant les contraintes et les fonctions du système analysé. Le résultat de cette phase permet de décrire le but de l'utilisation du système jusqu'au capacité de ce système. La seconde phase permet l'analyse des tâches de contrôle (*Control Task Analysis*) qui décrit les situations dans lesquelles les fonctions systèmes peuvent être ou sont habituellement exécutées. Puis, le processus de prise de décision des utilisateurs est étudié afin d'identifier les informations nécessaires à ce processus. La troisième phase (*Strategies Analysis*) permet d'identifier les contraintes qui définissent comment un système peut passer d'un état à un autre. La quatrième phase définie l'allocation de fonctions et de tâches entre les acteurs en se basant sur les résultats des phases précédentes (*Social Organisation and Cooperation Analysis*). Enfin, la dernière phase se base sur le modèle SRK (Rasmussen, 1983) (voir section 1.1.1) pour analyser les compétences des utilisateurs requises pour le travail ainsi conçu (*Workers Competencies Analysis*).
TKS (Task Knowledge Structure) (P. Johnson & Johnson, 1989) est une notation permettant d'identifier les connaissances des utilisateurs requises afin de réaliser leurs tâches. Cette notation permet d'identifier les actions et les objets réalisées ou manipulés par les différents rôles.
Également, **GTA (Groupware Task Analysis)** (van der Veer et al., 1996) est une méthode et une notation conçue pour modéliser les tâches réalisées dans un environnement complexe et coopératif. Les notions de rôles des acteurs, l'organisation, les protocoles, ou stratégies peuvent être définit. Cette méthode se veut à la fois descriptive et normative, un premier modèle de tâches décrit la situation actuelle, un second comment les tâches peuvent être réalisées dans une activité de re-conception.
GOMS (*Goals, Operators, Methods and Selection rules*) (S. K. Card et al., 1983) est une méthode et une notation textuelle décrivant un ensemble de buts, d'opérateurs, de méthodes pour accomplir les buts et de règles (i.e. « si … alors [method1] sinon [method2] ») pour choisir parmi ces méthodes. En se basant sur le modèle du processus humain des mêmes auteurs, la description de tâches permet une estimation de la performance de l'utilisateur. Dans une étude comparative de différents outils pouvant supporter GOMS dont QGOMS (*Quick and dirty* GOMS), Baumeister et

al. (Baumeister et al., 2000) concluent qu'aucun des outils testés ne permet un support complet de GOMS. Plusieurs méthodes et notations se basant sur GOMS ont été proposées se basant sur différent modèles d'interaction ou en ajoutant de nouveaux opérateurs tel que le parallélisme des tâches. GOMS est une évolution de *Keystroke-Level Model* (**KLM**) (S. K. Card et al., 1980) permettant également d'estimer la performance de l'utilisateur en décrivant les tâches : presser, pointer, déplacer la main d'un appareil à l'autre ou ajuster la position de la main sur l'appareil, et la préparation mentale à l'action. GOMS n'inclut pas la représentation des contraintes temporelles entre les tâches.

HAMSTERS (*Human-centered Assessment and Modeling to Support Task Engineering for Resilient Systems*) (Martinie et al., 2011) et une notation permettant de décrire de manière hiérarchique les tâches et sous-tâches et de les ordonner temporellement. L'outils associé permet de simuler les modèles de tâches et, ainsi, de produire des scénarios d'utilisation. La notation et l'outil associé sont décrits en détail dans la prochaine section.

2.4 Comparaison

Le Tableau 4 présente une synthèse des caractéristiques de l'échantillon représentatif de notations de modélisation de tâches utilisateur présenté ci-avant. Cette synthèse permet de montrer les critères généralement remplis par les notations existantes.

Tableau 4. Comparaison d'un échantillon représentatif de notations de modélisation de tâches utilisateurs étendant le travail de (Martinie De Almeida, 2011).

		HTA	GOMS	TKS	MAD	UAN	ANSI/CEA-2018	W3C	GTA	DIANE+	VTML	CWA	TAKD	CTT	AMBOSS	COMM	SAMANTA	CUA	CTML	CoTaL	MABTA	FlowiXML	TOUCHE	HAMSTERS
Data	Description des connaissance			x								x					x							x
	Description des périphérique manipulés																				x			x
	Description des informations							x					x	x			x		x	x		x	x	x
	Toutes les informations ci-dessus doivent être intégrées dans un seul modèle																							x
Raffinement de tâche	Tâche motrice																				x			x
	Tâche cognitive															x					x			x
	Tâche interactive							x						x		x	x		x	x		x	x	x
	Tâche perceptive																							x
Description d'une tâche	Criticité de la tâche																							x
	Temps max d'exécution de la tâche																							x
	Temps min d'exécution de la tâche																							x
Collaboration	Description des tâches coopératives							x						x		x	x	x	x	x	x	x	x	x
	Description des rôles							x						x		x	x	x	x	x	x	x	x	x

Notation	Textuelle	x	x	x		x	x				x		x					x						
	Graphique	x		x	x			x	x	x	x		x	x	x	x	x	x	x	x	x	x	x	x
	Décomposition Hiérarchique	x	x	x	x	x	x	x	x	x			x	x	x	x	x	x	x	x	x	x	x	x
	Relations temporelles entre les tâches		x	x	x	x		x	x	x	x			x	x	x	x	x	x	x	x	x	x	x
Structuration	Description des subroutines																							x
	Description des composant																							x
Automatisation	Autorité																							
	Responsabilité																							
	Transition de contrôle																							
	Partage de ressource																							
Outil	Outillé	Task-Architect	CAT-HCI	ADEPT	K-MADe	Quantum	ANSI/CEA-2018		EUTERPE	Isolde	SpecTRM		LUTAKD	CTTe	AMBOSS	eCOMM	SAMANTA		CTML Editor	CoTaSE		FlowiXML	TOUCHE CASE Tool	HAMSTERS

Le Tableau 4 nous permet de montrer que la plupart des notations permettent la description de la décomposition hiérarchique des tâches, leur ordonnancement temporel des tâches, et ceci de manière graphique. Cependant elle met aussi en évidence les points suivants :

- La possibilité de structurer les modèles, en utilisant des composants n'est seulement fournie par HAMSTERS.
- Seule la notation HAMSTERS permets de décrire de manière graphique des tâches utilisateurs raffinées (Motrice, perceptive), et pour les tâches cognitives, COMM, MABTA (décisions) et HAMSTERS le permettent.
- Aucune notation ne permet de représenter explicitement les éléments caractérisant l'automatisation.
- En revanche, toutes les notations de modélisation de tâches sont supportées par un outil, sauf CWA (des outils sont décrits mais ne semblent pas disponibles), CUA et MABTA.

3 Notation HAMSTERS et outil HAMSTERS

Cette section présente la notation et l'outil HAMSTERS (***H**uman-centered **A**ssessment and **M**odeling to **S**upport **T**ask **E**ngineering for **R**esilient **S**ystems*) avant les extensions apportées dans le cadre de cette thèse. Nous décrivons en détail la notation et l'outil HAMSTERS

La première version de la notation et de l'outil HAMSTERS est le résultat d'un chef d'œuvre de Master 2 IHM en 2009 et visait à créer un éditeur de modèles de tâches centré utilisateur de modèles. C. Martinie et E. Barboni ont poursuivi le développement de la notation et l'outil HAMSTERS. Depuis, l'outil et la notation HAMSTERS sont régulièrement étendus pour couvrir l'avancée des travaux de l'équipe ICS.

L'outil HAMSTERS permet l'édition, la simulation de modèles de tâches et la production de scénarios d'utilisation associés à ces modèles.

HAMSTERS reprend un certain nombre d'éléments de notations de modélisation de tâches existantes :

- La décomposition hiérarchique des buts en sous-buts,
- Le raffinement du type de tâches,
- L'utilisateur d'opérateurs temporels LOTOS (ISO, 1989) utilisés par la notation CTT (Paterno et al., 1997),
- Une représentation graphique des modèles de tâches,
- Un outil logiciel associé permettant d'éditer, de classer et de simuler des modèles de tâches.

3.1 Types de tâche

Les types de tâches sont un premier élément permettant la décomposition hiérarchique des buts et le raffinement des tâches.

3.1.1 Types de tâches génériques

Le Tableau 5 présente les types de tâches génériques de la notation HAMSTERS :

- Un **but** permet de décrire un but utilisateur décomposable en plusieurs sous-but ou tâches.
- Une **tâche abstraite** permet de décrire un sous-but ou une tâche décomposable en plusieurs des sous-tâches de types différents.
- Une **tâche abstraite utilisateur** permet de décrire un sous-but ou une tâche décomposable en plusieurs des sous-tâches de types utilisateurs.
- Une **fonction abstraite système** permet de décrire un sous-but ou une fonction décomposable en sous-fonctions de types systèmes.
- Une **tâche abstraite interactive** permet de décrire un sous-but ou une tâche décomposable en sous-tâches ou action de types interactives.
- Une **tâche utilisateur** permet de décrire une tâche décomposable en actions utilisateur.
- Une **fonction système** permet de représenter une fonction exécutée par le système.

Tableau 5. Types de tâches et fonctions génériques de la notation HAMSTERS

Types de tâche	Représentation Hamsters
But	
Tâche abstraite	
Tâche abstraite utilisateur	
Tâche abstraite système	
Tâche abstraite interactive	
Tâche utilisateur	
Fonction système	

Ces tâches sont dites génériques car ces tâches peuvent être raffinées en tâches perceptives, cognitives, motrices, interactives ou systèmes.

3.1.2 Tâches perceptives, cognitives, motrices et interactives

Dans cette sous-section, nous présentons le raffinement des tâches utilisateurs de la notation HAMSTERS.

3.1.2.1 Tâches perceptives, cognitives et motrices

Le raffinement des tâches utilisateur permet de décrire les différentes étapes d'une action. La théorie de l'action présentée en section 1.1.3 de ce chapitre décrit trois types de tâches :

- Perceptive (Perception du changement d'état)
- Cognitive (formalisation d'un but, établissement d'une intention, spécification d'une suite d'action, interprétation et évaluation de la perception)
- Motrice (exécution des actions)

Similairement, les travaux de (S. K. Card et al., 1983) présentés en section 1.1.2 de ce chapitre proposent un modèle du processeur humain qui représente les différents types de tâches selon les unités de traitement suivantes : processeur moteur, processeur perceptif et processeur cognitif.

La notation HAMSTERS permet la description de ces trois types de tâches utilisateur à l'aide des tâches représentées dans le Tableau 6.

Tableau 6. Types de tâches utilisateur

Unités de traitement	Types de Tâches HAMSTERS et leur représentation dans l'outil
Perceptif	
Cognitif	
Moteur	

Une **tâche perceptive** permet de décrire une action sensorielle (toucher, voir, sentir, gouter, entendre, …) de récupération d'information extérieures. Par exemple : regarder le plateau de jeu.

Une **tâche cognitive** permet de décrire un traitement de l'information. Par exemple : choisir une stratégie de jeu.

Une **tâche motrice** permet de décrire un mouvement physique de la part de l'utilisateur. Par exemple : prendre un jeton.

3.1.2.2 Tâches cognitives d'analyse et de décision et tâches interactives

La notation HAMSTERS permet de raffiner les tâches cognitives en tâches d'analyse et de décision. Ce raffinement se base sur le modèle simple en 4 étapes du traitement de l'information par (R. Parasuraman et al., 2000) présenté dans la Figure 31. Ce modèle a été adapté à nos objectifs de modélisation. La notation HAMSTERS regroupe dans la première étape le traitement sensoriel et la perception consciente de l'information.

Note :
Ce modèle nous permet, par exemple, de capturer la loi de Hick-Hyman (Hick, 1952) qui stipule que le temps nécessaire pour prendre une décision augmente logarithmiquement avec le nombre de choix présentés à l'utilisateur. Ainsi, il est possible de décrire avec l'adaptation du modèle : les tâches de perception des différents choix présentés, les tâches d'analyse de ces différents choix et la tâche de décision. Alors que selon le modèle de (R. Parasuraman et al., 2000), une seule étape regrouperait la perception et l'analyse. En effet, l'étape « Perception/Mémoire de travail » regroupe les opérations cognitives avant le point de décision (perception consciente, manipulation des informations traitées et récupérées dans la mémoire de travail). La perception (consciente) des différents choix et leur analyse serait représentées dans une seule tâche. Notre adaptation du modèle permet de décrire la perception des différents éléments présentés d'une part et l'analyse de ces éléments perçus d'autre part.

La Figure 34 présente les différentes étapes du modèle simple en 4 étapes du traitement de l'information utilisé dans la notation HAMSTERS et les types de tâches associés dans la notation HAMSTERS :

- L'étape de traitement sensoriel est décrite dans HAMSTERS par les tâches perceptives.
- L'étape de perception/mémoire de travail est décrite dans HAMSTERS par les tâches cognitives d'analyse.
- L'étape de prise de décision est décrite dans HAMSTERS par les tâches cognitives de décision.
- L'étape de la sélection de la réponse est décrite dans HAMSTERS par les tâches motrices.

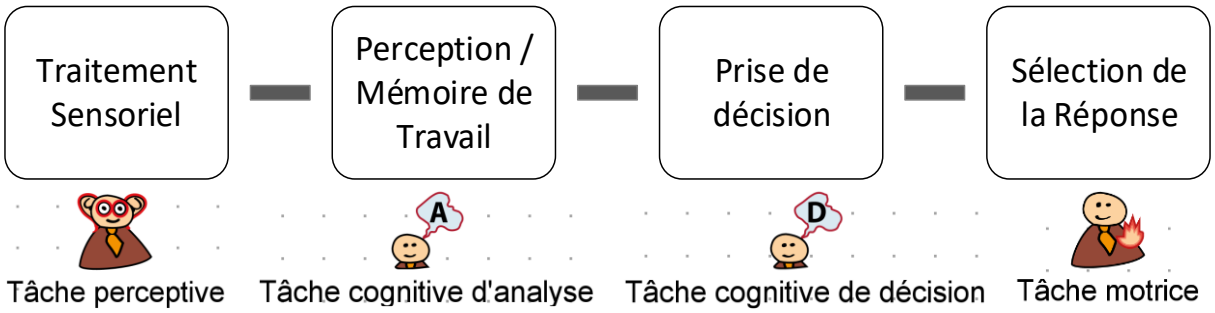


Figure 34. Etapes du modèle de (R. Parasuraman et al., 2000) et leur représentation dans la notation HAMSTERS

Une **tâche d'analyse cognitive** permet de décrire que l'utilisateur analyse une information. Par exemple : analyser les jetons disponibles pour le tour de jeu.

Une **tâche de décision cognitive** permet de décrire que l'utilisateur prend une décision en rapport avec une situation. Par exemple : décider de jouer le jeton numéro 5 pour ce tour de jeu.

3.1.3 Fonctions systèmes

La notation HAMSTERS permet de raffiner les fonctions systèmes selon une adaptation du modèle proposé par (R. Parasuraman et al., 2000). La notation HAMSTERS propose de décrire les fonctions comme une migration des tâches utilisateurs de l'étape correspondante et non plus en tant que fonctions soutenant la réalisation des tâches de l'étape correspondante.

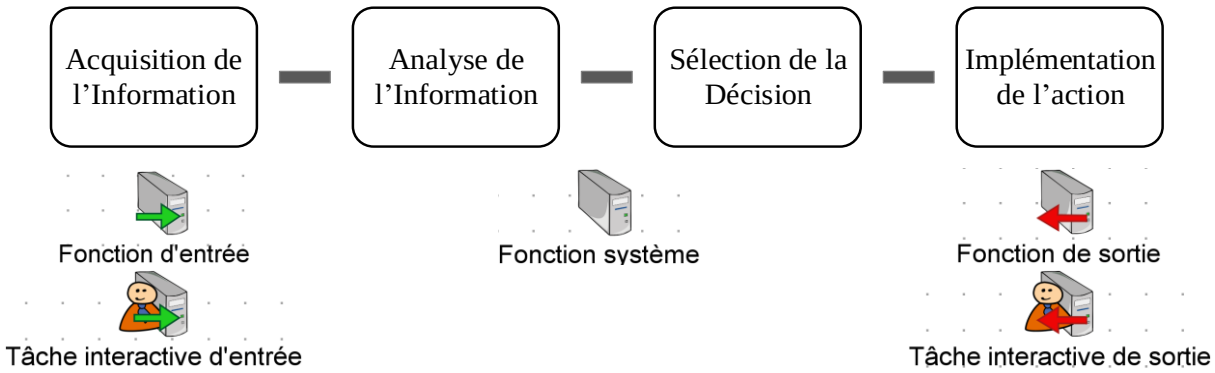


Figure 35. Modèle simple en quatre étapes du traitement système de l'information de (R. Parasuraman et al., 2000) et leur représentation dans la notation HAMSTERS

La Figure 35 présente les différentes étapes du modèle simple en 4 étapes du traitement système de l'information et leur représentation dans la notation HAMSTERS :

- L'étape d'acquisition de l'information est représentée par les fonctions d'entrée ou les tâches interactives d'entrée. Ces fonctions traitent les évènements d'une interaction utilisateur ou d'une modification de contexte. Ce sont aussi les fonctions d'enregistrement de l'environnement (enregistrement du son, de l'image, des mouvements, de la chaleur, ...).

Note :
Contrairement à la définition de (R. Parasuraman et al., 2000) nous ne prenons en compte dans cette étape que les fonctions d'entrée et non pas les fonctions de filtrage et de mise en valeur sur les données acquises.

- Les étapes d'analyse de l'information et de sélection de décision sont représentées par les fonctions système manipulant des données ou sélectionnant des données. Par exemple, des fonctions de filtrage, de tri ou de recommandation.
- L'étape d'implémentation de l'action est représentée par les fonctions de sortie ou les tâches interactive de sortie. Ce sont des affichages, des sorties son, des mouvements, etc.

Une **fonction d'entrée** permet de décrire que le système reçoit un flux entrant de données. Par exemple, le système reçoit une onde radio.

Une **fonction système** permet de décrire une fonction de traitement de données exécutée par le système. Par exemple, le système calcule la distance de l'avion.

Une **fonction de sortie** permet de décrire que le système fournit un flux sortant de donnée. Par exemple, le système émet une onde radio.

Une **tâche interactive d'entrée** permet de décrire que le système reçoit un flux entrant de données fournit par l'utilisateur. Par exemple : l'utilisateur sélectionne un jeton à jouer au système en cliquant sur le bouton gauche de la souris sur le jeton.

Une **tâche interactive de sortie** permet de décrire que le système fournit un flux sortant de donnée à l'utilisateur. Par exemple : le système affiche le tour de jeu à l'utilisateur sur un écran.

La **tâche interactive d'entrée/sortie** présentée dans le Tableau 7 permet de décrire que l'utilisateur envoie et reçoit des flux de données avec le système sans détailler les tâches perceptive, cognitive, motrice et systèmes réalisées. Par exemple : l'utilisateur fait défiler une page web à l'aide de la molette de la souris, et, en même temps, l'utilisateur voit la page web défiler.

Tableau 7. Tâches interactives d'entrée/sortie de la notation Hamsters

Types de tâche	Représentation Hamsters
Tâche interactive d'entrée/sortie	

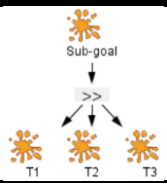
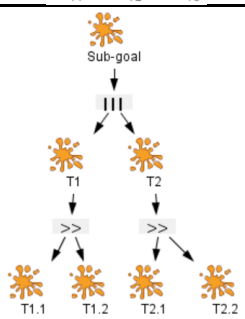
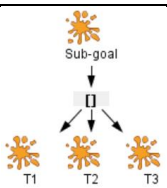
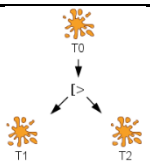
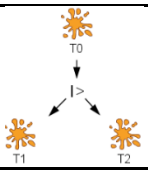
3.2 Hiérarchie et Ordonnancement temporel des tâches

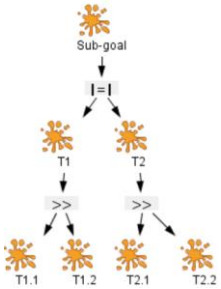
Dans cette section, nous présentons la décomposition hiérarchique des tâches de la notation HAMSTERS ainsi que les opérateurs temporels utilisés pour décrire l'ordonnancement des tâches.

3.2.1 Opérateurs d'ordonnancement temporel

Les relations temporelles qualitatives entre les tâches sont décrites grâce aux opérateurs LOTOS (ISO 8807, 1989).

Tableau 8. Opérateurs d'ordonnancement temporel utilisés par la notation HAMSTERS.

Type d'opérateur	Symbole	Illustration de tâches ordonnancées temporellement dans Hamsters
Enable	>>	
Concurrent	|||	
Choice	[]	
Disable	[>	
Suspend-resume	|>	

Order Independent	|=|	

Le Tableau 8 présente les opérateurs de la notation HAMSTERS. Seuls les opérateurs ***Disable*** et ***Suspend-resume*** fournissent une relation temporelle qualitative entre deux tâches. Les autres opérateurs peuvent être utilisés pour ordonnancer temporellement plus de deux tâches.

L'opérateur ***Enable*** permet de décrire que les tâches T1, T2 et T3 sont réalisées en séquence l'une après l'autre par l'utilisateur.

L'opérateur ***Concurrent*** permet de décrire que les tâches T1 et T2 sont réalisées simultanément de manière entrelacée par l'utilisateur. C'est-à-dire que les séquences suivantes de tâches sont possibles : (T1.1, T1.2, T2.1, T2.2), (T1.1, T2.1, T1.2, T2.2), (T1.1, T2.1, T2.2, T1.2), (T2.1, T2.2, T1.1, T1.2), (T2.1, T1.1, T1.2, T2.2), (T2.1, T1.1, T2.2, T1.2).

L'opérateur ***Choice*** permet de décrire que l'utilisateur réalise la tâche T1 ou la tâche T2.

L'opérateur ***Disable*** permet de décrire que la réalisation de la tâche T2 entraîne l'arrêt définitif de la tâche T2.

L'opérateur ***Suspend-resume*** permet de décrire que la réalisation de la tâche T2 entraîne l'arrêt temporaire de la tâche T1. Tant que toutes les sous-tâches de la tâche T1 ne sont pas réalisées, la tâche T1 peut être redémarrée à tout moment puis interrompue à nouveau par la tâche T2.

L'opérateur ***Order independant*** permet de décrire que l'utilisateur peut choisir de réaliser la tâche T1 ou la tâche T2 en premier. Cet opérateur indique que toutes les sous-tâches de la tâche choisie doivent être réalisée avant de passer à la tâche suivante. C'est-à-dire que les séquences de tâches suivantes sont possibles : (T1.1, T1.2, T2.2, T2.2), (T2.1, T2.2, T1.1, T1.2).

La notation HAMSTERS permet d'utiliser la composition d'opérateurs. Par exemple, dans le modèle de la Figure 36, en séquence de la tâche T1, l'utilisateur a le choix de réaliser la tâche T2 ou T3.

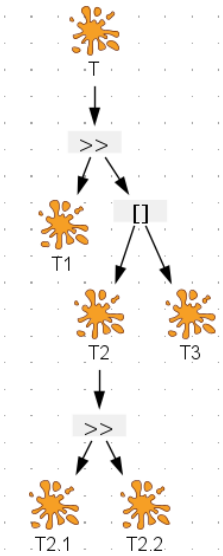


Figure 36. Exemple de composition d'opérateurs temporels.

3.2.2 Tâches itératives et optionnelles

La notation HAMSTERS permet de décrire qu'une tâche peut être optionnelle ou itérative ou optionnelle et itérative, sachant que :

- Une tâche optionnelle est une tâche dont la réalisation dépend du contexte ou d'une décision arbitraire de l'utilisateur.
- Une tâche itérative peut être réalisée une ou plusieurs fois jusqu'à ce qu'elle soit interrompue ou suspendue par une autre tâche.

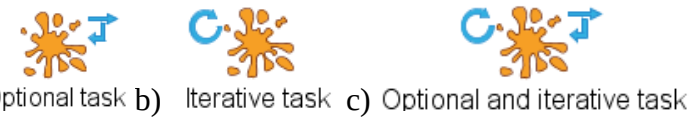


Figure 37. Représentations de tâches abstraites a) optionnelle b) itérative c) optionnelle et itérative

Tous les types de tâches peuvent être optionnelles et/ou itératives.

3.2.3 Relations temporelles quantitatives

La notation HAMSTERS permet de décrire les attributs temporels suivants pour chacun des types de tâches :

- Durée minimum d'exécution : une tâche peut s'effectuer en une durée minimale.
- Durée maximum d'exécution : une tâche peut s'effectuer en une durée maximale.

- Nombre d'itérations : une tâche itérative peut avoir un nombre maximal d'itération.

3.2.4 Copie, Sub-routines et Composants

La notation HAMSTERS fournit des éléments pour la modularité et la réutilisabilité des modèles de tâches.

3.2.4.1 Copy task

Inspiré d'un mécanisme de programmation du langage COBOL, la *copy task* permet de décrire une tâche identique à une autre et réalisée plusieurs fois par les utilisateurs. Une *copy task* peut être réalisée plusieurs fois dans un même modèle ou dans plusieurs modèles.

Figure 38. Représentation graphique de la *copy task*.

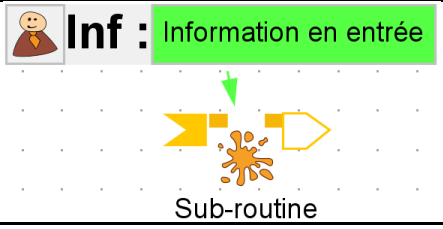
3.2.4.2 Sub-routines

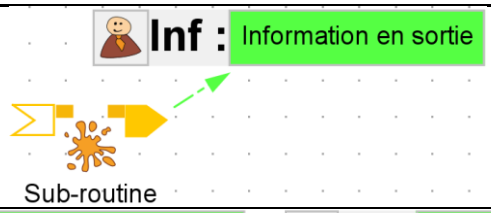
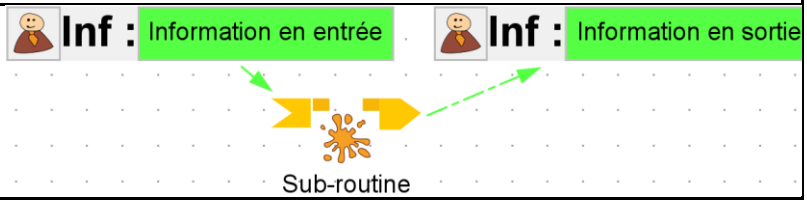
La *sub-routine* permet de définir un ensemble de tâches qu'un utilisateur réalise plusieurs fois. Cette *sub-routine* est, elle aussi, décrite de manière hiérarchique et ordonnée temporellement. La description des différents flux de données en entrée et en sortie d'une *sub-routine* permet de décrire les différents contextes dans lesquelles elle peut être réalisée. Une *sub-routine* est caractérisée par :

- Son nom.
- L'icône des tâches abstraites : une *sub-routine* peut être raffinée en tâches de différents types.
- Des ports d'entrée et sortie permettant de décrire les de flux de données entrants et/ou sortant de la *sub-routine*.

Le Tableau 9 décrit les différents types de *sub-routines* selon leurs flux de données en entrée et en sortie.

Tableau 9. Types de *sub-routines* dans la notation HAMSTERS.

Flux de données entrée ou sortie de la *sub-routine*	**Représentation avec la notation HAMSTERS**
Pas de flux de données entrant ni sortant	
Flux de données entrant seulement	

Flux de données sortant seulement	
Flux de données entrant et sortant	

3.2.4.3 Composants

Le composant permet de définir un ensemble de tâches réalisées plusieurs fois d'une manière plus générale que la *sub-routine*. La représentation d'un composant dans HAMSTERS est présentée dans la Figure 39.

Figure 39. Représentation d'un composant dans HAMSTERS

En plus de permettre la description des flux de données entrant et sortant d'un ensemble de tâches décrites dans le composant, le composant permet de décrire des paramètres génériques à ce composant. Lorsque le composant est instancié, les paramètres peuvent prendre des valeurs propres à cette instance.

La Figure 40 présente un composant décrivant le choix d'un jeton dont la valeur du paramètre <jeton> n'a pas été instanciée.

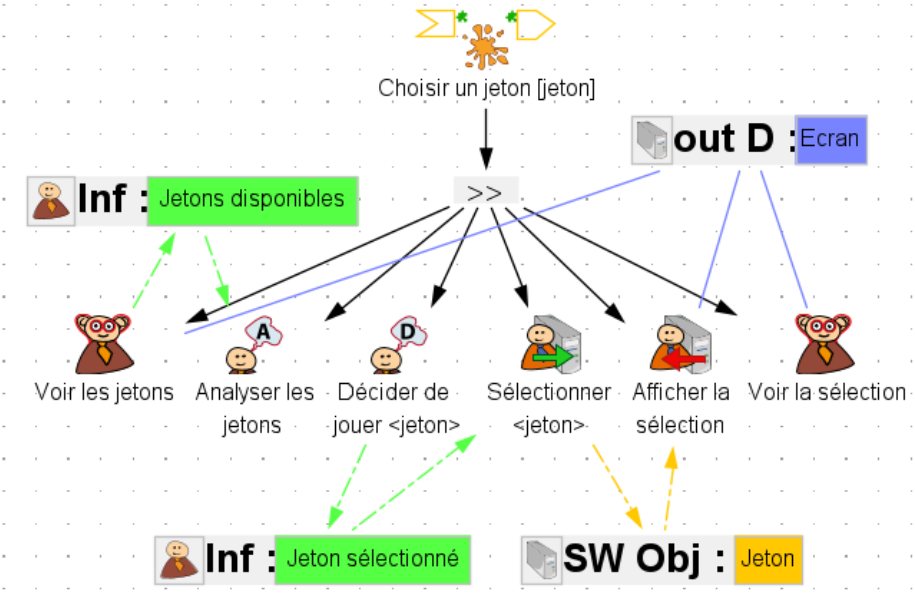


Figure 40. Composant avec le paramètre générique "jeton"

La Figure 41 présente une instance de ce composant dont le paramètre <jeton> prend la valeur 3.

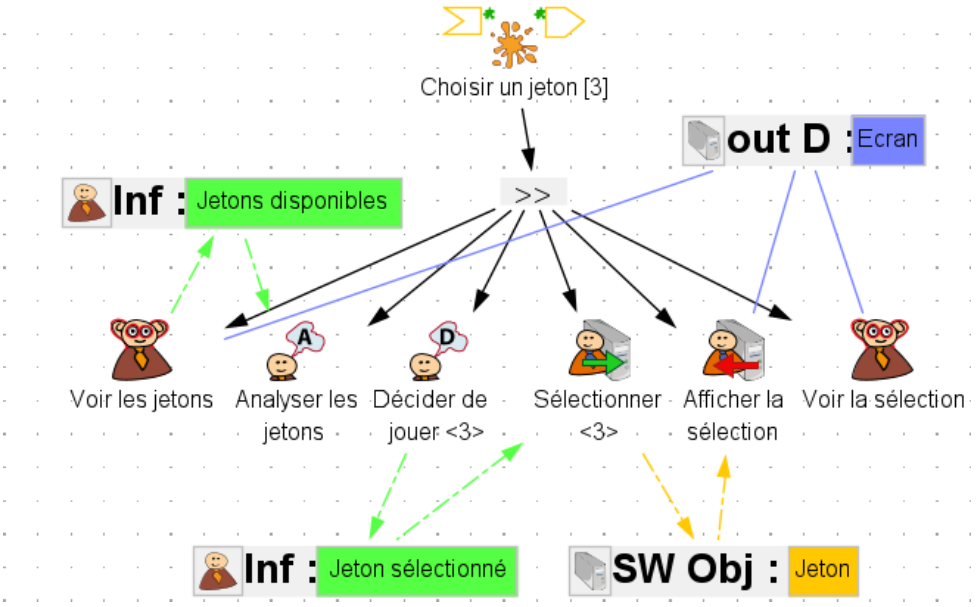


Figure 41. Composant le paramètre "jeton" instancié avec la valeur 3.

3.3 Description des données, périphériques et objets (DOD)

La notation HAMSTERS permet de typer les données, périphériques et objets requis et manipulés pour accomplir des tâches. La Figure 42 présente la représentation dans HAMSTERS des différents types de données, objets et périphériques.

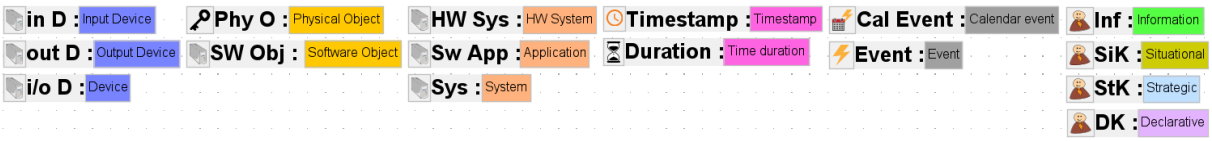


Figure 42. Représentation des données, objets et périphériques (DOD) dans HAMSTERS

3.3.1 Données et objets

- L'information (Inf) décrit une donnée mentale d'un utilisateur résultant d'une tâche perceptive, cognitive d'analyse ou de décision. Par exemple, lorsque l'utilisateur réalise une tâche cognitive de décision, une information décrivant cette décision est produite.
- Les connaissances sont manipulées par les tâches cognitives de l'utilisateur.
    - Les connaissances déclaratives (DK) décrivent le savoir nécessaire à la réalisation d'une tâche. Une connaissance déclarative peut être raffinée en connaissances situationnelles ou déclaratives (Martinie et al., 2013).
    - Les connaissances procédurales décrivent l'exercice de la connaissance pendant la réalisation de la tâche. Il est possible de décrire deux types de connaissances procédurale dans la notation HAMSTERS :
        - Les connaissances situationnelles (SiK) sont associées au raisonnement basé sur des cas et contiennent des informations spécifiques au domaine. Par exemple, une connaissance situationnelle peut être de savoir qu'en

cas d'alerte pendant le mode de conduite automatisée d'un véhicule, l'utilisateur doit immédiatement reprendre le contrôle. La Figure 43 présente la tâche « Reprendre le contrôle lors d'une alerte » pendant laquelle l'utilisateur fait un choix lié à sa connaissance situationnelle de reprendre le contrôle en cas d'alerte dans le véhicule.

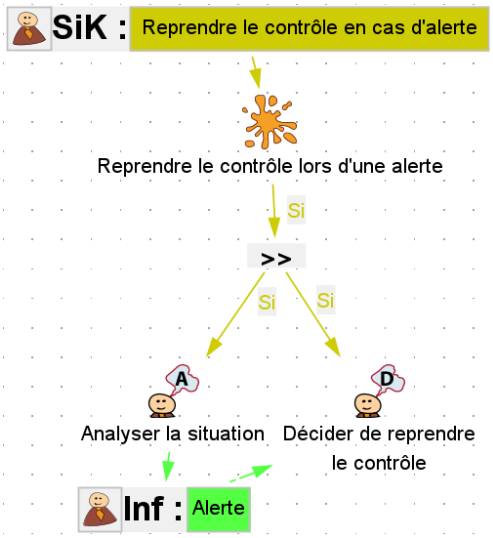


Figure 43. Exemple de modélisation d'une connaissance situationnelle dans HAMSTERS.

- Les connaissances stratégiques (StK) sont associées à un plan et une analyse des choix possibles. Par exemple, une connaissance stratégique peut être une stratégie de jeu. La Figure 44 présente la tâche « Choisir un numéro au hasard » pendant laquelle l'utilisateur réalise la stratégie de choisir un jeton au hasard.

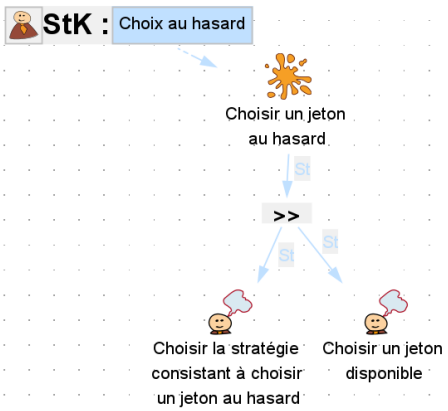


Figure 44. Exemple de modélisation d'une connaissance stratégique dans HAMSTERS.

- Les objets physiques (Phy O) décrivent les objets nécessaires à la réalisation d'une tâche ou d'une fonction. Par exemple, une clef de porte est un objet physique.
- Les objets logiciels (SW Obj) décrivent des données système. Par exemple, une liste de données est un objet logiciel.
- Hardware système (HW Sys) décrit un élément physique du système. Par exemple, une barrette de RAM est un élément hardware système.

- Application logicielle (Sw App) décrit une application logicielle ou une partie d'une application logicielle.
- Système (Sys) décrit un élément du système.
- Horodatage (Timestamp) décrit le moment auquel une tâche ou une fonction a été réalisée.
- Durée (Duration) décrit une durée pour une tâche. La Figure 45 présente un exemple d'utilisation du DOD Durée. Un symbole sablier est ajouté à la tâche pour laquelle une durée effective est définie.

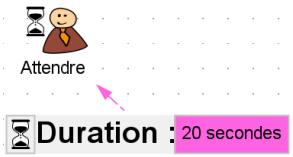


Figure 45. Exemple d'utilisation du DOD Durée.

- Evènement de calendrier (Cal Event) décrit un moment particulier déclenchant une tâche ou une fonction. Par exemple, un évènement lié à une date d'anniversaire déclenche une fonction d'affichage d'un rappel pour cet anniversaire.
- Evènement (Event) décrit un événement spécifique déclenchant une tâche ou une fonction. Par exemple, un événement émis par un système détectant les erreurs peut déclencher l'exécution d'une fonction de correction de l'erreur d'un autre système.

3.3.2 Périphériques

- Périphérique d'entrée / sortie décrit un périphérique utilisé pour réaliser une tâche décrivant une interaction.
- Périphérique d'entrée décrit un périphérique utilisé pour réaliser des tâches décrivant des interactions d'entrée. Par exemple, un clavier est un périphérique d'entrée.
- Périphérique de sortie décrit un périphérique utilisé pour des tâches décrivant des interactions de sortie. Par exemple, un écran est un périphérique de sortie.

3.3.3 Relations entre les DODs et les taches

La notation Hamsters permet de décrire les liens entre les tâches et les DODs. Ces liens sont décrits dans le Tableau 10. Les cases grisées indiquent qu'il n'y a pas de lien de relation possible entre le type de tâche et le DOD.

Tableau 10. Tableau des relations entre tâches et DOD.

DOD	**Relation selon le type de tâche**			
	Abstraite	**Utilisateur**	**Interactive**	**Système**
Système	Utilise	Utilise	Utilise	Utilise
HW Système	Utilise	Utilise	Utilise	Utilise
Périphérique d'entrée, de sortie et d'entrée/sortie	Utilise	Utilise	Utilise	Utilise
Application logicielle	Utilise	-	-	-

Information	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	**Sauf pour les tâches interactives de sortie :** Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	-
Connaissance	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	**Sauf pour les tâches interactives de sortie :** Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	
Objet Logiciel	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste		Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste
Objet Physique	Produit, consomme (une instance), accède, relâche, utilise (une instance)	Produit, consomme (une instance), accède, relâche, utilise (une instance)	Produit, consomme (une instance), accède, relâche, utilise (une instance)	Produit, consomme (une instance), accède, relâche, utilise (une instance)
Evènement	Trigger (catch), Fire (summon)	Trigger (catch), Fire (summon)	Trigger (catch), Fire (summon)	Trigger (catch), Fire (summon)
Evènement de calendrier	Trigger	Trigger	Trigger	Trigger
Horodatage	Ecrit, lit	Ecrit, lit	Ecrit, lit	Ecrit, lit
Duration	Ecrit, lit	Ecrit, lit	Ecrit, lit	Ecrit, lit
DOD personnalisé	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	Produit, consomme (une instance), accède, stocke, modifie (les attributs d'une instance), teste	Produit, consomme (une instance), accède, stocke, modifie (les attributs

				d'une instance), teste

3.4 Outil HAMSTERS

L'outil logiciel HAMSTERS permet d'éditer des modèles de tâches en utilisant les différents éléments de notation HAMSTERS présentés dans les sections précédentes. L'outil HAMSTERS permet également de simuler les modèles de tâches.

3.4.1 Edition de modèle de tâches

L'interface de l'outil HAMSTERS présentée dans la Figure 46 est composée :

1. D'un navigateur de projets comprenant : les modèles de tâches, les sub-routines, les composants, la concept map de l'application, les scénarios, les acteurs et les descriptions des formations.
   La concept map est une représentation graphique des connaissances constituée de concepts et des relations entre ces concepts (Ragosta, 2015).
2. D'un cadre central permettant d'éditer et de consulter plusieurs modèles de tâches à la fois.
3. D'une palette d'édition permettant de glisser/déposer les éléments de la notation.
4. D'une fenêtre d'édition des propriétés d'un élément de la notation.
5. D'une vue arborescente du modèle de tâches édité.

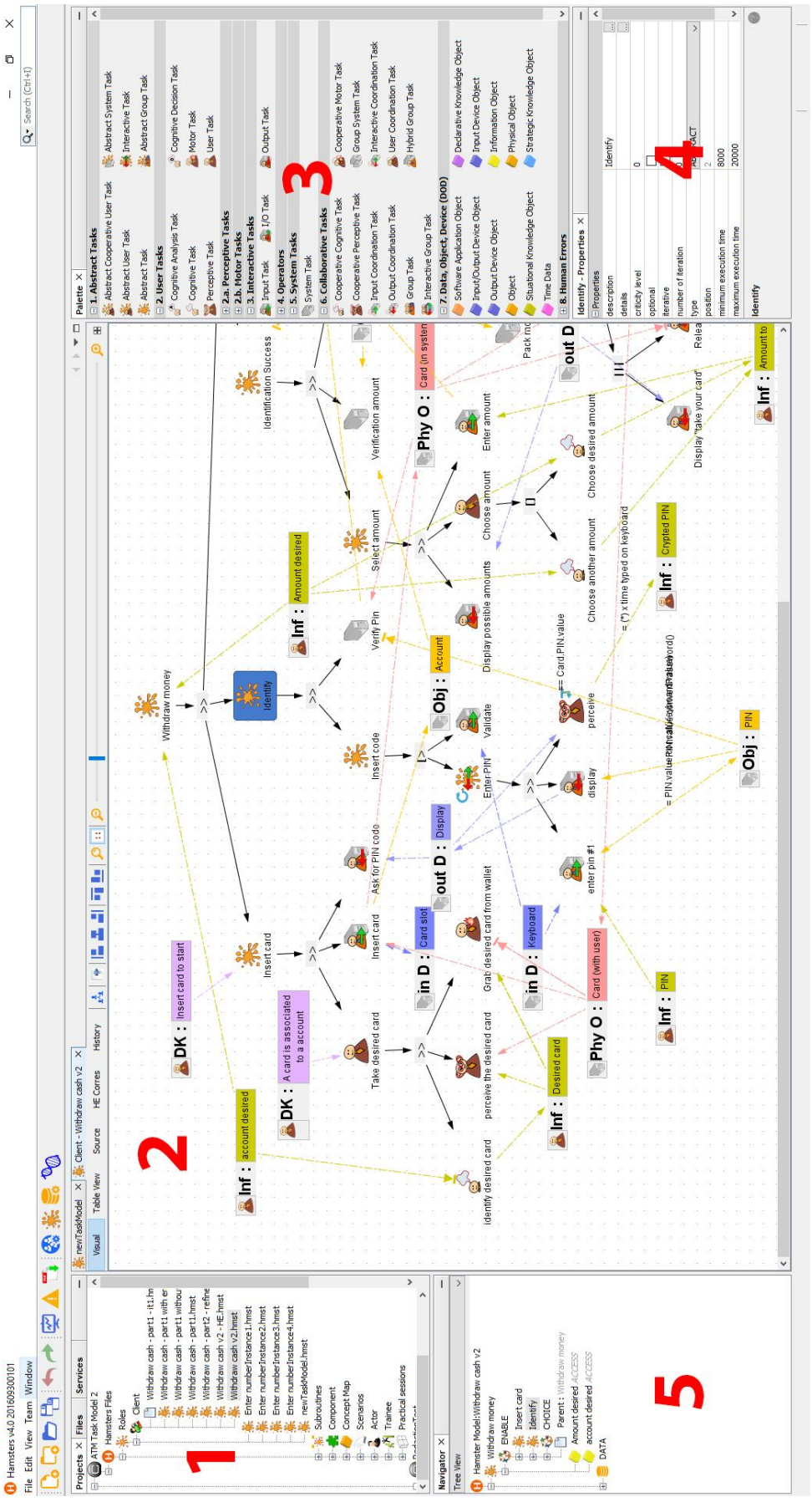


Figure 46. Vue globale de l'éditeur de tâches HAMSTERS

3.4.2 Simulation de modèle de tâches

L'outil HAMSTERS fournit un support graphique à la simulation des modèles de tâches.

La Figure 47 montre la simulation des modèles de tâches dans l'outil HAMSTERS. Un panneau de contrôle de la simulation est affiché sur la droite de l'outil. Ce panneau de contrôle de la simulation contient la liste des tâches pouvant être réalisées (1- liste de la partie supérieure) et la liste des tâches déjà réalisées ordonnées selon leur ordre de réalisation (3- liste de la partie inférieure). Pour chaque tâche réalisée, le modèle de tâches dans lequel elle est décrite est indiqué. La zone centrale (2) permet au concepteur de saisir des données contextuelles sur l'exécution d'une tâche. Par exemple, le concepteur peut saisir la valeur de la donnée nécessaire à la réalisation d'une tâche.

Dans la partie contenant la liste des modèles de tâches du projet, le concepteur peut définir un modèle de tâches principal pour chaque rôle décrit. Le modèle de tâches principal de chaque rôle est exécuté en simulation. Pour définir un modèle de tâches principal pour un rôle, il faut réaliser un clic droit sur le modèle de tâches souhaité et sélectionner « Set as Main Role Model ». L'icône indique que le modèle de tâches et le modèle de tâches principal pour le rôle.

Lors de la simulation, les tâches des modèles de tâches exécutés apparaissent différemment selon leur disponibilité :

- Les tâches réalisées apparaissent avec le symbole suivant .
- Les tâches disponibles sont surlignées en vert

.
- Les tâches désactivées par des choix effectués par le concepteur apparaissent surlignées en rouge

.

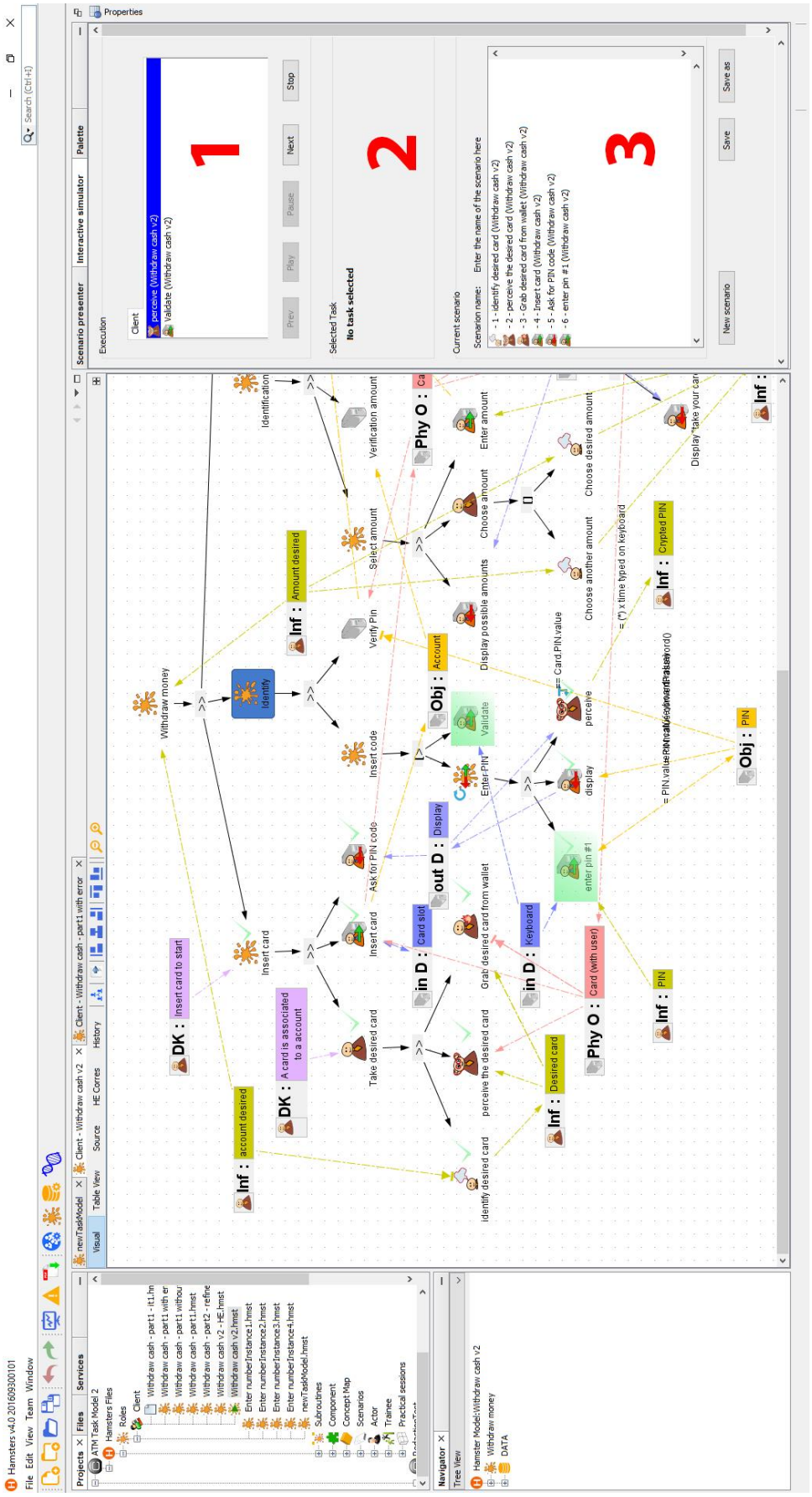


Figure 47. Copie d'écran d'une simulation de modèles de tâches avec l'outil logiciel HAMSTERS (exemple de tâches effectuées et actives)

4 Conclusion

Dans ce chapitre, nous avons présentés différents modèles de l'humain et de ses tâches. Également, nous avons présentés des notations et des outils existants pour la modélisation de tâches. Nous avons aussi présenté de manière détaillée la notation et l'outil associé HAMSTERS. Les notations et outils présentés ne permettent pas de décrire explicitement les allocations des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre les utilisateurs et les systèmes. Nous présentons dans les chapitres suivants détaillant les contributions de cette thèse, un cadre conceptuel proposant de décrire les allocations des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité par rapport aux allocations de tâches et de fonctions (RCRAFT) décrits dans des modèles de tâches. Nous proposons des extensions à la notation et l'outil HAMSTERS afin de soutenir le processus d'analyse et de description RCRAFT.

Partie II. Contribution

Chapitre 4. Cadre conceptuel pour l'analyse des systèmes interactifs contenant de l'automatisation et des tâches de leurs utilisateurs

Dans ce chapitre, nous définissons les éléments décrivant l'automatisation retenus après l'étude de l'état de l'art (voir Chapitre 2) pour une approche basée modèles de l'analyse de systèmes interactifs contenant de l'automatisation. Nous proposons de décrire les caractéristiques d'un ou plusieurs systèmes contenant de l'automatisation et des tâches de leurs utilisateurs à travers le cadre conceptuel RCRAFT qui permet d'analyser l'allocation :

- Des **R**essources,
- Des transitions de **C**ontrôle et de l'initiative,
- De la **R**esponsabilité,
- De l'**A**utorité,
- Des **F**onctions et des **T**âches

entre les rôles des acteurs humains et systèmes.

Nous proposons dans ce chapitre des définitions de chacun de ces éléments que nous utilisons dans la suite de cette thèse. Ces définitions sont issues de l'analyse de l'état de l'art présenté au Chapitre 2. Nous avons adapté ces définitions à un grain fin de description des tâches et des fonctions. Ces définitions nous permettent l'analyse et la description de chacun de ses éléments dans une approche à base de modèles de tâches. Ces définitions peuvent être adaptées au domaine d'application.

Ce cadre conceptuel nous permet d'analyser des systèmes interactifs de commande et contrôle contenant de l'automatisation par la description explicite au niveau des tâches et des fonctions de chacun des éléments RCRAFT.

Ce chapitre se décompose en six sections :

1. Dans un premier temps, l'exemple illustratif du jeu des 15 est présenté. Cet exemple est utilisé dans chacune des sections suivantes afin d'illustrer chacun des éléments RCRAFT.
2. Dans la seconde section, nous présentons la définition de la description de l'allocation des fonctions et des tâches entre les rôles des acteurs que nous utilisons dans notre approche basée modèles.
3. Dans la troisième section, nous présentons la définition de la description de l'allocation des ressources entre les rôles des acteurs utilisée dans le cadre conceptuel RCRAFT.
4. Dans la quatrième section, nous présentons la définition de la description de l'allocation de l'autorité entre les rôles des acteurs utilisée dans le cadre conceptuel RCRAFT. La définition de la description de l'allocation de l'autorité étant nécessaire pour définir la description de l'allocation des transitions de contrôle et de l'initiative, nous définissons

d'abord la description de l'allocation de l'autorité avant la description de l'allocation des transitions de contrôle et de l'initiative.

5. Dans la cinquième section, nous présentons la définition de la description de l'allocation des transitions de contrôle et de l'initiative entre les rôles des acteurs utilisée dans le cadre conceptuel RCRAFT.
6. Dans la sixième section, nous présentons la définition de la description de l'allocation de la responsabilité entre les rôles des acteurs utilisée dans le cadre conceptuel RCRAFT.
7. La septième section conclut ce chapitre.

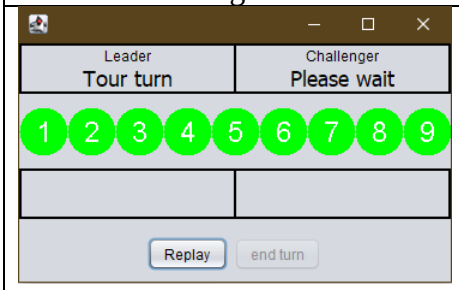
1 Un exemple illustratif : Le jeu des 15

Cette section présente l'exemple illustratif du jeu des 15 utilisé dans la suite de ce chapitre et dans les prochains chapitres de cette thèse. Cet exemple a été choisi pour sa simplicité malgré la nécessité d'au moins trois différents rôles pour ce jeu : le joueur qui commence, le second joueur et l'arbitre gérant la partie. Différentes tâches de ce jeu peuvent être automatisées : la détection d'un gagnant, la gestion des tours de jeu, la sélection d'un jeton par un joueur géré par le système, etc. C'est-à-dire que ces tâches peuvent être migrées vers un système contenant de l'automatisation. Nous présentons la version de ce jeu utilisée à des fins d'illustration dans la suite de ce chapitre.

1.1 Présentation du jeu des 15

Le jeu des 15 est un jeu se jouant avec deux joueurs. Ils disposent de jetons représentant les chiffres de 1 à 9. Chacun leur tour, ils doivent choisir un jeton qui devient indisponible pour le joueur adverse. Lorsque l'un des deux joueurs possède trois jetons dont la somme est égale à 15, la partie est terminée et ce joueur gagne. Ici, nous considérons que deux joueurs humains jouent l'un contre l'autre à travers un système permettant de jouer au jeu des 15. Le système indique les tours de jeux, permet de mettre fin à un tour de jeu, de rejouer et affiche les jetons disponibles et joués par les deux joueurs. Les joueurs peuvent choisir de rejouer une partie à tout moment. Le Tableau 11 résume la version du jeu des 15 que nous allons utiliser à des fins d'illustration dans la suite de ce chapitre.

Tableau 11. Jeu des 15 utilisé à des fins d'illustration du cadre conceptuel RCRAFT

Interface et Désignation	Description
 Le jeu des 15	Les jetons disponibles, les jetons joués par les joueurs et le joueur courant sont affichés par le système. Lorsqu'un joueur sélectionne un jeton, le bouton pour mettre fin au tour devient disponible. Le tour de jeu se termine lorsque le joueur presse ce bouton et c'est à l'adversaire de jouer. Les joueurs peuvent à tout moment décider de rejouer une partie ou d'arrêter.

1.2 Concept map du jeu des 15

Pour jouer au jeu des 15, différents rôles doivent être distribués, des ressources et des règles sont requises. La Figure 48 présente la *concept map* du jeu des 15. Une *concept map* est un réseau sémantique représenté graphiquement qui peut être utilisée pour représenter des

connaissances. Nous avons représenté dans cette *concept map* : les ressources, les rôles et les règles pour jouer au jeu des 15. Pour jouer au jeu des 15 avec la version du jeu présentée ci-avant, il faut deux joueurs et un arbitre. Les deux joueurs prennent le rôle de *Leader* ou de *Challenger* selon si le joueur commence à jouer ou non. L'arbitre gère les jetons du jeu et les tours de jeu. Ces rôles sont indiqués en violet dans la *concept map* du jeu des 15 présentée dans la Figure 48. L'arbitre fait respecter les règles pour lancer une partie et pour jouer. Les joueurs gèrent la fin de jeu selon les règles de fin de jeu du jeu des 15. Ces règles sont indiquées en vert dans la *concept map* du jeu des 15 présentée dans la Figure 48. Nous considérons les ressources suivantes comme nécessaires pour pouvoir jouer des parties du jeu des 15 :

- Les rôles (chacun des acteurs doit connaitre son rôle),
- Les jetons disponibles (les jetons à sélectionner de 1 à 9),
- Les jetons du *Leader*, les jetons du *Challenger*,
- Une stratégie de jeu pour chacun des 2 joueurs,
- Le jeton sélectionné à chaque tour par les joueurs,
- Le joueur courant (le joueur dont c'est le tour de jeu),
- La fin de tour (pour changer de tour de jeu),
- Le résultat du jeu,
- La possibilité de rejouer et
- Le lancement d'une nouvelle partie.

Ces ressources sont indiquées en bleu dans la *concept map* du jeu des 15 présentée dans la Figure 48.

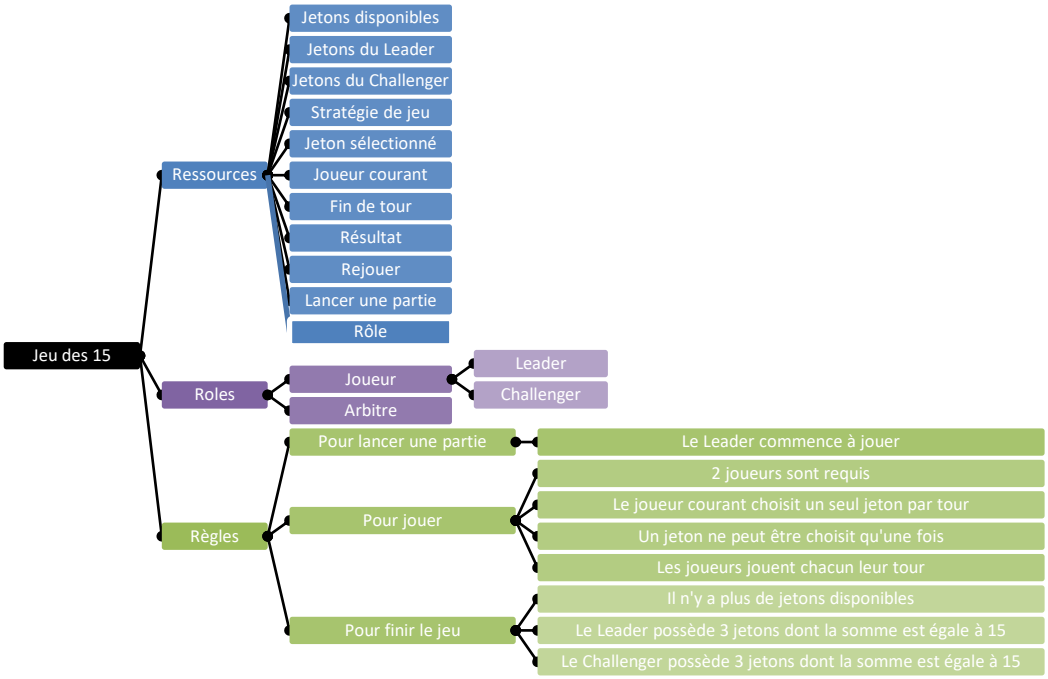


Figure 48. Concept map du Jeu des 15.

2 Fonctions et de Tâches

Dans cette section, nous présentons la description de l'allocation des fonctions et des tâches. Les différentes approches existantes traitant de l'allocation de de fonctions et de tâches ont été présentées au Chapitre 2.3. Nous débutons en rappelant la définition des fonctions et des tâches. Puis, nous proposons la définition de la description de l'allocation des fonctions et des tâches dans le cadre conceptuel RCRAFT. Puis, nous explicitons ces caractéristiques et nous décrivons l'allocation des fonctions et des tâches sur l'exemple du jeu des 15.

2.1 Définitions de fonctions et de tâches

Nous utilisons les définitions de fonctions et de tâches adaptées de (Paterno et al., 1997) présentées au Chapitre 3.2.2. Nous les rappelons ci-après.

Définition d'une tâche adaptée :

> *On appelle **tâche** ce qui est réalisé par un utilisateur pour atteindre un but dans un domaine d'application spécifié.*

Définition d'une fonction adaptée :

> *On appelle **fonction** ce qui est exécuté par un système pour soutenir un but utilisateur dans un domaine d'application spécifié.*

Les tâches des utilisateurs sont décomposées en tâches :

- Perceptive (perceptif abstrait, perceptif),
- Cognitive (cognitif abstrait, analyse, décision),
- Motrice (moteur abstrait, moteur),
- Coopérative (coopératif abstrait, perceptif coopératif, moteur coopératif),
- Abstraite

Les fonctions des systèmes sont décomposées en fonctions :

- D'entrée,
- De traitement (traitement abstrait, traitement),
- De sortie,
- Coopérative (abstraite coopérative, entrée coopérative, sortie coopérative),
- Abstraite

2.2 Définition de la description de l'allocation des Fonctions et des Tâches

Définition de la description de l'allocation des fonctions et des tâches à partir des travaux présentés au Chapitre 2.3 et des principes d'analyse et de description de tâches présentés au Chapitre 3.2 :

> *On appelle description de l'allocation des fonctions et des tâches l'identification des tâches réalisées par les rôles des acteurs utilisateur et des fonctions exécutées par les rôles des acteurs systèmes.*
>
> *La description de l'allocation des fonctions et des tâches permet d'identifier l'ordonnancement temporel entre les tâches et fonctions identifiées.*
>
> *La description de l'allocation des fonctions et des tâches inclut l'identification des données, objets, périphériques requis pour la réalisation des tâches et l'exécution des fonctions.*

Nous proposons de décrire de manière complète et non ambigüe les tâches et les fonctions allouées aux rôles des acteurs lors de l'activité d'allocation des tâches et des fonctions (voir Chapitre 2.3). Nous proposons également de décrire les différents types de tâches et de fonctions ainsi que l'ordenancement temporel entre ces fonctions et ces tâches de la même manière que les techniques d'analyse et de modélisation des tâches utilisateurs présentées au Chapitre 3.2.

2.3 Rôles des utilisateurs et des systèmes, buts, finalités et abilités

Les fonctions et les tâches identifiées lors de la description de l'allocation des fonctions et des tâches permettent d'accomplir différent buts associés aux rôles assignés des acteurs. Un acteur utilisateur peut se voir assigner plusieurs rôles et un ou plusieurs buts. De la même manière, un acteur système, peut se voir assigner plusieurs rôles pour soutenir un ou plusieurs buts utilisateur. Comme défini au Chapitre 3.2.3.1, un acteur représente une entité capable d'effectuer un ensemble de tâches ou de fonctions. Cette capacité à réaliser des tâches et des fonctions est liée aux abilités de l'acteur. (Frank Flemisch et al., 2012) définit l'abilité comme la possession des moyens et des compétences pour percevoir et/ou sélectionner une action et/ou agir de façon appropriée. Cela inclut les caractéristiques physiques, physiologiques, les connaissances, les compétences ou l'expérience par exemple dans le cas d'un acteur humain. Pour un acteur système, cela inclut les données auxquelles le système a accès, son architecture, ses caractérisques hardwares et ses fonctionnalités par exemple. L'abilité des acteurs n'est pas un élément du cadre conceptuel RCRAFT car l'abilité des acteurs évolue avec l'entrainement et les progrès technologiques. Également, les tâches et les fonctions alouées aux rôles des acteurs décrivent leurs abilités : les capacités des acteurs pour réaliser leurs buts.

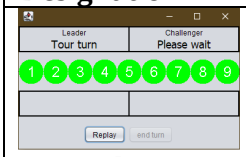
Contrairement à des approches telle que MABA MABA (présentée au Chapitre 2.3), nous n'incluons pas dans la description de l'allocation des fonctions et des tâches l'analyse de quel acteur est le meilleur pour réaliser la tâche à allouer. Nous proposons de décrire l'allocation des fonctions et des tâches effective sans évaluer la qualité de cette allocation. La qualité de cette allocation des fonctions et des tâches étant liée à la conception du ou des systèmes, elle ne peut

être évaluée que partiellement par une approche basée modèles et doit être complétée par une évaluation impliquant des utilisateurs.

2.4 Description de l'allocation des Fonctions et des Tâches sur l'exemple du jeu des 15

Le Tableau 12 décrit l'allocation des fonctions et des tâches entre les rôles attribués aux trois acteurs impliqués pour jouer au jeu des 15 : les deux utilisateurs et le système interactif contenant de l'automatisation. Le rôle d'Arbitre est attribué à l'acteur système. Les utilisateurs ont le rôle de Joueurs. L'Arbitre attribue avant le début de la partie les rôles de *Challenger* et de *Leader* aux Joueurs. Le *Leader* joue en premier. Le *Leader* et le *Challenger* veulent atteindre le même but : Gagner la partie. L'arbitre soutient le but des utilisateurs en faisant respecter les règles du jeu des 15.

Tableau 12. Distribution des tâches et des fonctions pour le Jeu des 15

Désignation	**Description des Tâches et des Fonctions**
 Le jeu des 15	**Rôle Joueur :** • Le joueur voit le bouton pour lancer la partie. • Les deux joueurs collaborent pour décider qui lance la partie. Selon le joueur choisi : o Le joueur clique avec la souris sur le bouton pour lancer la partie, ou o Le joueur voit l'autre joueur lancer la partie • Etre assigné à un rôle pour le jeu : o Le joueur voit l'allocation des rôles o Il analyse l'indication et comprend son rôle (*Leader* ou *Challenger*) Le joueur réalise ensuite les tâches du rôle que l'arbitre lui a alloué.

	Rôle « Leader » : • Attendre : Le Leader voit la requête d'attente du système, analyse cette requête et comprend qu'il doit attendre. • Voir le jeu : Après que le système ait affiché le jeu, le *Leader* voit le jeu via l'écran (le joueur courant, les jetons disponibles, le jetons joués par le Challenger et le Leader vides, le bouton fin de tour désactivé, le bouton rejouer), • Il analyse les informations affichées et comprend que c'est à lui de commencer le jeu. Jouer les tours de jeu (itératif) : • Jouer son tour : o Le *Leader* trouve des alternatives de jeu en appliquant ou modifiant sa stratégie de jeu et choisit un jeton. Le *Leader* sélectionne le jeton en déplaçant et en cliquant avec la souris sur un des jetons disponibles. o Le *Leader* voit qu'il est possible de mettre fin au tour (le jeton sélectionné est dans ses jetons joués et retiré des jetons disponibles, le bouton fin de tour est disponible). Il analyse que le jeton sélectionné est bien joué et décide de mettre fin à son tour. • Mettre fin à son tour : o Le Leader clique avec la souris sur le bouton fin de tour. o Il voit que : le bouton fin de tour est désactivé, les jetons disponibles sont activés, le tour du Challenger est affiché. o Il analyse ces informations et comprend que c'est le tour du Challenger. • Attendre pendant le tour du Challenger : o Le Leader peut voir les actions du Challenger pendant son tour et les affichages de l'arbitre. Il peut analyser les informations vues et comprendre que le Challenger va mettre fin à son tour. • Attendre la fin de tour du Challenger : o Le Leader voit les actions du Challenger pendant qu'il termine son tour et les affichages de l'arbitre. o Il analyse les informations affichées et décide que c'est à son tour de jouer. Résultat du jeu et relancer une partie (interrompt « Jouer les tours de jeu ») : • Le Leader peut décider que la fin de partie est atteinte. • Ensuite il collabore avec le Challenger pour mettre fin à la partie. Il analyse la possibilité de rejouer. Selon sa volonté de rejouer, il peut : • Rejouer : o Le Leader décide de rejouer. o Les joueurs se mettent d'accord pour rejouer. o Le Leader peut cliquer avec la souris sur le bouton rejouer s'il est le joueur qui relance la partie. • Arrêter de jouer : o Le Leader décide de rejouer. o Les joueurs se mettent d'accord pour rejouer.

	Rôle « Challenger » : *Le Challenger réalise les mêmes tâches que le Leader mais commence l'itération pour « Jouer les tours » avec la tâche : Attendre pendant le tour du Leader.*
	Rôle Arbitre : • L'arbitre commence par lancer le jeu : o L'arbitre initialise le bouton pour lancer une partie et l'active. o L'arbitre affiche le bouton pour lancer la partie. o L'arbitre reçoit la pression sur le bouton pour lancer la partie. Soutenir les matchs de jeu (itératif) • Allouer les rôles aux joueurs : o L'arbitre alloue aléatoirement les rôles aux deux joueurs. o L'arbitre affiche les rôles pour les joueurs. • Initialiser le jeu : o L'arbitre affiche la requête d'attente. o Puis, il initialise les différents objets : le bouton pour rejouer, les jetons disponibles, les jetons du Leader, les jetons du Challenger, le joueur courant, le bouton pour mettre fin au tour et le résultat du jeu. Il désactive le bouton fin de tour. Il active les jetons disponibles et le bouton rejouer. • Afficher le jeu : o L'arbitre affiche : les jetons disponibles, la zone des jetons joués par le Leader, la zone des jetons joués par le Challenger, le joueur courant, le bouton fin de tour désactivé, le bouton rejouer activé. Il masque la requête d'attente et le résultat du jeu. Soutenir la partie (itératif) • Soutenir le tour du Leader : o L'arbitre reçoit le jeton sélectionné par le biais de la pression d'un widget représentant les jetons disponibles. o L'arbitre traite la sélection, active le bouton pour mettre fin au tour et désactive les jetons disponibles. o L'arbitre affiche le jeton sélectionné dans les jetons joués par le Leader et le retire des jetons disponibles. Il affiche que le bouton fin de tour est disponible. • Soutenir la fin de tour du Leader : o L'arbitre reçoit la fin de tour du Leader par le biais de la pression du bouton fin de tour. o L'arbitre modifie le joueur courant, désactive le bouton fin de tour et active les jetons disponibles. o L'arbitre affiche le tour suivant : il affiche le bouton fin de tour désactivé, les jetons disponibles activés et le Challenger en tant que joueur courant. • Soutenir le tour du Challenger : o L'arbitre reçoit le jeton sélectionné par le biais de la pression d'un widget représentant les jetons disponibles. o L'arbitre traite la sélection, active le bouton pour mettre fin au tour et désactive les jetons disponibles.

	○ L'arbitre affiche le jeton sélectionné dans les jetons joués par le Challenger et le retire des jetons disponibles. Il affiche que le bouton fin de tour est disponible. • Soutenir la fin de tour du Challenger : ○ L'arbitre reçoit la fin de tour du Challenger par le biais de la pression du bouton fin de tour. ○ L'arbitre modifie le joueur courant, désactive le bouton fin de tour et active les jetons disponibles. ○ L'arbitre affiche le tour suivant : il affiche le bouton fin de tour désactivé, les jetons disponibles activés et le Leader en tant que joueur courant. • Relancer une partie (interrompt la fonction itérative « Soutenir la partie ») : ○ L'arbitre reçoit la pression sur le bouton rejouer ○ L'arbitre traite la requête pour rejouer

2.5 Relations de la description de l'allocation des fonctions et des tâches avec les autres éléments du cadre conceptuel RCRAFT

A partir de la description de l'allocation des fonctions et des tâches, il est possible d'analyser le ou les systèmes interactifs contenant de l'automatisation et les tâches de leurs utilisateurs relativement à l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre les rôles des acteurs. La description de l'allocation des ressources résulte de l'analyse des données, objets et périphériques requis pour la réalisation des tâches et des fonctions. La description de l'allocation des transitions de contrôle et de l'initiative résulte de l'analyse des tâches et fonctions coopératives des rôles des acteurs. La description de l'allocation de l'autorité résulte de l'analyse des tâches et des fonctions contraignant et réalisant les actions (ex. un ordre, une restriction, un conseil, une délégation, …). La description de l'allocation de la responsabilité résulte de l'analyse des modifications apportées aux données, objets, périphériques qui peuvent causer une déviation du résultat attendu. Ces concepts composant le reste du cadre conceptuel RCRAFT sont présentés dans la suite de ce chapitre.

3 Ressources

Dans cette section, nous définissons la description de l'allocation des ressources. Les approches existantes traitant de l'allocation des ressources sont présentées au Chapitre 2.4.2. Nous débutons en définissant le terme « ressource » du cadre conceptuel RCRAFT. Puis, nous présentons la définition de la description de l'allocation des ressources. Ensuite, nous explicitons ces caractéristiques et nous décrivons l'allocation des ressources sur l'exemple du jeu des 15.

3.1 Définition de Ressource

Nous proposons de définir les ressources du cadre conceptuel RCRAFT de la façon suivante :

On appelle ressource un artefact manipulé par une tâche ou une fonction par un plusieurs acteurs.

Par exemple la ressource « jetons disponibles » du jeu des 15 est présentée sous la forme d'un élément de la couche interface par le système, elle est perçue sous forme d'information par les utilisateurs et elle est traitée sous forme d'objet logiciel par le système.

3.2 Définition de la description de l'allocation des Ressources

Nous utilisons les données, objets et périphériques (DODs) définis par la notation HAMSTERS présentées au Chapitre 3.3.3 pour décrire l'allocation des ressources entre les acteurs. Nous définissons la description de l'allocation des ressources afin de pouvoir décrire ce qui est partagé ou non entre les rôles et ainsi décrire les aspects liés aux partages entre les utilisateurs et les systèmes contenant de l'automatisation relatif à la compréhension de leur comportement mutuel de la transparence présenté au Chapitre 2.4.2.

Définition de la description de la description de l'allocation des ressources :

On appelle description de l'allocation des ressources l'identification des possesseurs des ressources et les acquéreurs des ressources si elles sont partagées entre les rôles des acteurs utilisateurs ou systèmes à travers des données, périphériques et objets (DOD) requis pour réaliser les tâches utilisateur et exécuter les fonctions système.

3.3 Identification des instances, des rôles et acteurs possesseurs et acquéreurs des ressources

La description de l'allocation des ressources permet de définir qui sont les possesseurs d'une ressource et quelles sont les ressources partagées.

Les acquéreurs d'une ressource peuvent être :

- Tous les acteurs (toutes les instances des rôles),
- Tous les rôles (au moins une instance de chaque rôle),
- Un groupe restreint d'acteurs (un groupe d'instances de rôles),
- Un groupe de rôle (au moins une instance d'un groupe de rôles),
- Un seul acteur (une instance d'un rôle),
- Un seul rôle (toutes les instances d'un rôle),
- Aucun acteur : la ressource n'est pas partagée.

Une ressource peut être partagée entre les rôles des acteurs par des tâches et fonctions coopératives. Nous présentons la définition des tâches et fonctions coopératives utilisée lors de cette thèse au Chapitre 5.2.

Lorsque la ressource est partagée entre différents rôles ou acteurs par une tâche ou une fonction coopérative, elle peut être :

- Toujours partagée : l'allocation des fonctions et des tâches ne décrit pas de condition sur les tâches et fonctions coopératives permettant de partager la ressource.
- Partagées sous conditions : l'allocation des fonctions et des tâches décrit des conditions sur les tâches et fonctions coopératives permettant de partager la ressource.

- Non-partagées : l'allocation des fonctions et des tâches ne décrit pas de tâches et fonctions coopératives permettant de partager la ressource.

Les ressources sont instanciées par les DODs manipulés par les rôles :

- Les données (information perçue ou résultant d'une analyse ou concernant une décision, une connaissance, une connaissance stratégique ou tactique),
- Les périphériques (entrée ou de sortie),
- Les objets (objet logiciels produit lors d'une fonction d'entrée ou produit ou modifié par une fonction de traitement, les éléments de la couche interface et les objets physiques),

Une ressource partagée n'est pas possédée par les différents rôles des acteurs à travers une unique instance. La ressource est partagée par le biais d'un (ou plusieurs) DOD commun à plusieurs rôles des acteurs. Ces DODs peuvent décrire la modalité du partage de ressource ou le périphérique d'entrée ou de sortie utilisé lors du partage de ressource. Cette ressource est émise ou reçue par les rôles des acteurs lors de tâches et fonctions coopératives. Ensuite, chacun des rôles des acteurs avec lesquels la ressource est partagée, possède une nouvelle instance de la ressource : l'information perçue pour les acteurs utilisateur et la donnée acquise par les acteurs systèmes.

3.4 Description de l'allocation des Ressources sur l'exemple du jeu des 15

Le Tableau 13 décrit l'allocation des ressources entre les rôles assignés aux acteurs utilisateurs et système pour le jeu des 15. Les ressources relatives aux jetons disponibles, aux jetons sélectionnés et joués par les deux joueurs, à la fin d'un tour et à la volonté de rejouer sont toujours partagées entre tous les rôles par le biais des périphériques d'entrée (souris) et de sortie (écran) et des éléments de l'interface du système. Le rôle *Leader* assigné à un utilisateur ne partage jamais sa stratégie de jeu (connaissance stratégique). De la même manière, le rôle *Challenger* ne partage jamais sa stratégie de jeu (connaissance stratégique). Les ressources relatives au gagnant de la partie sont partagées entre les rôles *Leader* et *Challenger* par le biais de la parole.

Tableau 13. Distribution des Ressources pour le jeu des 15

Ressources	Instance de la ressource	Rôle(s) Possesseur(s)	Rôle (s) Acquéreur(s)
Jetons disponibles	Couche Interface « Widgets des Jetons Disponibles »	Arbitre	Leader, Challenger
	Objet Logiciel « Jetons disponibles »	Arbitre	
	Information « Jetons disponibles »	Leader, Challenger	
Jeton sélectionné	Couche Interface « Widgets des Jetons Disponibles »	Arbitre	Leader, Challenger
	Objet Logiciel « Jeton sélectionné par le Leader »	Arbitre	

	Objet Logiciel « Jeton sélectionné par le Challenger »	Arbitre	
	Information « Jeton sélectionné »	Leader, Challenger	
	Information « Alternatives de choix de jetons possibles »	Leader, Challenger	
Jetons du Leader	Couche Interface « Widgets des jetons joués par le Leader »	Arbitre	Leader, Challenger
	Objet logiciel « Jetons joués par le Leader »	Arbitre	
	Information « jetons joués »	Leader, Challenger	
	Information « jetons joués par le Leader »	Challenger	
Jetons du Challenger	Couche Interface « Widgets des jetons joués par le Challenger»	Arbitre	Leader, Challenger
	Objet logiciel « Jetons joués par le Challenger»	Arbitre	
	Information « jetons joués »	Challenger	
	Information « jetons joués par le Challenger»	Leader	
Fin de tour	Couche Interface « Bouton fin de tour »	Arbitre	Leader, Challenger
	Objet Logiciel « Fin de tour »	Arbitre	
	Information « Bouton fin de tour disponible »	Leader, Challenger	
	Information « Volonté de mettre fin à son tour »	Leader, Challenger	
	Information « Bouton fin de tour désactivé »	Leader, Challenger	
	Information « Le tour est terminé »	Leader, Challenger	
Joueur Courant	Couche Interface « Texte d'indication des tours des joueurs »	Arbitre	Leader, Challenger
	Couche Interface « Texte demandant aux joueurs d'attendre »	Arbitre	Leader, Challenger
	Objet Logiciel « Joueur courant »	Arbitre	
	Information « Je commence à jouer »	Leader	
	Information « Je dois attendre »	Challenger	
	Information « Le Leader doit attendre »	Leader	
	Information « Le Challenger doit jouer »	Challenger	
	Information « Le Leader doit jouer »	Leader	
	Information « Le Challenger doit attendre »	Challenger	
	Information « Joueur courant »	Leader, Challenger	
	Information « Requête d'attente »	Leader, Challenger	

	Information « Requête analysée »	Leader, Challenger	
Résultat de la partie	Parole « Fin de partie »	Leader, Challenger	Leader, Challenger
	Information « Etat de la partie »	Leader, Challenger	
	Information « Résultat du jeu »	Leader, Challenger	
Rejouer	Couche Interface « Bouton Rejouer »	Arbitre	Leader, Challenger
	Parole « Rejouer »	Leader, Challenger	Leader, Challenger
	Parole « Arrêter de jouer »	Leader, Challenger	Leader, Challenger
	Objet Logiciel « Rejouer »	Arbitre	
	Information « Volonté de rejouer »	Leader, Challenger	
	Information « Bouton pour rejouer »	Leader, Challenger	
	Information « Rejouer »	Leader, Challenger	
	Information « Rôle qui prend la souris pour lancer une nouvelle partie »	Leader, Challenger	
Stratégie de jeu	Information Stratégique « Stratégie de jeu »	Leader, Challenger	
Stratégie d'attribution des rôle	Données Stratégique « Aléatoire »	Arbitre	
Lancer partie	Couche Interface « Bouton pour lancer la partie »	Arbitre	Joueur
	Parole « Lancer la partie »	Leader, Challenger	Leader, Challenger
	Objet Logiciel « Lancer partie »	Arbitre	
	Information « Bouton pour lancer une partie »	Joueur	
	Information « Joueur qui presse le bouton lancer la partie »	Joueur	
Rôle	Couche Interface « Texte décrivant l'allocation des rôles aux joueurs »	Arbitre	Joueur
	Objet Logiciel « Rôles des joueurs »	Arbitre	
	Objet Logiciel « Challenger »	Arbitre	
	Objet Logiciel « Leader »	Arbitre	
	Information « Distribution des rôles affichée »	Joueur	
	Information « Rôle »	Joueur	

4 Autorité

La définition de la description de l'allocation des transitions de contrôle et de l'initiative requiert la définition de l'autorité. Nous présentons donc d'abord définition de l'autorité dans le cadre conceptuel RCRAFT. Les approches existantes traitant de l'allocation de l'autorité sont décrites au Chapitre 2.4.1.2.

Dans cette section, nous définissons la description de l'allocation de l'autorité dans le cadre conceptuel RCRAFT. Nous débutons en définissant le terme « autorité » du cadre conceptuel RCRAFT. Puis, nous présentons la définition de la description de l'allocation de l'autorité.

Ensuite, nous présentons des exemples de contraintes possibles définies par des tâches ou fonctions à autorité. Puis, nous décrivons l'allocation de l'autorité sur l'exemple du jeu des 15.

4.1 Définition de l'Autorité

(Frank Flemisch et al., 2012) définit l'autorité comme exercer le contrôle ou une certaine distribution du contrôle. Nous proposons de raffiner cette définition afin de décrire explicitement certain type de tâche où le contrôle s'exerce. C'est-à-dire, les tâches ou les fonctions par lesquelles l'autorité de contraindre les tâches ou les fonctions d'un rôle est exercée pour influencer le résultat de l'exécution de l'ensemble de ces tâches et fonctions.

Nous proposons de définir l'autorité du cadre conceptuel RCRAFT de la façon suivante :

On appelle autorité :

- *La définition par les rôles de contraintes sur des tâches ou des fonctions des rôles des acteurs. Les contraintes sont définies sur les données, périphériques et objets manipulés par les rôles des acteurs.*
- *La réalisation par les rôles de tâches ou de fonctions modifiant l'état du système, d'un objet ou de l'environnement pour l'atteinte d'un but ou d'un sous-but.*

4.2 Définition de la description de l'allocation de l'Autorité

Définition de l'allocation de l'autorité :

On appelle description de l'allocation de l'autorité l'identification des tâches ou fonctions des rôles des acteurs qui :

- *Définissent des contraintes sur des tâches ou des fonctions des rôles des acteurs.*
- *Réalisent des tâches ou des fonctions modifiant l'état du système, d'un objet ou de l'environnement pour l'atteinte d'un but ou d'un sous-but.*

4.3 Les contraintes posées aux acteurs par les acteurs

La description de l'allocation de l'autorité est l'identification des tâches ou fonctions qui définissent une ou des contraintes sur les tâches et fonctions à réaliser par les rôles des acteurs. Ces contraintes peuvent porter par exemple sur :

- Quelles sont les tâches et fonctions à réaliser (nature de l'action, nombre de tâches et de fonctions),
- Comment réaliser ces tâches et fonctions (méthode à utiliser pour réaliser l'action),
- Quel rôle réalise les tâches et fonctions,
- Les objets, périphériques ou données à manipuler,
- La ou les valeurs des objets, périphériques ou données à manipuler lors des tâches et fonctions,

- Les aspects liés à la temporalité de ces tâches et fonctions à réaliser (quand démarrer la ou les tâches et fonctions, la durée de la ou les tâches et fonctions, quand finir la ou les tâches et fonctions).

Lorsqu'un rôle réalise une action, ce rôle possède l'autorité. Ce rôle possède le contrôle final pour redéfinir l'action ou de terminer de définir l'action. Cette définition se rapporte à la définition de l'autorité de contrôle de (Frank Flemisch et al., 2012).

4.4 Description de l'allocation de l'Autorité sur l'exemple du jeu des 15

Le Tableau 14 décrit l'allocation de l'autorité entre les rôles pour jouer au jeu des 15. Les joueurs ont l'autorité de lancer la partie. Ils partagent l'autorité pour choisir quel joueur lance la partie. L'arbitre a l'autorité d'allouer les rôles *Leader* et *Challenger* aux joueurs. Le *Leader* et le *Challenger* ont l'autorité pour : choisir leur stratégie de jeu, choisir la valeur du jeton sélectionné, mettre fin à leur tour, pour déterminer le résultat de la partie, pour rejouer ou arrêter le jeu des 15, pour définir quel rôle relance la partie. L'arbitre a l'autorité pour désigner le joueur courant (quand un joueur doit jouer et quand il doit attendre), pour définir comment relancer une partie (bouton « rejouer » et clic souris sur le bouton), pour définir quand et comment choisir un jeton (widgets jetons et possibilité d'en sélectionner un par un clic souri sur le widget) et pour définir quand et comment mettre fin au tour.

Tableau 14. Description de l'allocation de l'autorité.

Tâche ou Fonction	Contrainte
Fonction de l'arbitre : « Désactiver les jetons disponibles »	Empêcher la sélection d'un jeton
Fonction de l'arbitre : « Activer les jetons disponibles »	Autoriser la sélection d'un jeton
Fonction de l'arbitre : Initialiser les jetons disponibles	Presser le widget pour sélectionner la valeur du jeton sélectionné
Tâche motrice du Leader : « Sélectionner le jeton »	Effectuer la sélection
Tâche motrice du Challenger : « Sélectionner le jeton »	Effectuer la sélection
Tâche de décision du Leader : « Choisir un jeton »	Valeur du jeton à sélectionner
Tâche de décision du Challenger : « Choisir un jeton »	Valeur du jeton à sélectionner
Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Empêcher la pression du bouton fin de tour
Fonction de l'arbitre : « Activer le bouton fin de tour »	Autoriser la pression du bouton fin de tour
Fonction de l'arbitre : « Initialiser le bouton fin de tour »	Bouton à presser pour mettre fin au tour
Tâche motrice du Leader : « Mettre fin au tour »	Mettre fin au tour
Tâche motrice du Challenger : « Mettre fin au tour »	Mettre fin au tour
Fonction de l'arbitre : « Initialiser le joueur courant »	Définir valeur premier joueur
Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Challenger
Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Leader
Tâche cognitive de décision du Leader : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Tâche cognitive de décision du Challenger : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Initialiser le bouton rejouer	Presser le bouton pour rejouer

Activer le bouton rejouer	Autoriser pression bouton rejouer
Tâche motrice du Leader : « Rejouer »	Lancer une nouvelle partie
Tâche motrice du Challenger : « Rejouer »	Lancer une nouvelle partie
Tâche cognitive de décision du Leader : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche cognitive de décision du Challenger : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche de groupe du Leader et du Challenger : « Les joueurs se mettent d'accord pour rejouer »	Défini qui presse le bouton pour rejouer
Tâche cognitive d'analyse du Leader : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Tâche cognitive d'analyse du Challenger : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Fonction de l'Arbitre : « Initialiser le bouton lancer partie »	Presser le bouton lancer partie pour lancer la partie
Fonction de l'Arbitre : « Activer le bouton lancer partie »	Activer bouton lancer partie
Tâche motrice du Joueur : « Lancer la partie »	Lancer la partie
Tâche motrice parler du joueur : « Annoncer la partie du jeu des 15 »	Définir quel joueur lance la partie
Fonction de l'arbitre : « Allouer les rôles aléatoirement aux joueurs »	Définir la valeur du rôle des joueurs

5 Transitions de contrôle et de l'initiative

Dans cette section, nous définissons la description de l'allocation des transitions de contrôle et de l'initiative. Les approches existantes traitant de l'allocation des transitions de contrôle et de l'initiative sont présentées au Chapitre 2.4.1.4. Nous débutons définissant les termes « transition de contrôle » et « initiative » du cadre conceptuel RCRAFT. Puis, nous présentons notre définition de la description de l'allocation des transitions de contrôle et de l'initiative. Ensuite, nous détaillons les aspects temporels et conditionnels des transitions de contrôle et de l'initiative. Puis, nous décrivons l'allocation des transitions de contrôle et de l'initiative sur l'exemple du jeu des 15.

5.1 Définition des transitions de contrôles et de l'initiative

Nous proposons de définir les transitions de contrôle du cadre conceptuel RCRAFT de la façon suivante en précisant la nature du changement d'état défini par (Frank Flemisch et al., 2012) ou (Lu et al., 2016) :

> *On appelle transition de contrôle la modification de l'allocation de l'autorité entre les rôles sur une tâche ou une fonction.*

Nous considérons, dans le cadre conceptuel RCRAFT, les tâches et fonctions à initiative qui initient des coopérations. Nous présentons les définitions des tâches et fonctions coopératives au Chapitre 5.2. Nous proposons de définir l'initiative du cadre conceptuel RCRAFT de la façon suivante :

On appelle initiative la réalisation par les rôles de la ou des premières tâches ou fonctions d'une coopération.

5.2 Définition de l'allocation des Transitions de contrôle et de l'initiative

Définition de la description de l'allocation des transitions de contrôle à partir des travaux présentés au Chapitre 2.4.1.4 dont nous précisons le type de tâche et de fonction concernées :

On appelle description de l'allocation des transitions de contrôle comme étant l'identification des tâches et des fonctions coopératives des rôles modifiant l'allocation de l'autorité entre les rôles sur une tâche ou une fonction.

Définition du *handover* et du *takeover* à partir des travaux présentés au Chapitre 2.4.1.4,

On appelle handover un type de transition de contrôle pour laquelle un des acteurs laisse l'autorité sur une tâche ou une fonction à un ou plusieurs autres rôles.

On appelle takeover un type de transition de contrôle pour laquelle un des rôles prend l'autorité sur une tâche ou une fonction à un ou plusieurs autres rôles.

Définition de l'allocation de l'initiative permettant d'identifier quel est le rôle initiant une transition de contrôle (voir Chapitre 2.4.1.4) :

On appelle description de l'allocation de l'initiative comme étant l'identification de la ou des premières tâches ou fonctions d'une coopération des rôles.

5.3 Aspects temporels et les conditions des transitions de contrôle et de l'initiative

Nous présentons dans cette section les aspects temporels et les conditionnels des transitions de contrôle et de l'initiative. Nous nous référons aux opérateurs temporels définis au Chapitre 3.3.2.1.

5.3.1 Séquence ou interruption

Les transitions de contrôle peuvent être initiées pour gérer une modification dans le contexte nécessitant d'interrompre les tâches et fonctions courantes. Les transitions de contrôle peuvent être décrites par une interruption de tâches ou de fonctions dans la description de l'allocation des fonctions et des tâches entre les rôles des acteurs. Également, les transitions de contrôle peuvent être initiées pour gérer le passage d'un sous-but à un autre sans que cette transition n'interrompe le premier sous-but. Dans ce cas, la transition de contrôle sera traduite par une séquence dans la description de l'allocation des fonctions et des tâches entre les rôles des acteurs. Il est à noter que, lorsqu'une transition de contrôle est réalisée par les acteurs, un autre sous-but réalisé en concurrence peut ne subir aucune modification.

5.3.2 Contraintes temporelles des transitions de contrôle

Les transitions de contrôle peuvent être soumises à des contraintes temporelles ou conditionnée par le résultat d'une tâche ou fonction. Une contrainte temporelle peut définir :

- Une durée avant le déclenchement d'une transition de contrôle :
    - Par exemple, un système peut afficher un décompte au bout duquel il appliquera de nouveaux paramètres sauf si l'utilisateur annule les nouveaux paramètres avant la fin du décompte. Cette contrainte temporelle décrit une durée avant le déclenchement d'une transition de contrôle : l'annulation ou l'application des nouveaux paramètres.
- Une durée après le déclenchement d'une transition de contrôle :
    - Par exemple, une voiture peut rendre le contrôle de longitudinal (vitesse) et surveiller la vitesse de la voiture quelques secondes avant de rendre totalement le contrôle à l'utilisateur. La transition de contrôle est initiée et l'utilisateur a réalisé une tâche coopérative pour gérer la vitesse de la voiture. Cependant, le système exécute une fonction coopérative pour surveiller la vitesse du véhicule avant de mettre fin à la transition de contrôle.
- Une durée des tâches et fonctions coopératives réalisées lors de la transition de contrôle.
    - Par exemple, pour déverrouiller une porte, deux acteurs peuvent avoir à laisser leur doigt sur un capteur pendant 5 secondes. La transition de contrôle requiert que les tâches coopératives aient une durée de 5 secondes.

Cette contrainte temporelle est définie par une tâche ou une fonction à autorité de l'un des rôles.

Les transitions de contrôle peuvent être conditionnée par le résultat d'une tâche ou fonction. Par exemple, si un rôle attend la validation d'un code, le rôle qui réalise la validation du code initiera une transition de contrôle en fonction de la validité du code. La transition de contrôle est conditionnée par la valeur du code.

5.3.3 Transitions de contrôle optionnelles ou nécessaires

Les transitions de contrôle peuvent également être optionnelle ou obligatoire. Si la transition de contrôle est optionnelle, sa réalisation ou non n'a pas d'impact dans l'atteinte du but décrit. Si la transition de contrôle est obligatoire, sa non-réalisation compromet l'atteinte du but décrit et résulte d'une erreur d'un des acteurs.

5.4 Description de l'allocation des Transitions de contrôle et de l'initiative sur l'exemple du jeu des 15

Le Tableau 15 décrit l'allocation des transitions de contrôle et de l'initiative pour le jeu des 15. Par exemple, l'arbitre affiche aux joueurs le bouton pour lancer la partie. L'arbitre relâche le contrôle sur l'élément de la couche interface représentant le bouton pour lancer la partie du jeu des 15 (*handover*). Les acteurs ayant le rôle de Joueur reçoivent le contrôle et peuvent utiliser ce bouton. Lors du tour du *Leader*, après avoir choisi un jeton, le *Leader* le sélectionne. Le *Leader* relâche le contrôle sur l'élément de la couche interface représentant les « widgets des jetons disponibles » (*handover*). L'arbitre désactive ensuite les « widgets des jetons disponibles » et empêche les joueurs d'utiliser cet élément de la couche interface. L'arbitre prend le contrôle sur les « widgets des jetons disponibles » (*takeover*).

Tableau 15. Distribution des Transitions de Contrôle et de l'initiative

DOD	Rôle Initiateur	Rôle recevant le contrôle
Couche Interface « Bouton lancer partie »	Arbitre : « Afficher le bouton lancer la partie » **HANDOVER**	Joueur : « Voir le bouton lancer la partie »
Couche Interface « Bouton lancer partie » et Périphérique d'entrée « Souris »	Joueur : « Lancer la partie » **HANDOVER**	Arbitre : « Recevoir la pression bouton »
Couche Interface « Widgets des jetons disponibles »	Arbitre : « Afficher les jetons disponibles » **HANDOVER**	Leader : « Voir les jetons disponibles »
Couche Interface « Bouton Rejouer »	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Leader : « Voir le bouton rejouer »
	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Challenger : « Voir le bouton rejouer »
Couche Interface « Widgets des jetons disponibles » et Périphérique d'entrée « Souris »	Leader : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »
	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Arbitre
Couche interface « Bouton fin de tour »	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Leader : « Voir le bouton fin de tour activé »
Couche interface « Bouton fin de tour » et Périphérique d'entrée « Souris »	Leader : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »
	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Arbitre
Couche Interface « Widgets des jetons disponibles »	Arbitre : « Afficher les jetons disponibles actifs » **HANDOVER**	Challenger : « Voir les jetons disponibles »
Couche Interface « Widgets des jetons disponibles » et Périphérique d'entrée « Souris »	Challenger : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »
	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Arbitre
Couche interface « Bouton fin de tour »	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Challenger : « Voir le bouton fin de tour activé »
Couche interface « Bouton fin de tour » et Périphérique d'entrée « Souris »	Challenger : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »
	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Arbitre
Couche Interface « Widgets des jetons disponibles »	Arbitre : « Afficher les jetons disponibles actifs » **HANDOVER**	Leader : « Voir les jetons disponibles »
Couche Interface « Bouton rejouer »	Leader : « Rejouer »	Arbitre : « Recevoir rejouer »
	Challenger : « Rejouer »	Arbitre : « Recevoir rejouer »

6 Responsabilité

Dans cette section, nous définissons la description de l'allocation de la responsabilité. Les approches existantes traitant de l'allocation de la responsabilité sont présentées au Chapitre 2.4.1.3. Nous débutons définissant le terme « responsabilité » du cadre conceptuel RCRAFT. Puis, nous présentons la définition de la description de l'allocation de la responsabilité. Ensuite, nous explicitons ces caractéristiques et nous décrivons l'allocation de la responsabilité sur l'exemple du jeu des 15.

6.1 Définition de la Responsabilité

Nous proposons de définir la responsabilité du cadre conceptuel RCRAFT de la façon suivante à partir des travaux présentés au Chapitre 2.4.1.3 :

> *On appelle responsabilité le fait qu'un rôle puisse causer une déviation du résultat attendu.*
>
> *On appelle résultat attendu l'état final du système, de l'environnement ou des objets dans lequel le but utilisateur est atteint.*

Nous précisons que nous identifions les rôles des acteurs pouvant faire dévier le résultat attendu pour l'atteinte du but.

6.2 Définition de la description de l'allocation de la Responsabilité

Définition de la description de l'allocation de la responsabilité :

> *On appelle description de l'allocation de la responsabilité l'identification des tâches et fonctions des rôles pouvant causer une déviation du résultat attendu.*

De la même manière que les travaux présentés au Chapitre 2.4.1.3, les tâches des rôles pour lesquels une responsabilité est allouée peuvent être les causes de la non-atteinte d'un but désiré par un ou plusieurs rôles.

6.3 Description de l'allocation de la Responsabilité en détail

Le ou les résultats attendus sont définis par rapport aux buts des différents rôles des acteurs. Lorsque le résultat attendu est atteint, le but est lui aussi atteint. Une façon de vérifier l'atteinte du but d'un rôle est de comparer le résultat effectif avec le résultat attendu. Si le résultat effectif et différent du résultat attendu, alors une ou plusieurs tâches ou fonctions ont causées cette déviation. Ces tâches ou fonctions sont notées comme étant à responsabilité. Ces tâches ou fonctions modifient, produisent ou suppriment des données, périphériques ou objets (DOD) dont les valeurs définissent le résultat attendu. Si une erreur sur la production ou la modification de ces DODs se produit, alors le résultat attendu ne sera pas atteint.

Les tâches ou fonctions notées comme étant à responsabilité ne traduisent pas forcément la culpabilité de l'acteur qui les réalise. En effet, la culpabilité est définie par des règles ou des lois des organisations qui régissent les tâches et les fonctions des acteurs. Les tâches ou fonctions à responsabilité indiquent seulement quelles tâches ou fonctions ont une incidence directe sur l'atteinte du résultat attendu.

6.4 Description de l'allocation de la Responsabilité sur l'exemple du jeu des 15

Le Tableau 16 décrit l'allocation de la responsabilité entre les rôles des acteurs du jeu des 15. Les rôles *Leader* et *Challenger* jouent pour gagner la partie. Le résultat attendu associé à ce but est de posséder 3 jetons dont la somme est égale à 15. Les DODs dont les valeurs définissent ce résultat attendu sont ceux manipulés pour la sélection des jetons. En conséquence, les tâches et fonctions à responsabilité pour le résultat attendu du *Leader* et du *Challenger* concernent la sélection des jetons. Le rôle arbitre soutient le but des joueurs en faisant respecter les règles du jeu des 15. Les règles du jeu sont :

- Le *Leader* commence à jouer
- Deux joueurs sont requis
- Le joueur courant ne choisit qu'un seul jeton par tour
- Un jeton ne peut être choisit qu'une fois
- Les joueurs jouent chacun leur tour

Les résultats attendus de l'arbitre sont que chacune de ces règles soient respectées. En conséquence, les DOD dont les valeurs définissent ces résultats attendus sont ceux relatifs à l'allocation des rôles, à la gestion des tours de jeu et à la gestion des jetons.

Tableau 16. Distribution de la responsabilité pour le jeu des 15 Classique #1 et le jeu des 15 Annulation Sélection #5.

DOD	Tâche ou Fonction	Résultat attendu possiblement impacté
Couche Interface « Widgets des jetons disponibles »	Tâche motrice Leader : « Sélectionner un jeton »	Leader : « Posséder 3 jetons dont la somme est égale à 15 »
	Tâche motrice Challenger : « Sélectionner un jeton »	Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
	Fonction de l'arbitre : « Initialiser les jetons disponibles »	Arbitre : « Un seul jeton est sélection chaque tour »
	Fonction de sortie de l'arbitre : « Afficher les jetons disponibles activés »	
	Fonction de l'arbitre : « Désactiver les jetons disponibles »	
	Fonction de sortie de l'arbitre : « Retirer le jeton sélectionné des jetons disponibles et afficher les jetons disponibles désactivés »	
Objet Logiciel « Jeton sélectionné par le Leader »	Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »

Objet Logiciel « Jeton sélectionné par le Challenger »	Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Couche Interface « Widgets des jetons joués par le Leader »	Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Leader »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Couche Interface « Widgets des jetons joués par le Challenger »	Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Challenger »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Objet Logiciel « Jetons du Challenger »	Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Couche Interface : « Bouton fin de tour »	Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Arbitre : « Les joueurs jouent chacun leur tours »
	Fonction de sortie de l'arbitre : « Afficher le bouton fin de tour désactivé »	
	Fonction de l'arbitre : « Activer le bouton fin de tour »	
	Fonction de sortie : « Afficher le bouton fin de tour disponible »	
	Tâche motrice du Leader : « Mettre fin au tour »	
	Tâche motrice du Challenger : « Mettre fin au tour »	
Couche Interface « Texte d'indication des tours des joueurs »	Fonction de sortie de l'arbitre : « Afficher le joueur courant »	
Objet Logiciel : « Joueur courant »	Fonction de l'arbitre : « Initialiser le joueur courant »	
	Fonction de l'arbitre : « Changer le joueur courant »	
Objet Logiciel « Rôles des joueurs »	Fonction de l'arbitre : « Allouer les rôles aléatoirement »	Arbitre : « Le jeu est joué avec 2 joueurs »
Couche Interface : Texte décrivant l'allocation des rôles aux joueurs	Fonction de sortie de l'arbitre : « Afficher les rôles des joueurs »	

7 Conclusion

Dans ce chapitre, nous avons présenté le cadre conceptuel RCRAFT décrivant l'allocation : des **r**essources, des transitions de **c**ontrôle et de l'initiative, de la **r**esponsabilité et de l'**a**utorité par rapport à l'allocation des **f**onctions et des **t**âches entre les rôles des acteurs. Chacun de ces éléments a été illustré sur l'exemple du jeu des 15. Les définitions proposées pour ce cadre conceptuel peuvent être adaptées au domaine d'application. Ce cadre conceptuel propose de définir chacun des concepts présentés pour leur analyse et leur description au niveau des tâches et des fonctions des rôles.

Ce cadre conceptuel nous permet de décrire explicitement au niveau des tâches et des fonctions les éléments requis pour soutenir l'analyse d'un ou plusieurs systèmes contenant de l'automatisation et des tâches de leurs utilisateurs. Ce cadre conceptuel est le résultat d'un travail itératif. Nous avons d'abord proposé de décrire les allocations de la responsabilité et de l'autorité par rapport à l'allocation des fonctions et des tâches (Elodie Bouzekri, Canny, Martinie, et al., 2019b). Nous avons montré qu'une telle description permettait de lever les ambiguïtés des niveaux d'automatisation de (R. Parasuraman et al., 2000). En effet, plusieurs solutions de conception d'un même niveau d'automatisation sont possibles pour un même niveau d'automatisation (Elodie Bouzekri, Canny, Martinie, et al., 2019b). Nous avons ensuite proposé d'ajouter les descriptions des allocations des ressources, des transitions de contrôle et de l'initiative par rapport aux allocations de fonctions et de tâches (Elodie Bouzekri et al., 2020). Ces travaux présentent une analyse RCRAFT des niveaux d'automatisation de la SAE (SAE J3016, 2019) pour la tâche de conduite démontrant que : l'allocation des ressources est peu décrite, les transitions de contrôles sont décrites partiellement et le conducteur a la même responsabilité alors que moins de ressources sont partagées avec lui aux niveaux 3 et 4.

Afin de pouvoir documenter l'évolution de l'allocation RCRAFT au fur et à mesure du cycle de conception et de développement des systèmes contenant de l'automatisation et pour permettre une analyse RCRAFT des systèmes contenant de l'automatisation déployés, nous proposons d'intégrer les éléments RCRAFT aux modèles de tâches. Également, l'analyse de l'allocation des ressources, des transitions de **c**ontrôle et de l'initiative, de la responsabilité et de l'autorité nécessite une description précise des tâches et fonctions coopératives entre les acteurs. En effet, ce sont ces tâches et fonctions qui peuvent décrire un ordre, un échange de ressource, l'interruption des tâches d'un autre rôle, etc. Dans le chapitre suivant, nous présentons une notation permettant la représentation explicite de chacun des concepts RCRAFT dans les modèles de tâches.

Chapitre 5. Notation pour la représentation des éléments RCRAFT

Ce chapitre présente un principe de modélisation des fonctions et des tâches, puis les éléments de notation pour la représentation explicite de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité grâce à l'outil HAMSTERS. Nous proposons un principe de modélisation sous la forme de modèles de tâches, différents de ceux présentés au Chapitre 3.2.3, afin de pouvoir réaliser notre objectif d'analyse des systèmes contenant de l'automatisation et des tâches de leurs utilisateurs.

La première section défini les modèles de tâches intégrés et les modèles de tâches ségrégués et les modèles de fonctions. Puis, nous explicitons l'intérêt de représenter l'allocation des tâches et des fonctions à travers des modèles de tâches ségrégués et des modèles de fonctions plutôt qu'à travers des modèles de tâches intégrés.

La seconde section présente un principe de description et de modélisation de la coopération entre utilisateurs et systèmes contenant de l'automatisation.

La troisième section présente les éléments de notation pour la description de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité dans les modèles de tâches ségrégués et de fonctions.

La quatrième section conclut ce chapitre.

1 D'une modélisation intégrée à une modélisation ségréguée

Cette section présente le principe de modélisation utilisé pour décrire l'allocation des fonctions et des tâches entre les systèmes et les utilisateurs. Nous utilisons les termes « but utilisateur », « modèle de tâches », « tâche » et « fonction » tel que défini au Chapitre 3.1.1.3.

Définition d'un modèle de tâches intégré :

*On appelle **modèle de tâches intégré** un modèle de tâches contenant dans un même modèle les tâches réalisées par un utilisateur et les fonctions exécutées par le système interactif utilisé.*

- Une tâche dans un modèle de tâches intégré est de type :

- *Perceptif (perceptif abstrait, perceptif),*
- *Cognitif (cognitif abstrait, analyse, décision),*
- *Moteur (moteur abstrait, moteur),*
- *Coopératif (coopératif abstrait, perceptif coopératif, moteur coopératif, de groupe),*

- *Interactif (interactif d'entrée, interactif de sortie, interactif entrée-sortie),*
- *Abstrait*

- Une fonction dans un modèle de tâches intégré est de type :

- *Interactive (interactive d'entrée, interactive de sortie, interactive d'entrée-sortie)*
- *Coopératif (abstrait coopératif, interactive d'entrée coopératif, interactive de sortie coopératif, de groupe)*
- *Traitement*
- *Abstrait*

Définition d'un modèle de tâches ségrégué :

*On appelle **modèle de tâches ségrégué** un modèle de tâches contenant exclusivement les tâches réalisées par un utilisateur. Les fonctions que le système doit exécuter pour que l'utilisateur atteigne son but ne peuvent être représentées dans ce modèle.*

- Une tâche dans un modèle de tâches ségrégué est de type :

- *Perceptif (perceptif abstrait, perceptif),*
- *Cognitif (cognitif abstrait, analyse, décision),*
- *Moteur (moteur abstrait, moteur),*
- *Coopératif (coopératif abstrait, perceptif coopératif, moteur coopératif),*
- *Abstrait*

Définition d'un modèle de fonctions :

*On appelle **modèle de fonctions** un modèle de tâches contenant exclusivement les fonctions exécutées par un système. Les tâches que l'utilisateur doit réaliser pour atteindre son but ne peuvent être représentées dans ce modèle.*

- Une fonction dans un modèle de fonctions est de type :

- *Entrée,*
- *Traitement (traitement abstrait, traitement),*
- *Sortie,*
- *Coopératif (abstrait coopératif, entrée coopératif, sortie coopératif),*
- *Abstrait.*

Afin de décrire l'allocation des fonctions et des tâches entre les systèmes et les utilisateurs, nous proposons une approche de modélisation différente de celle proposée dans la littérature (voir Chapitre 3.2.3). En effet, nous proposons de modéliser les fonctions des acteurs systèmes de la même manière que les tâches des acteurs utilisateurs. C'est-à-dire, nous proposons de modéliser les fonctions systèmes pour soutenir l'atteinte des buts utilisateurs en décrivant chaque rôle assigné à des systèmes. Ainsi, le système peut être décrit en tant qu'acteur avec lequel les utilisateurs coopèrent pour accomplir un but.

Afin d'illustrer ce principe de modélisation, nous utilisons un exemple illustratif simple : retirer de l'argent au distributeur automatique de billets (DAB). Nous illustrons la modélisation de cet exemple avec l'outil HAMSTERS présenté au Chapitre 3.2.4.

1.1.1 L'exemple illustratif sur un distributeur automatique de billets

Le distributeur automatique de billet permet aux utilisateurs de retirer de l'argent grâce à leur carte de crédit. La distribution des billets était, avant l'invention du distributeur de billets, une tâche attribuée au guichetier de banque. L'introduction du distributeur de billet a donc automatisée la tâche de distribution de billets : la tâche a migré d'un humain vers un système.

Nous considérons un distributeur de billets qui fonctionne comme suit :

- Le distributeur automatique de billets propose d'abord à l'utilisateur d'insérer sa carte de crédit.
- Ensuite, le distributeur automatique de billets demande à l'utilisateur d'insérer le code PIN associé à sa carte de crédit en maximum trois tentatives.
    - Si l'utilisateur échoue, le distributeur de billets conserve la carte de crédit de l'utilisateur et met fin à la tâche de l'utilisateur.
    - Si l'utilisateur réussi, le distributeur automatique de billets lui présente les montants disponibles.
- Lorsque l'utilisateur a sélectionné le montant désiré, le distributeur vérifie la possibilité du retrait sur le compte bancaire de l'utilisateur.
    - Si le retrait est refusé, le distributeur affiche le message de refus et rend la carte bancaire à l'utilisateur.
    - Si le retrait est autorisé, le distributeur automatique de billets rend la carte de crédit à l'utilisateur. Ensuite, le distributeur donne l'argent demandé et le ticket de retrait à l'utilisateur.

Figure 49. Un distributeur automatique de billet.

Nous utilisons cet exemple pour illustrer les principes de modélisation intégrée et ségréguée présentés ci-après.

1.1.2 La modélisation intégrée du comportement de l'utilisateur avec le système interactif

Les notations pour la modélisation de tâches (présentées au Chapitre 3.2.3) comme CTT (Paterno et al., 1997) ou HAMSTERS (Martinie et al., 2011) proposent de décrire les tâches des utilisateurs et les interactions avec le ou les systèmes interactifs, pour accomplir leurs buts selon leurs rôles. Les tâches des différents rôles des utilisateurs sont décrites dans différents modèles de tâches intégrés. Certaines fonctions systèmes utiles pour comprendre un affichage par exemple sont également décrites.

Nous appelons une telle modélisation une modélisation intégrée des tâches utilisateurs et des fonctions systèmes. Une telle modélisation intégrée décrit, dans un même modèle, les tâches d'un rôle des utilisateurs et les fonctions du système interactif utilisé pour atteindre le but du rôle utilisateur. De cette manière, l'ordonnancement des tâches utilisateurs, des fonctions systèmes et les interactions sont décrites de manière explicite et sans ambiguïté. La Figure 50 est un exemple de modèle de tâches intégré. Les tâches utilisateurs et les fonctions systèmes sont décrites dans un même modèle de tâches intégré décrivant le but de l'utilisateur client : « Retirer de l'argent ».

Lorsque l'utilisateur retire de l'argent au distributeur automatique de billets (DAB), l'utilisateur a le rôle de client. Le modèle de tâches décrivant les tâches utilisateurs permettant d'accomplir le but « Retirer de l'argent » est présenté dans le modèle de tâches HAMSTERS de la Figure 50. La séquence des tâches permettant d'accomplir ce but sont les suivantes :

- Le client insère sa carte de crédit dans le DAB (tâche abstraite : insérer la carte) : Le client perçoit le DAB, prend sa carte dans son portefeuille et insère sa carte.
- Ensuite, le client soit s'identifier avec son code PIN (tâche abstraite : identifier avec PIN) : Le DAB affiche la demande de code PIN. Le client perçoit la demande de code PIN et entre chaque chiffre de son code PIN à l'aide du pavé numérique du DAB (tâche

abstraite itérative : entrer le code PIN). Le client interrompt la saisie du code PIN en le validant lorsqu'il a terminé (tâche abstraite : valider code PIN).

- Le client doit sélectionner un montant (tâche abstraite : Sélectionner le montant) : Après que le DAB ait vérifié le code, le DAB affiche les montants disponibles. Le client voit ces montants, se souvient du montant désiré et sélectionne ce montant avec le bouton du DAB correspondant.
- Le client doit attendre pendant le traitement de sa requête (tâche abstraite : Attendre le traitement)
- Ensuite, le DAB rend la carte de crédit et fournit les billets et le ticket de retrait (tâche abstraite : Finaliser de retirer de l'argent) :
    - Le client doit récupérer sa carte (tâche abstraite : Récupérer la carte) : L'affichage demandant de récupérer la carte interrompt l'attente du client. Le client perçoit cet affichage ainsi que la carte et la récupère.
    - Le DAB fournit les billets et le ticket de retrait (tâche abstraite : Fournir ticket et billets) : le DAB affiche « Récupérer vos billets » et donne les billets, le DAB affiche « Prendre votre ticket » et donne le ticket.
    - Le client peut récupérer les billets et le ticket dans l'ordre qu'il souhaite (tâches abstraites : « Récupérer les billets » et « Récupérer le ticket » avec l'opérateur ordre indépendant) : Le client perçoit la demande de récupérer les billets et les billets et les récupère. Le client perçoit la demande de récupérer le ticket et le ticket et le récupère.

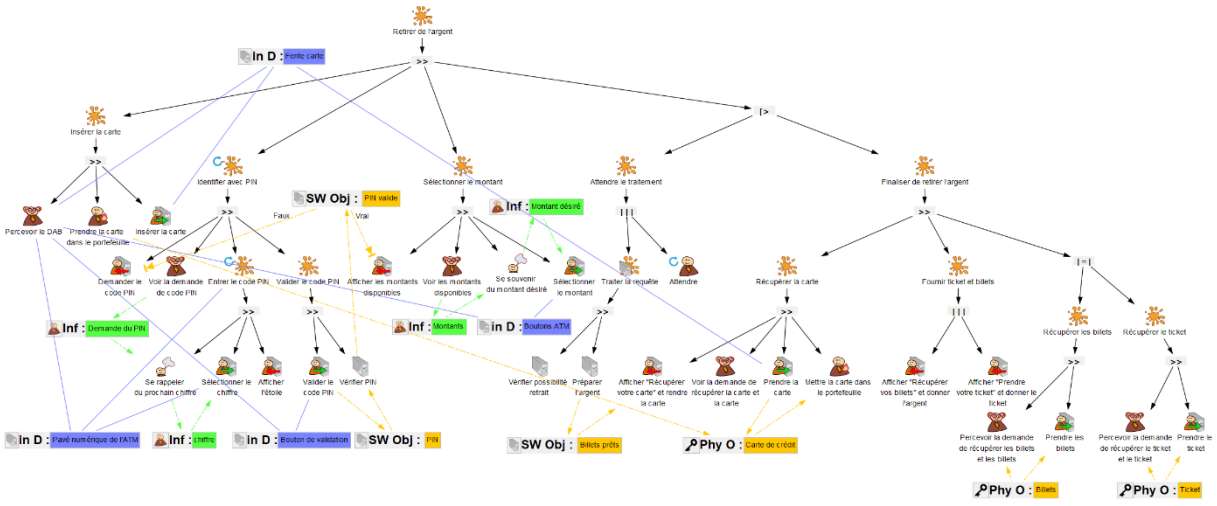


Figure 50. Modèle de tâches intégré du client pour retirer de l'argent au distributeur de billets

Dans une telle modélisation, le système est un outil dont l'utilisateur se sert pour accomplir un but. Dans l'exemple du distributeur de billet, le distributeur est le système utilisé par le client pour retirer des billets. Le distributeur de billets n'est pas un acteur avec lequel le client coopère pour retirer des billets.

1.1.3 La modélisation ségréguée du comportement de l'utilisateur avec le comportement du système

Si une modélisation intégrée des tâches et des fonctions systèmes possède l'avantage de fournir une vision de l'ensemble des tâches d'un rôle utilisateur et des fonctions systèmes et de leurs enchainements, les fonctions systèmes sont dupliquées pour chacun des rôles utilisateurs décrits. Cette duplication devient problématique dans la mesure où nous avons comme objectif d'analyser le comportement d'un système contenant de l'automatisation et que, dans cet objectif, nous allons décrire non plus seulement les fonctions système permettant l'interaction, mais aussi les fonctions système dont l'utilisateur peut ne pas avoir connaissance. Également, nous considérons le système comme un acteur soutenant le but des rôles des acteurs utilisateurs dans l'approche proposée. En conséquence, le système peut endosser plusieurs rôles pour soutenir les utilisateurs dans l'atteinte de leurs buts. Ainsi, à des fins d'analyse, nous pouvons modéliser le comportement du système selon ses différents rôles pour soutenir les buts des utilisateurs. Aussi, la modélisation au même niveau d'abstraction des tâches des utilisateurs et des fonctions du système permet la comparaison du travail alloué aux acteurs et permet de changer l'allocation des tâches et des fonctions d'un rôle d'un acteur à un autre au besoin.

C'est pourquoi nous proposons d'utiliser un paradigme de modélisation où le système contenant de l'automatisation est un acteur au même titre que les acteurs utilisateurs. C'est-à-dire, les fonctions des acteurs systèmes sont modélisées dans des modèles de fonctions des rôles de ces acteurs. Ainsi, nous obtenons des modèles de tâches ségrégués décrivant les tâches des rôles des utilisateurs pour accomplir leurs buts. Puis, nous obtenons également des modèles de fonctions décrivant les fonctions des rôles systèmes pour soutenir l'atteinte des buts utilisateurs. La Figure 51 et la Figure 52 sont respectivement : le modèle de tâches ségrégué du rôle client d'un acteur utilisateur et le modèle de fonctions du rôle distributeur d'un acteur système. Nous y retrouvons les tâches et fonctions présentes dans le modèle de tâches intégré de la Figure 50 dans chacun des deux modèles en les décomposant pour les rôles client et distributeur : le modèle de tâches ségrégué ne décrit que les tâches utilisateurs réalisées par le rôle client pour accomplir le but « Retirer de l'argent » et le modèle de fonctions décrit les fonctions système permettant de soutenir le but utilisateur en lui distribuant des billet (racine de l'arbre : « Distribuer des billets »). Le modèle de fonctions détaille les fonctions du système, en particulier les fonctions gérant la disponibilité des widgets de l'interface et des boutons.

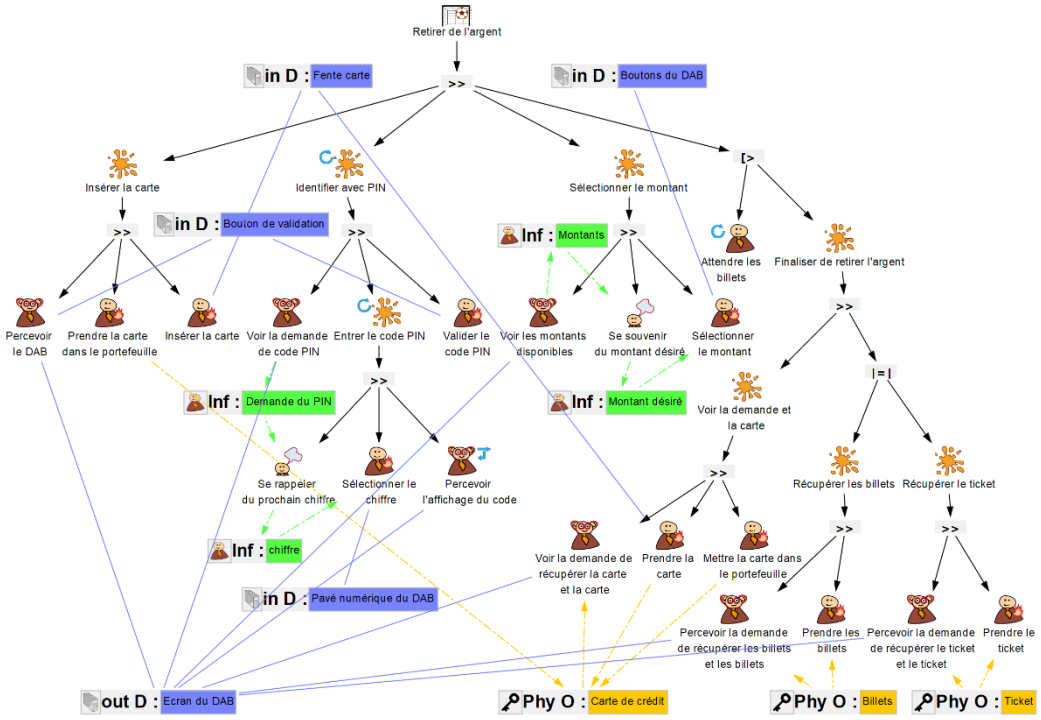


Figure 51. Modèle de tâches ségrégué du client pour retirer de l'argent

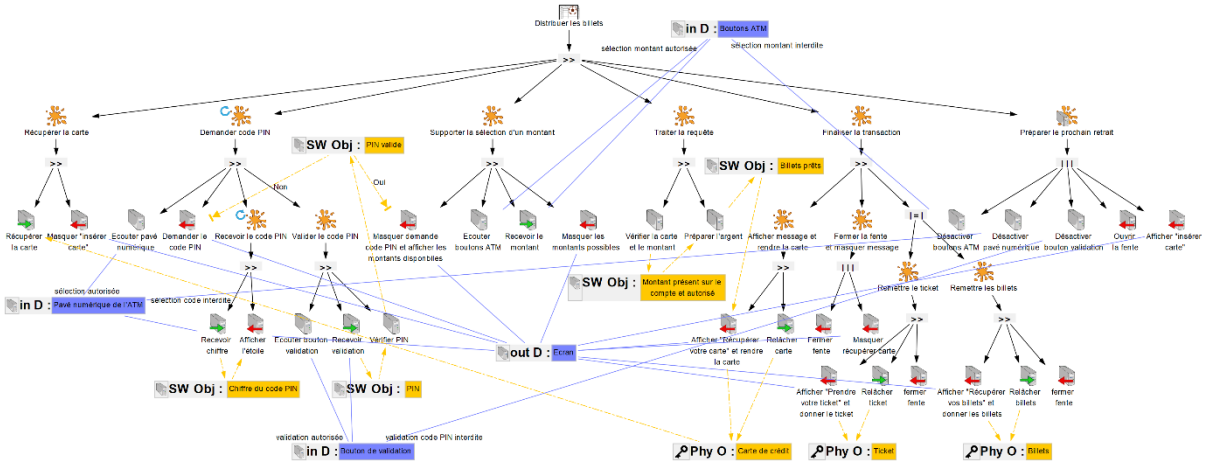


Figure 52. Modèle de fonctions du distributeur pour distribuer des billets.

Cependant, en séparant les modèles de tâches ségrégués et les modèles de fonctions nous introduisons une ambiguïté sur l'ordonnancement entre les fonctions et les tâches décrites dans les différents modèles et sur les ressources échangées. Par exemple sans élément de description supplémentaire, le client peut voir le message demandant le code PIN avant que ce message soit effectivement affiché par le distributeur de billet.

Les éléments de description manquant concernent les tâches et fonctions coopératives entre les acteurs :

- Les ressources partagées entre les rôles acteurs,
- Les conditions de ce partage de ressource dont l'ordonnancement temporel entre les tâches et fonctions coopératives.

C’est pourquoi nous proposons de modéliser la coopération entre les acteurs. C’est-à-dire, la coopération entre les rôles et les différentes instances d’un même rôle. Nous proposons de décrire la coopération avec des tâches et fonctions coopératives et des protocoles de coopération.

2 Coopération entre tâches de modèles de tâches ségrégués et fonctions de modèle de fonctions

Dans cette section, nous introduisons les protocoles de coopération entre tâches et fonctions coopératives afin de modéliser la coopération entre les acteurs tels que définis dans notre approche. Nous illustrons ces éléments de notation dans la notation HAMSTERS.

2.1 Tâches et fonctions coopératives

Au Chapitre 3.1.3, nous avons présenté les définitions d’une tâche coopérative. Ces tâches coopératives sont définies comme des tâches composées de plusieurs sous-tâches réalisées par différents acteurs.

Nous avons choisi une modélisation ségréguée des tâches et des fonctions des rôles des acteurs. Par conséquent, un modèle de tâches ségrégué ou un modèle de fonctions décrit seulement les tâches ou les fonctions d’un rôle des acteurs. Nous choisissons d’appeler tâche ou fonction coopérative une tâche ou fonction réalisée par un utilisateur ou un système qui émet une ressource à d’autres acteurs ou reçoit une ressource de la part d’un ou plusieurs autres acteurs. Ces tâches ou fonctions coopératives sont les tâches et les fonctions qui permettent aux acteurs de se coordonner, de communiquer ou de produire une ou plusieurs ressources ensemble. L’ensemble de ces tâches et fonctions manipulant une ou plusieurs mêmes ressources constitue ce que nous désignons une coopération.

Nous présentons, ci-après, les définitions des tâches et fonctions coopératives que nous utilisons dans cette thèse.

2.1.1 Tâche coopérative

Définition d’une tâche coopérative :

*On appelle **tâche coopérative** réalisée par un utilisateur une tâche coopérative qui :*

- *Nécessite une ressource émise par une autre tâche coopérative, ou*
- *Émet une ressource qui sera nécessaire à une autre tâche coopérative, ou*
- *Nécessite une ressource émise par une fonction coopérative, ou*
- *Émet une ressource qui sera nécessaire à une fonction coopérative.*

2.1.2 Fonction coopérative

Définition d’une fonction coopérative :

*On appelle **fonction coopérative** une fonction coopérative exécutée par un système qui :*

- *Nécessite une ressource émise par une tâche coopérative d'un autre utilisateur, ou*
- *Émet une ressource qui sera nécessaire à la tâche coopérative d'un autre utilisateur, ou*
- *Nécessite une ressource émise par une fonction coopérative d'un système, ou*
- *Émet une ressource qui sera nécessaire à la fonction coopérative d'un système*

2.1.3 Exemple de tâche et de fonction coopérative sur l'exemple du distributeur de billets

Par exemple, lorsque le distributeur de billet demande au client son code PIN, le distributeur de billet exécute une fonction de sortie coopérative (un affichage) afin de communiquer cette demande au client. La ressource émise par le distributeur permet de communiquer la demande de code PIN. Le client réalise une tâche perceptive coopérative lorsqu'il lit cette demande.

Cette définition d'une tâche ou fonction coopérative nous permet de décrire les tâches et fonctions réalisées par un utilisateur ou un système que ce soit dans le cadre d'une tâche coopérative ou collaborative définies par (Grudin, 1994) et (Roschelle & Teasley, 1995). En effet, les conditions de la coopération (les ressources, type, temporalité, cardinalité et localisation de la coopération) sont définies dans le protocole de coopération entre tâches et fonctions coopératives.

2.2 Protocole de coopération

Définition d'un protocole de coopération :

*Un **protocole de coopération** décrit l'émission et la réception d'une ou plusieurs ressources entre plusieurs tâches et fonctions coopératives.*

Un protocole de coopération est composé de 5 attributs :

- *Les ressources émises ou reçues par le biais d'un ou plusieurs DOD pouvant décrire les modalités ou les périphériques utilisés lors de la coopération.*
- *Type de coopération dont la valeur appartient à l'ensemble {Communication, Production, Coordination},*
- *Temporalité de la coopération dont la valeur appartient à l'ensemble {Synchrone, Asynchrone}*
- *Cardinalité dont la valeur décrivant le type de cardinalité appartient à l'ensemble {Broadcast, Unicast, Multicast, Anycast} et dont la valeur indiquant le nombre d'acteur impliqué appartient à l'ensemble {2..n}*
- *Localisation dont la valeur appartient à l'ensemble {Local, Distant}*

Un protocole de coopération possède un identifiant unique permettant de faire référence plusieurs fois à un même protocole de coopération dans différents modèles de tâches.

La définition d'un protocole de coopération nécessite la définition d'une valeur pour le 6 attributs qui le composent.

En accord avec les définitions présentées ci-dessus, nous désignons dans la suite de cette thèse la **coopération** comme étant un ensemble de tâches et de fonctions coopératives émettant ou recevant un même protocole de coopération.

La structure d'un protocole de coopération est présentée dans le Tableau 17. Un protocole de coopération est identifié par un identifiant unique dans les différents modèles de tâches ou de fonctions ou ce protocole de coopération est émis ou reçu.

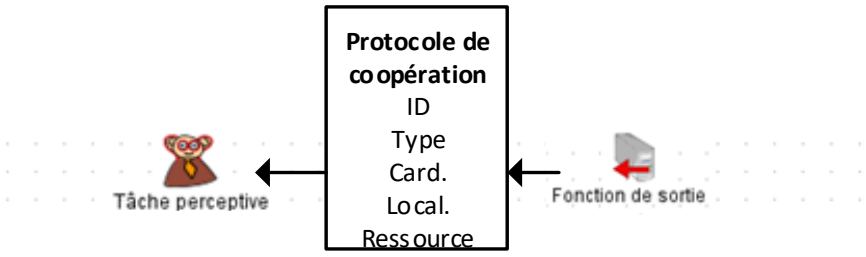
Tableau 17. Structure d'un protocole de coopération

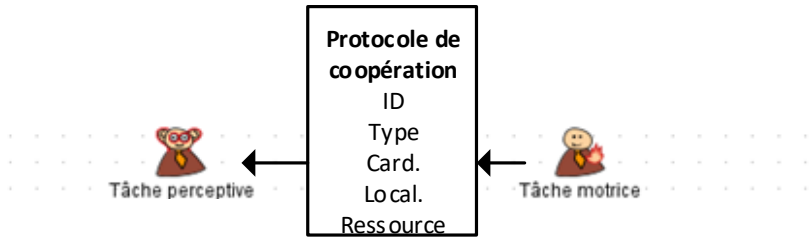
ID	Ressource(s)		Type de coopération	Temporalité	Cardinalité	Localisation
	Valeur	Type				
		[Couche Interface, Parole, Signe, Evènement, …]	Communication, Production, Coordination	[Synchrone, Asynchrone]	[Broadcast, Unicast, Multicast, Anycast] [2..n]	[Local, Distant]

Une tâche ou fonction coopérative recevant un protocole de coopération ne peut pas être réalisée tant que la tâche ou fonction coopérative émettant ce protocole de coopération n'a pas été réalisée. La tâche ou fonction coopérative est conditionnée par la réalisation de ou des tâches ou fonctions coopératives émettant le protocole de coopération.

Le Tableau 18 présente des exemples de tâches et de fonctions coopératives avec la notation HAMSTERS. Une tâche ou fonction coopérative est représentée par le tuple formé par : la tâche ou la fonction, le protocole de coopération et l'arc entre ces 2 éléments (émission ou réception).

Tableau 18. Exemples de tâches et fonctions coopératives avec la notation HAMSTERS

Exemples de Coopération	**Exemple avec la notation HAMSTERS**
Coopération utilisateur-système	

Coopération utilisateur-utilisateur	

2.2.1 Type de coopération du protocole de coopération

De façon similaire à la propriété « Fonctionnalités spécifique aux CSCW (collecticiels) » des *co-interaction* de Penichet et al (Victor M. R. Penichet, Lozano, Gallud, et al., 2009), le protocole de coopération peut être décrit selon les caractéristiques des collecticiels. Nous utilisons les espaces du trèfle du collecticiel : production, communication et coordination (Salber et al., 1995). Cependant, nous ne nous limitons pas à la description de la coopération médiatisée avec un système. Nous considérons également la coopération humain-système. Cela a un impact sur la définition de l'espace de communication. Le système n'est pas toujours seulement le messager de la ressource pour lequel le contenu sémantique échangé lors de la coopération est inconnu. Le système peut être un acteur de la coopération et « connaitre » le contenu sémantique de la ressource reçue ou émise.

- Communication : La communication est l'échange de contenu sémantique entre les acteurs d'une coopération.
- Production : La production est la création, la manipulation et le partage de ressources (texte, livre, plan, etc.).
- Coordination : La coordination est la définition des acteurs, des fonctions et des tâches (leurs relations temporelles), et enfin l'allocation des tâches et des fonctions aux acteurs.

Nous proposons d'indiquer le ou les types de coopération prédominants dans le protocole de coopération. Par exemple, lorsque le client récupère sa carte de crédit, qu'il tire sur la carte pour la reprendre et que le système la relâche, la coopération consiste à manipuler l'objet « carte de crédit » par le biais de la fente à carte du distributeur. Le type de coopération prédominant est « production ».

2.2.2 Protocole de coopération synchrone ou asynchrone

De la même manière que la notation CTT (Paternò, 2004) (Mori et al., 2002) ou la notation proposée par (Victor M. R. Penichet, Lozano, Gallud, et al., 2009) par exemple, le protocole de coopération peut être décrit comme synchrone ou asynchrone. Par exemple, lorsque le client récupère sa carte de crédit, il tire sur la carte pour la reprendre (tâche motrice coopérative « Prendre la carte »). La fonction système du distributeur déclenchée est la fonction d'entrée coopérative « Relâcher la carte ». L'interaction n'est pas possible si le distributeur ne relâche pas la carte lorsque l'utilisateur la prend : la coopération est synchrone. Au contraire, lorsque le distributeur de billet affiche la demande de code PIN (fonction de sortie coopérative : « Demander le code PIN »), le client peut voir la demande de code PIN à n'importe quel moment lorsque l'affichage a eu lieu : la coopération est asynchrone.

2.2.3 Cardinalité du protocole de coopération

De la même manière que la notation proposée dans CTT (Paternò, 2004) (Mori et al., 2002), la cardinalité du protocole de coopération permet de spécifier si la coopération se réalise point à point (entre 2 instances de rôle : unicast) ou avec tous les acteurs (c.-à-d. entre tous rôles ou instances d'un même rôle : broadcast). Nous proposons d'ajouter la possibilité de spécifier si la coopération se réalise avec un groupe restreint d'acteurs (multicast) ou avec n'importe quel acteur (anycast). Le nombre d'acteurs impliqués dans la coopération peut être précisé similairement à la notation CTT (Mori et al., 2002). Par exemple, lorsque le client récupère sa carte de crédit et que le distributeur relâche la carte de crédit, la cardinalité de la coopération est unicast entre 2 instances de rôles : un utilisateur avec le rôle de client et un système avec le rôle de distributeur.

2.2.4 Localisation du protocole de coopération

De la même manière que la notation proposée par (Victor M. R. Penichet, Lozano, Gallud, et al., 2009) par exemple, la localisation spatiale des acteurs de la coopération peut être décrite comme locale ou distante dans le protocole de coopération. Par exemple, lorsque le client récupère sa carte de crédit et que le distributeur relâche la carte de crédit, les deux rôles partagent le même espace : la coopération est locale.

2.2.5 Les ressources reçues ou émises du protocole de coopération

La ressource reçue ou émise lors de la coopération peut être définie de la même manière que les objets de coordination présenté par Sire (Sire, 2000) : un objet qui sert à un moment donné à coordonner les tâches des individus. Cette ressource peut être reçue ou émises sous la forme d'un objet physique (une clef par exemple), un signe (un hochement de tête par exemple), des paroles, un élément de la couche interface du système, etc. Cette ressource peut être reçue ou émise par le biais d'un périphérique d'entrée ou de sortie.

Par exemple, lorsque le client récupère sa carte de crédit et que le distributeur relâche la carte de crédit, la valeur de la ressource émise par le client au distributeur de billet est « le client récupère la carte ». Les ressources émises et reçues lors de la coopération sont : l'objet physique carte de crédit par le biais du périphérique d'entrée fente du distributeur de billet.

3 Eléments de représentation RCRAFT

Dans cette section, nous présentons les éléments de notations soutenant la représentation explicite de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Afin de représenter l'allocation des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité, nous avons utilisé des symboles se démarquant à fois par leur forme et leur luminosité (Bertin, 1973) des autres représentations graphiques utilisées dans la notation HAMSTERS. Le choix de ces symboles de notation devra bénéficier d'une étude avec les utilisateurs.

3.1 Allocation des fonctions et tâches

L'allocation des fonctions et des tâches entre les rôles des acteurs est représentée dans les modèles de tâches ségrégués et de fonctions présentés dans la section 1 de ce chapitre.

L'allocation des fonctions et des tâches inclus l'identification des coopérations entre les rôles des acteurs. La coopération est représentée dans les modèles de tâches ségrégués et de fonctions par les tâches et fonctions coopératives et les protocoles de coopération présentés dans la section 2 de ce chapitre.

3.2 Allocation de ressources

L'allocation des ressources entre les rôles des acteurs est représentée par les DODs présentés au Chapitre 3.3.3 et rappelés dans la Figure 53.

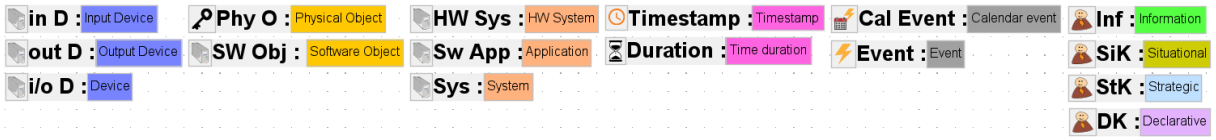


Figure 53. Représentation des données, objets et périphériques (DOD) dans HAMSTERS

La Figure 54 présente des exemples de DODs personnalisés pour la représentation de ressources émises ou reçues lors de coopérations : un élément de la couche interface (DOD : Interface Layer), un mouvement (DOD : Movement) et une parole (DOD : Speech). Dans les exemples illustratifs et l'étude de cas de cette thèse, nous utiliserons les DODs personnalisés représentant un élément de la couche interface (DOD : Interface Layer) et la parole (DOD : Speech). Selon le domaine d'application, de nouveaux DODs personnalisés peuvent être ajoutés afin de couvrir les besoins de modélisation.

Figure 54. Représentation de DODs personnalisés pour la représentation de ressources émises ou reçues lors de la coopération

Une ressource peut être représentée grâce à l'instanciation d'un type de DOD parmi ceux présentés ci-dessus. Par exemple, une ressource peut être affichée sur un élément de la couche interface (DOD : Interface Layer), perçue sous forme d'information (DOD : Information) par les utilisateurs et est traitée sous forme d'objet logiciel (DOD : Software Object) par le système.

3.3 Allocation des transitions de contrôle et de l'initiative

L'allocation des transitions de contrôle et de l'initiative entre les rôles des acteurs est représentée sur les tâches et les fonctions par le symbole présenté en Figure 55.

Figure 55. Symbole représentant une transition de contrôle.

Chaque tâches ou fonctions identifiées comme initiant une transition de contrôle est annotée du symbole comme présenté en Figure 56.

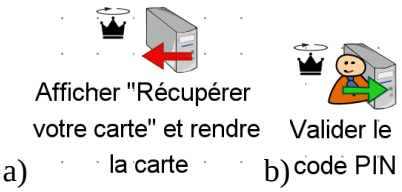


Figure 56. Symbole représentant une transition de contrôle associé à a) une fonction et à b) une tâche

Les opérateurs permettent de décrire si une transition de contrôle cause une interruption d'une tâche ou d'une fonction ou non. Ces opérateurs présentés au Chapitre 3.3.2.1 sont rappelés dans le Tableau 8.

Tableau 19. Opérateurs d'ordonnancement temporel utilisés par la notation HAMSTERS.

Type d'opérateur	Symbole
Enable	>>
Concurrent	|||
Choice	[]
Disable	[>
Suspend-resume	|>
Order Independent	|=|

La Figure 57 présente une partie du modèle de tâches ségrégué du client et du modèle de fonctions du distributeur de billet. La tâche « Attendre les billets » du client est interrompue par l'affichage du « Texte indiquant de récupérer la carte de crédit » exécuté par la fonction « Afficher « Récupérer votre carte » et rendre la carte » du distributeur de billet initiant une transition de contrôle.

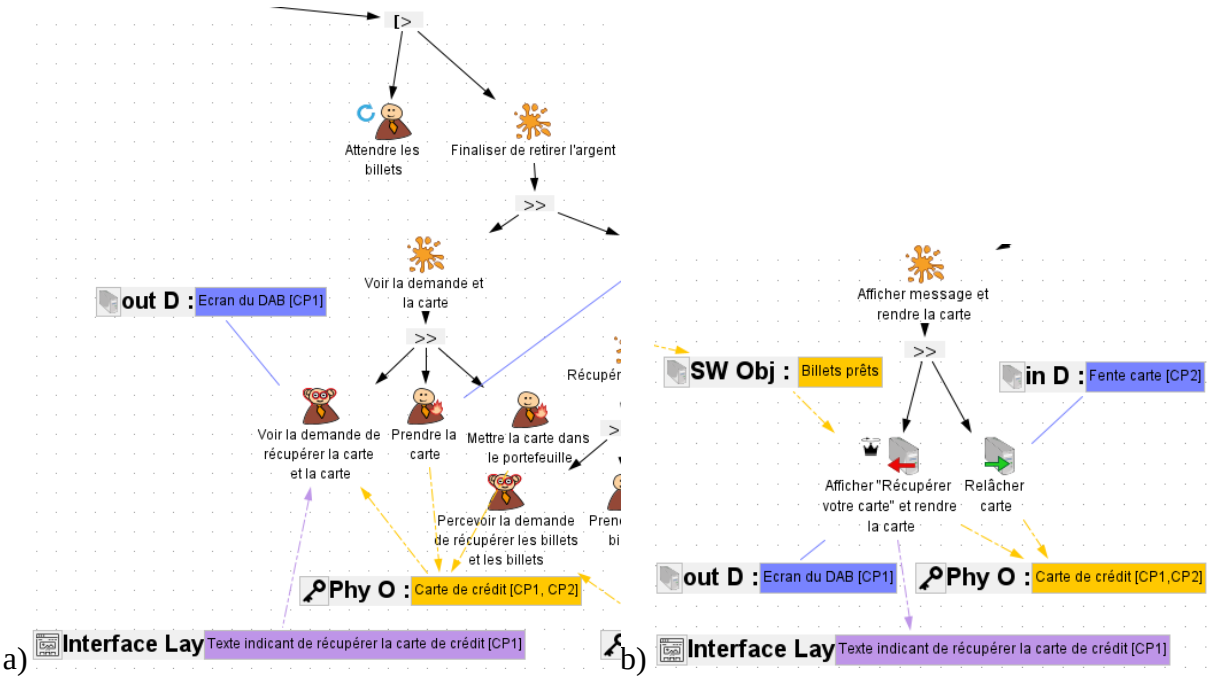


Figure 57. a) Tâche « Attendre les billets » interrompue par l'affichage du « Texte indiquant de récupérer la carte de crédit » exécuté par b) la fonction « Afficher « Récupérer votre carte » et rendre la carte » du distributeur de billet

3.4 Allocation de la responsabilité

L'allocation de la responsabilité entre les rôles des acteurs est représentée sur les tâches et les fonctions par le symbole présenté en Figure 58.

Figure 58. Symbole représentant la responsabilité.

Chaque tâches ou fonctions identifiées comme à responsabilité est annotée du symbole comme présenté en Figure 59.

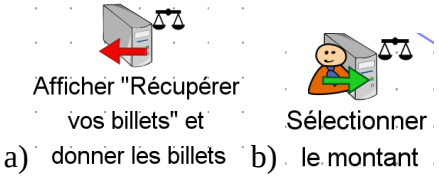


Figure 59. Symbole de la responsabilité associé à a) une fonction et à b) une tâche

3.5 Allocation de l'autorité

L'allocation de l'autorité entre les rôles des acteurs est représentée sur les tâches et les fonctions par le symbole présenté en Figure 60.

Figure 60. Symbole représentant l'autorité.

Chaque tâches ou fonctions identifiées comme à responsabilité est annotée du symbole comme présenté en Figure 61.

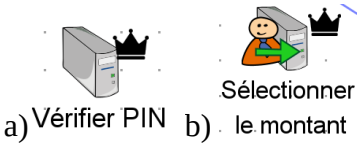


Figure 61. Symbole de l'autorité associé à a) une fonction et à b) une tâche

4 Conclusion

Dans ce chapitre, nous avons décrit un principe de modélisation considérant les systèmes contenant de l'automatisation comme des acteurs au même titre que les utilisateurs. En conséquence, nous avons proposé de décrire les fonctions des rôles des systèmes contenant de l'automatisation avec la même notation et le même niveau d'abstraction que les tâches utilisateurs. Ceci, dans le but d'analyser les tâches et fonctions allouées aux rôles des acteurs en termes d'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Les modèles de fonctions ne constituent pas une modélisation formelle du comportement du système et n'existent qu'à des fins d'analyse.

Dans la seconde section de ce chapitre, nous présentons les protocoles de coopération entre tâches et fonctions coopératives afin de décrire la coopération dans les modèles de tâches ségrégués et de fonctions. Un protocole de coopération est caractérisé par : le type de coopération, les ressources émises et reçues, la cardinalité et la localisation de la coopération.

Dans la troisième section, nous présentons les éléments de représentation graphique de chacun des éléments RCRAFT que nous utilisons dans la suite de cette thèse. Ces éléments de notation permettent de décrire explicitement dans les modèles de tâches ségrégués et de fonctions les éléments RCRAFT suite à l'analyse RCRAFT.

Dans le chapitre suivant, nous présentons un processus permettant de réaliser la modélisation des modèles de tâches ségrégués et des modèles de fonctions ainsi que cette analyse et la description des éléments RCRAFT dans les modèles.

Chapitre 6. Processus RCRAFT d'analyse et de description systématique de RCRAFT

Ce chapitre présente la proposition d'un processus d'analyse et de description systématique des éléments RCRAFT. Ce processus permet de réaliser l'analyse et la description de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité à partir de l'analyse et de la description des tâches et des fonctions.

La première section présente le processus RCRAFT :

1. Dans un premier temps, une vue haut-niveau du processus est présentée.
2. Ensuite, le processus est présenté phase par phase de manière détaillée.

La seconde section présente l'application du processus RCRAFT à l'exemple du Jeu des 15.

La troisième section conclut ce chapitre.

1 Description du processus RCRAFT

Cette section présente le processus RCRAFT que nous proposons pour réaliser l'analyse RCRAFT intégrant : la modélisation des tâches et fonctions des différents rôles explicitant l'allocation de tâches et des fonctions, puis l'analyse et la description de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre les rôles des acteurs.

1.1 Vision haut-niveau du processus RCRAFT

Cette section présente une vue haut-niveau du processus RCRAFT qui a été défini dans le cadre de cette thèse. De la même manière que les processus de conception centrée utilisateur présentés au Chapitre 1.1.3.1, ce processus est itératif. Également, le processus proposé comprend une phase de compréhension et de modélisation du comportement des différents acteurs suivie d'une phase d'identification de l'allocation RCRAFT et, enfin, d'une dernière phase se terminant par une évaluation au regard des objectifs de conception de l'automatisation. Ce processus ne fait pas directement intervenir les utilisateurs mais utilise les connaissances des experts sur le travail des utilisateurs et des méthodes IHM dont, en particulier, l'analyse et la modélisation de tâches présentée au Chapitre 3.2. Nous présentons une vue d'ensemble du processus pour permettre d'en comprendre le cheminement. Puis, nous présentons dans les prochaines sections ce processus en détail.

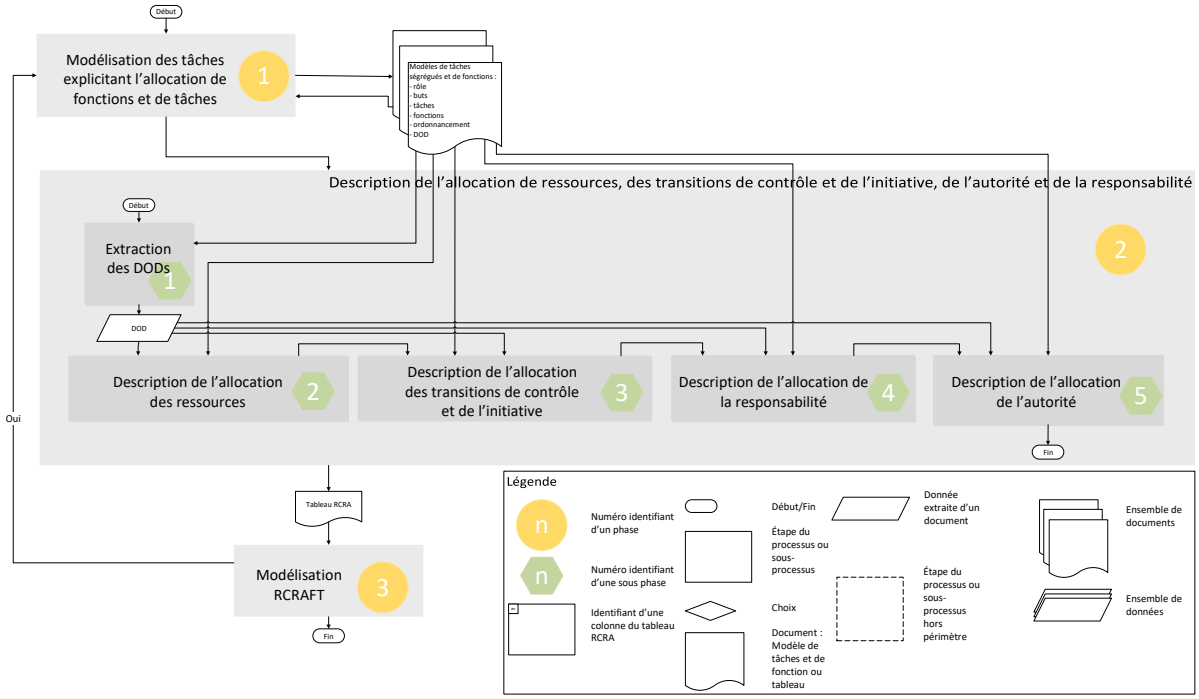


Figure 62. Vue haut niveau du processus RCRAFT

la Figure 62 présente une vue haut-niveau du processus RCRAFT composé de trois phases :

1. Le première phase intitulée « Modélisation des tâches explicitant l'allocation de fonctions et de tâches » décrit les étapes de modélisation des tâches utilisateurs et des fonctions systèmes pour chacun des rôles des acteurs. Cette phase du processus permet de produire les modèles de tâches ségrégués et de fonctions. Ces modèles sont utilisés dans les deux prochaines phases pour analyser un ou des systèmes contenant de l'automatisation et les tâches de leurs utilisateurs. Les résultats de cette analyse sont d'abord décrits dans le tableau RCRA présenté dans le Tableau 20 lors de la phase 2, puis décrits dans les modèles de tâches ségrégués et de fonctions lors de la phase 3.
2. La phase « Description de l'allocation de ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité » décrit les cinq sous-phases nécessaires pour décrire l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité dans le tableau RCRA grâce à l'analyse des modèles de tâches ségrégués et de fonctions produits pendant la phase 1. Cette phase du processus permet de produire le tableau RCRA présenté dans le Tableau 20.
    1. La première sous-phase permet d'extraire tous les DODs des modèles de tâches ségrégués et de fonctions. Ensuite, les sous-phases 2, 3, 4 et 5 sont appliquées à chacun des DODs un par un.
    2. La sous-phase « Description de l'allocation des ressources » permet d'analyser quelle(s) ressource(s) est associée au DOD. Puis, quel rôle est le possesseur du DOD. Et enfin, si le DOD est une instance d'une ressource (modalité ou périphérique) d'un protocole de coopération, quels rôles sont les acquéreurs du DOD.

3. La sous-phase « Description de l'allocation des transitions de contrôle et de l'initiative » permet d'analyser à partir des modèles de tâches ségrégués et de fonctions quelles sont les tâches et fonctions coopératives qui émettent ou reçoivent le DOD, si ce DOD est une instance d'une ressource (modalité ou périphérique) d'un protocole de coopération. La sous-phase permet d'analyser quel rôle initie cette coopération et si cette coopération permet une transition de contrôle.
4. La sous-phase « Description de l'allocation de la responsabilité » permet d'analyser si une modification du DOD peut conduire à une déviation du résultat attendu de l'un des rôles. Les tâches et fonctions pouvant modifier la valeur du DOD sont identifiées comme à responsabilité.
5. La sous-phase « Description de l'allocation de l'autorité » permet d'analyser si les tâches ou fonctions manipulant le DOD sont à autorité. C'est-à-dire, si ces tâches ou fonctions définissent des contraintes sur d'autres tâches ou fonctions par le biais du DOD qu'elles manipulent.

Tant que tous les DODs ne sont pas traités la phase 2 itère sur l'extraction d'un DOD.

3. La dernière phase du processus « Modélisation RCRAFT » permet d'ajouter les éléments de notation RCRA dans les modèles de tâches ségrégués et de fonctions. Si un problème de modélisation a été repéré lors de la phase précédente, une itération peut être effectuée pour modifier la description des tâches et des fonctions en conséquence (phase 1).

Tableau 20. Tableau RCRA.

Allocation des ressources					**Allocation des transitions de contrôle et de l' initiative**					**Allocation de l' autorité**		**Allocation de la responsabilité**	
Ressource [Partagée]	Description DOD	Type DOD	Posses-seur	Acqué-reur	Trans-ition	Initia-tive	*Handover* ou émetteur	*Takeover* ou receveur	Tâches et Fonctions	Tâche/ Fonction	Con-trainte	Tâche/ Fonction	Résultat attendu possible-ment impacté
					□								

1.2 Vue détaillée du processus RCRAFT

Cette section présente une vue détaillée de chacune des phases du processus RCRAFT. Nous détaillons d'abord la phase de modélisation des tâches explicitant l'allocation de fonctions et de tâches. Puis, nous détaillons la phase de description de l'allocation de ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Enfin nous détaillons la phase de modélisation RCRAFT.

1.2.1 Phase de modélisation des tâches explicitant l'allocation de fonctions et de tâches

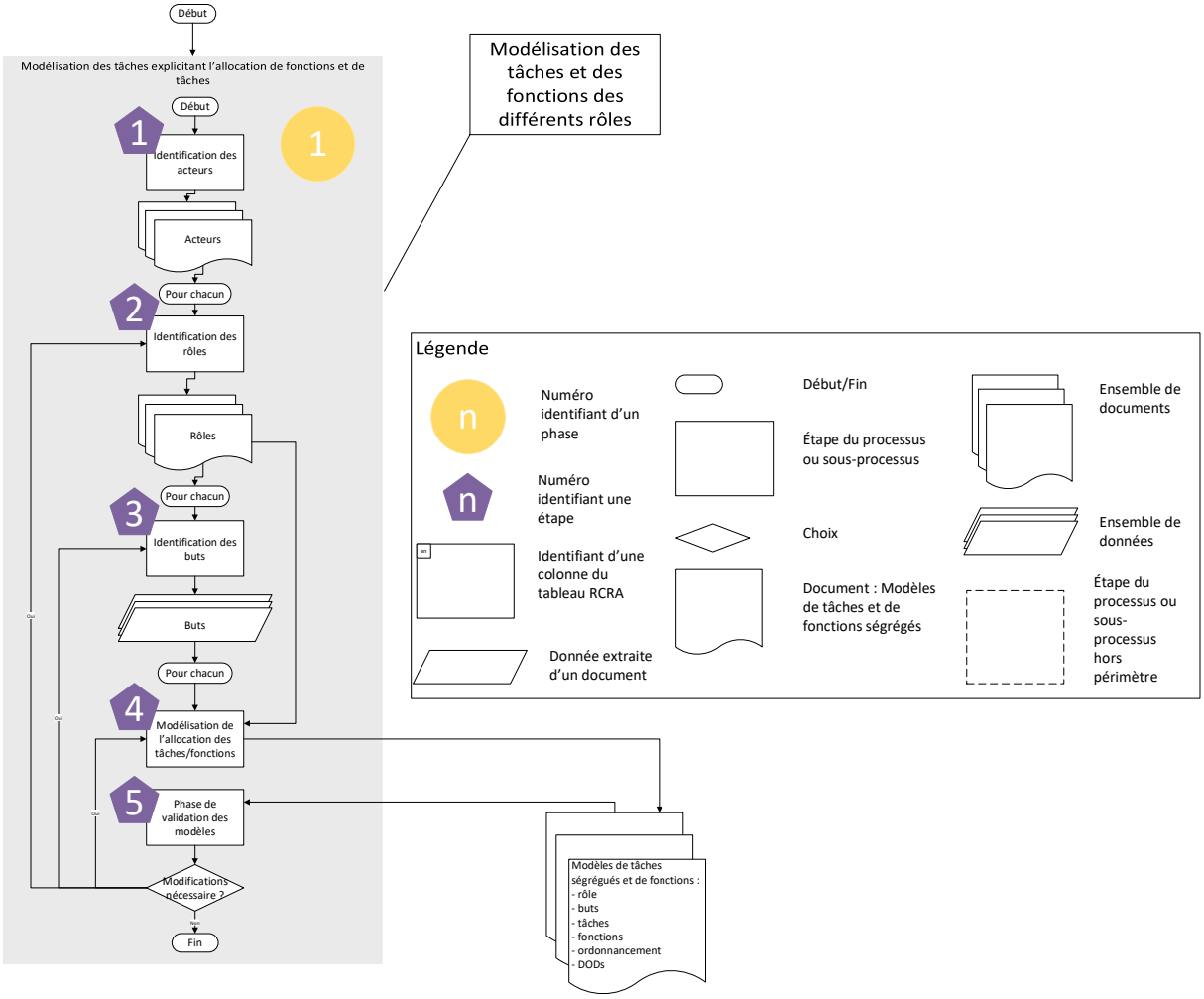


Figure 63. Vue détaillée de la phase de modélisation des tâches explicitant l'allocation de fonctions et de tâches.

La Figure 63 décrit en détail la phase intitulée « Modélisation des tâches explicitant l'allocation de fonctions et de tâches ». Cette phase nous permet de produire les modèles de tâches ségrégués et de fonctions. Il s'agit de la première phase du processus RCRAFT.

1 La première étape de cette phase consiste à identifier les acteurs. Selon l'approche proposée dans cette thèse, le ou les systèmes impliqués sont également considérés comme acteurs.

2. La seconde étape de consiste à identifier les rôles des acteurs identifiés. Selon l'approche proposée dans cette thèse, le ou les acteurs systèmes impliqués ont également des rôles.
3. La troisième étape consiste à identifier le ou les buts des rôles des acteurs utilisateurs que le ou les systèmes soutiennent.
4. La quatrième étape consiste à modéliser les tâches des rôles attribués à des utilisateurs pour atteindre leurs buts et les fonctions des rôles attribués à des systèmes pour soutenir les utilisateurs dans l'atteinte de leurs buts. Le principe de modélisation utilisé est décrit au Chapitre 5. Les fonctions systèmes sont modélisées hiérarchiquement avec un niveau d'abstraction similaire à la modélisation des tâches utilisateurs afin de pouvoir réallouer la fonction à un autre acteur si besoin. Cependant, ces modèles de fonctions, s'ils nous permettent d'analyser par la suite l'allocation des éléments RCRA entre les rôles dont le comportement est décrit de manière similaire, ne constituent pas une description formelle du comportement du système et n'existent qu'à des fins d'analyse. Dans ce processus, nous proposons d'utiliser l'outil HAMSTERS.
5. La cinquième étape consiste à tester les modèles de tâches ségrégués et de fonctions modélisés avec HAMSTERS. Pour cela, il est possible d'exécuter différents scénarios sur chacun des modèles afin de tester la complétude et l'exactitude des tâches et fonctions modélisées (voir Chapitre 3.3.4.2). Si des modifications des modèles sont nécessaires, la phase itère sur l'étape 4.

1.2.2 Phase de description de l'allocation de ressources, les transitions de contrôle et de de l'initiative, de la responsabilité et de l'autorité

La phase 2 du processus RCRAFT (voir Figure 62) permet d'effectuer l'analyse de l'allocation de ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité et sur les modèles de tâches ségrégués et de fonctions produits en phase 1. Cette phase produit le tableau RCRA.

La phase 2 propose de parcourir systématiquement les DODs des modèles de tâches ségrégués et de fonction afin d'analyser ce DOD et les tâches et fonctions des rôles le manipulant en termes d'allocation de ressources, des transitions de contrôle et d'initiative, de responsabilité et d'autorité.

1.2.2.1 Extraction des DODs

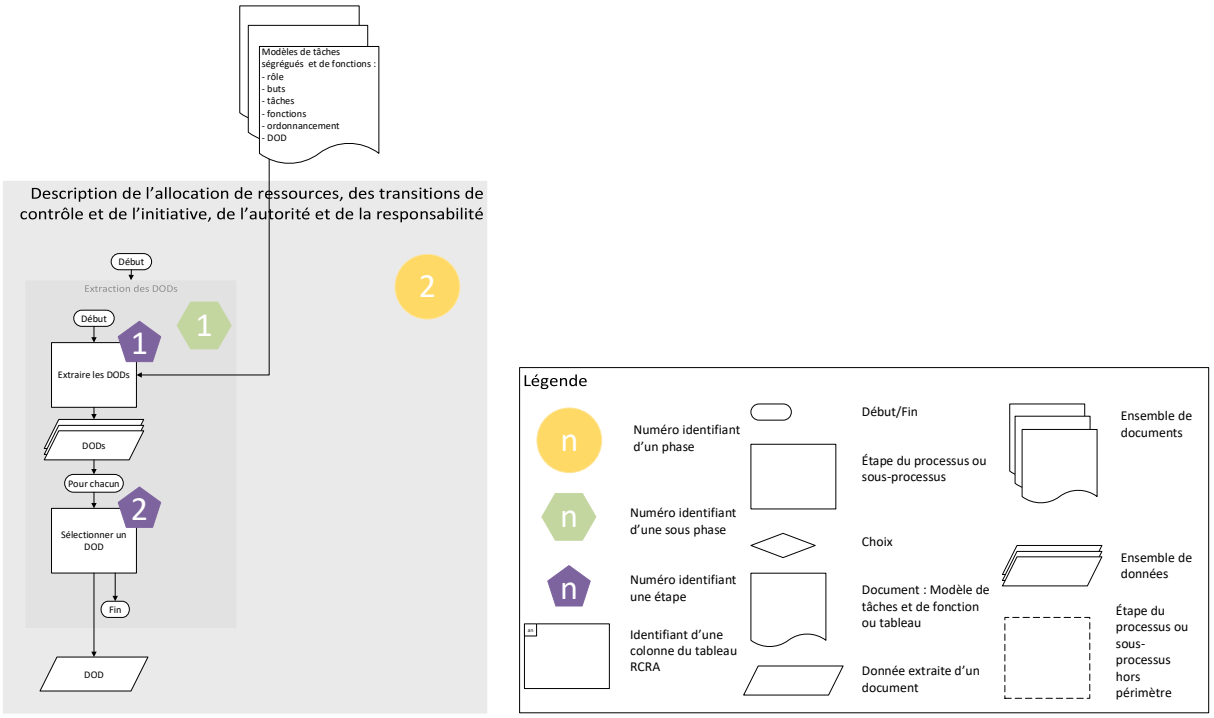


Figure 64. Vue détaillée de la sous-phase d'extraction des DODs.

La première sous-phase « Extraction des DODs » de la phase 2 permet d'extraire les DODs des modèles de tâches ségrégués et de fonctions et de les sélectionner un à un pour les sous-phases suivantes.

la Figure 64 présente la vue détaillée de cette sous-phase. La sous-phase 1 est composée des étapes suivantes :

1. La première étape consiste à extraire tous les DODs des modèles de tâches ségrégués et de fonctions afin de les traiter un par un.
2. La seconde étape consiste à sélectionner un DOD à traiter dans les sous-phases suivantes.

1.2.2.2 Description de l'allocation de ressources

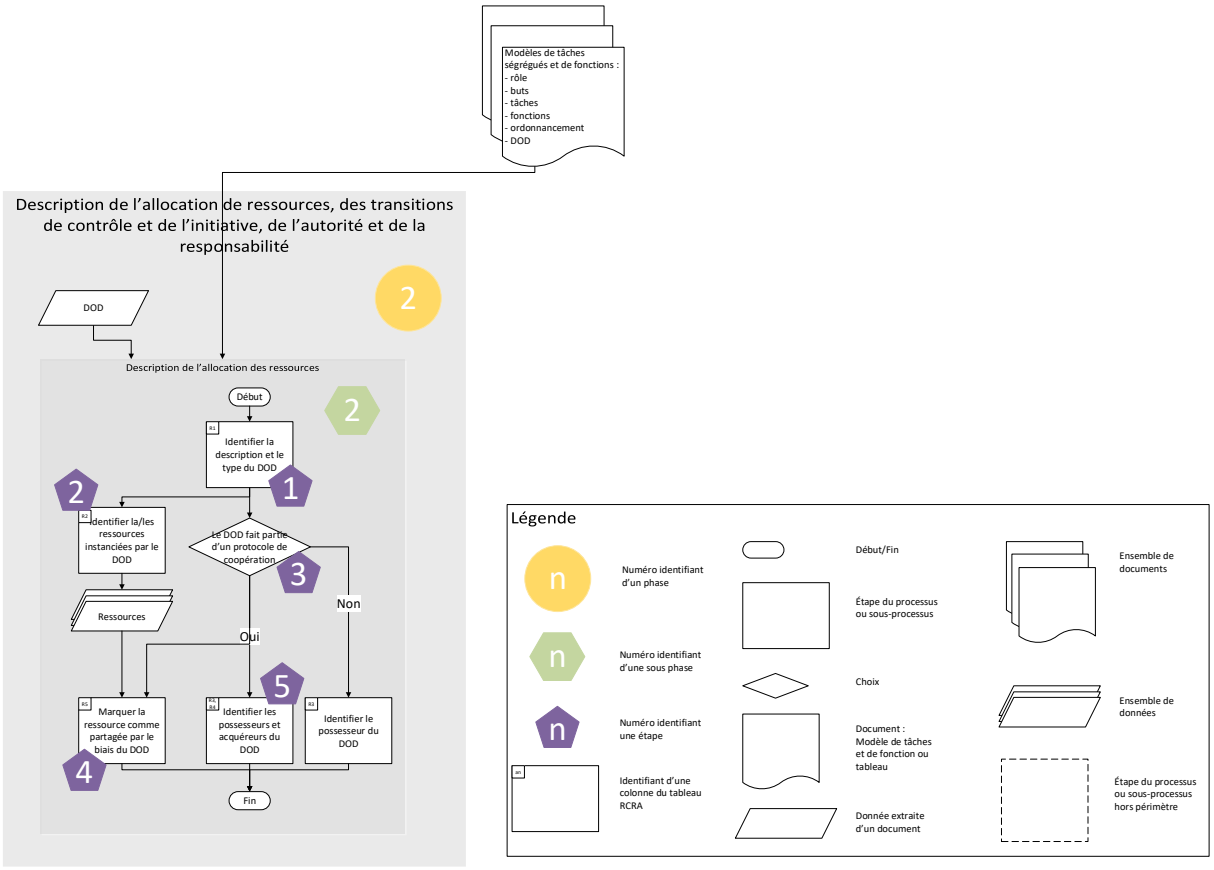


Figure 65. Vue détaillée de la sous-phase de description de l'allocation de ressources

La sous-phase 2 de la phase 2 permet de décrire l'allocation des ressources (défini en Chapitre 4.3) dans les colonnes du tableau RCRA présentées dans le Tableau 21.

La Figure 65 présente la vue détaillée de la sous-phase 2. La sous-phase 2 est composée des étapes suivantes (l'ordre de description des étapes est arbitraire pour les étapes 2, 3, 4, 5 et 6) :

1. La première étape consiste à identifier la description et le type du DOD. Cette étape permet de remplir les colonnes notées R1.
2. L'étape numérotée 2 consiste à identifier la ou les ressources instanciées par le DOD. Par exemple, dans l'exemple du jeu des 15, l'information « Le tour est terminé » du rôle Leader est une instanciation de la ressource « Fin de tour ». Une *concept map* peut permettre de faciliter l'identification des ressources. Par exemple, dans l'exemple du jeu des 15, « Fin de tour » est une ressource identifiée comme nécessaire au jeu dans la *concept map* du jeu des 15 (voir Figure 48). Cette étape permet de remplir la colonne notée R2 avec la ou les ressources identifiées.
3. Le choix numéroté 3 consiste à identifier si le DOD est une instance d'une ressource d'un protocole de coopération. Si c'est le cas, la sous-phase passe aux étapes 4 et 5. Sinon, la sous-phase passe à l'étape 6.

4 L'étape numérotée 4 consiste à marquer la ressource dont le DOD est une instance comme étant partagée entre des rôles. Cette étape permet d'ajouter l'indication [P] notée R5 dans le Tableau 21.

5 L'étape numérotée 5 consiste à identifier les possesseurs et les acquéreurs du DOD. Dans les modèles de tâches ségrégués ou de fonctions où le protocole de coopération est émis, le ou les rôles sont possesseurs du DOD. Dans les modèles de tâches ségrégués ou de fonctions où le protocole de coopération est reçu, le ou les rôles sont acquéreurs du DOD. Cette étape permet de remplir les colonnes R3 et R4 du Tableau 21 avec les possesseurs et acquéreurs identifiés.

6 L'étape numérotée 6 consiste à identifier le possesseur du DOD. Cette étape permet de remplir la colonne R3 du Tableau 21.

Tableau 21. Colonnes du tableau RCRA décrivant l'allocation des ressources

Allocation des ressources				
Ressource R2 [Partagée] R5	Description DOD R1	Type DOD R1	Possesseur R3	Acquéreur R4

1.2.2.3 Description de l'allocation des transitions de contrôle et de l'initiative

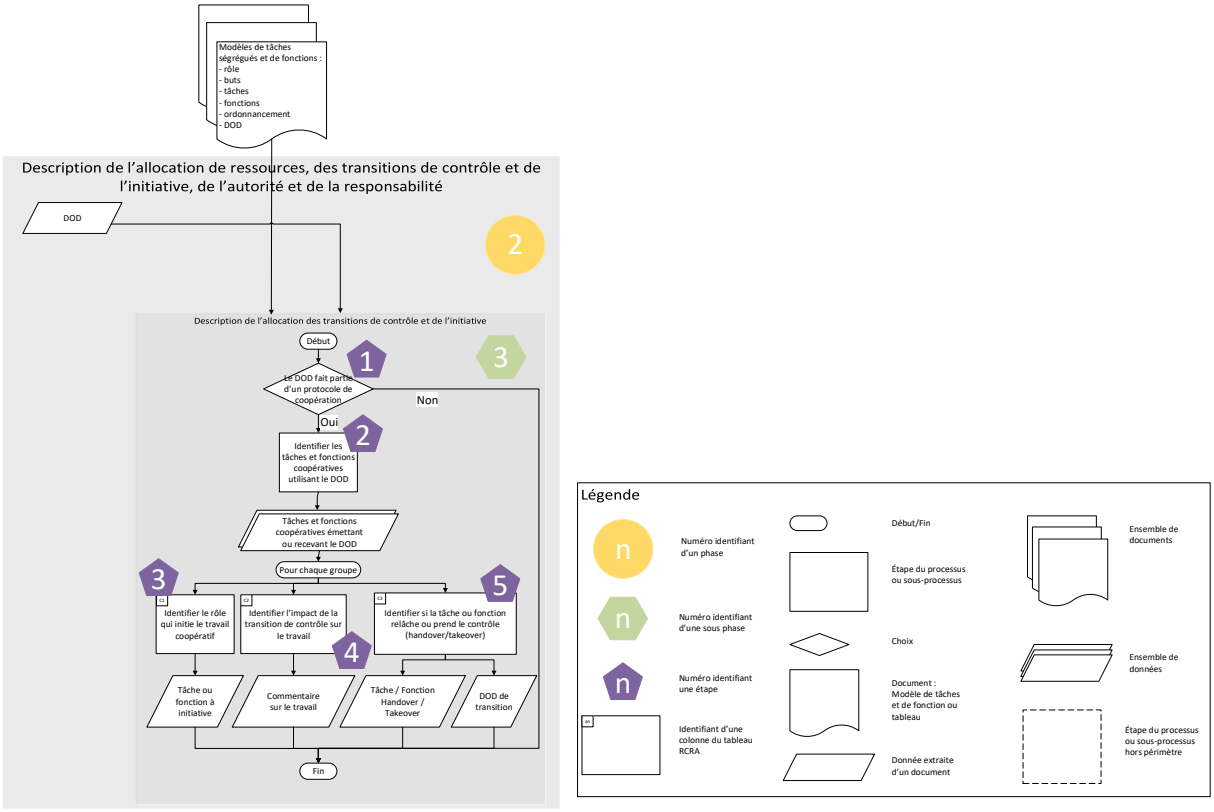


Figure 66. Vue détaillée de la sous-phase de description de l'allocation des transitions de contrôle et de l'initiative

La sous-phase 3 permet de décrire l'allocation des transitions de contrôle et de l'initiative (défini en Chapitre 4.4) dans les colonnes du tableau RCRA présentées dans le Tableau 22. Pour cette sous-phase, les modèles de tâches ségrégués et de fonctions produits lors de la phase 1 sont utilisés.

La Figure 66 présente la vue détaillée de la sous-phase 3 de la phase 2. La sous-phase 3 est composée des étapes suivantes (l'ordre de description des étapes est arbitraire pour les étapes 3, 4 et 5) :

1. Le choix numéroté 1 consiste à déterminer si le DOD est une instance d'une ressource d'un protocole de coopération. Si c'est le cas, la sous-phase passe à l'étape 2. Sinon, la sous-phase se termine.
2. L'étape numérotée 2 consiste à identifier les tâches et fonctions coopératives des modèles ségrégués qui émettent ou reçoivent le DOD. Les étapes suivantes sont appliquées à chacune des tâches et fonctions identifiées.
3. L'étape numérotée 3 consiste à identifier le rôle qui initie la coopération. C'est-à-dire que l'étape identifie la ou les tâches ou fonctions qui émettent le DOD lors de la coopération. Le ou les rôles qui réalisent ces tâches ou fonctions coopératives initient la coopération. Cette étape permet de remplir la colonne C1 du Tableau 22. Si la coopération implique plus de deux rôles, nous créons autant de lignes que de rôles qui reçoivent le DOD.
4. L'étape numérotée 4 consiste à commenter si la coopération provoque une interruption d'une tâche ou d'une fonction, déclenche une tâche ou une fonction, met fin à une tâche ou une fonction, etc. Cette étape permet de remplir la colonne C2 du Tableau 22.
5. L'étape numérotée 5 consiste à identifier si des tâches ou fonctions coopératives de la coopération prennent ou relâchent le contrôle sur un DOD et, donc, sur les tâches ou fonctions qui le manipule. Si lors de la coopération, un des rôles communique qu'il ne manipule plus un DOD et laisse les autres rôles le manipuler, il réalise un *handover*. Si lors de la coopération, un des rôles communique qu'il retire la possibilité aux autres rôles de manipuler un DOD, ce rôle réalise un *takeover*. Si la transition de contrôle ne peut pas être qualifiée de *handover* ou de *takeover*, la case à cocher de la colonne « Transition » est cochée et les tâches et fonction concernées sont renseignées sans les qualifier de *handover* ou de *takeover*. Par exemple, cette situation se produit lors d'un contrôle partagé entre les rôles du DOD. Cette étape permet de remplir les colonnes notées C3 et C4 du Tableau 22.
   Si le DOD identifié lors de cette étape est différent du DOD sélectionné lors de cette phase, alors le DOD identifié est indiqué dans la colonne C4. Nous indiquons alors, que le DOD sélectionné lors de cette phase permet de communiquer lors de la coopération une transition de contrôle sur le DOD identifié.
   Si les tâches ou fonctions coopératives ne prennent ou ne relâchent pas le contrôle sur un DOD durant la coopération, seules les tâches et fonctions coopératives sont renseignées.

Tableau 22. Colonnes du tableau RCRA décrivant les transitions de contrôle et l'allocation de l'initiative

Allocation des transitions de contrôle et de l'initiative				
Transition C4	Initiative C1	*Handover* ou émetteur C3	*Takeover* ou receveur C3	Tâches et Fonctions C2

□ [DOD]				

1.2.2.4 Description de l'allocation de la responsabilité

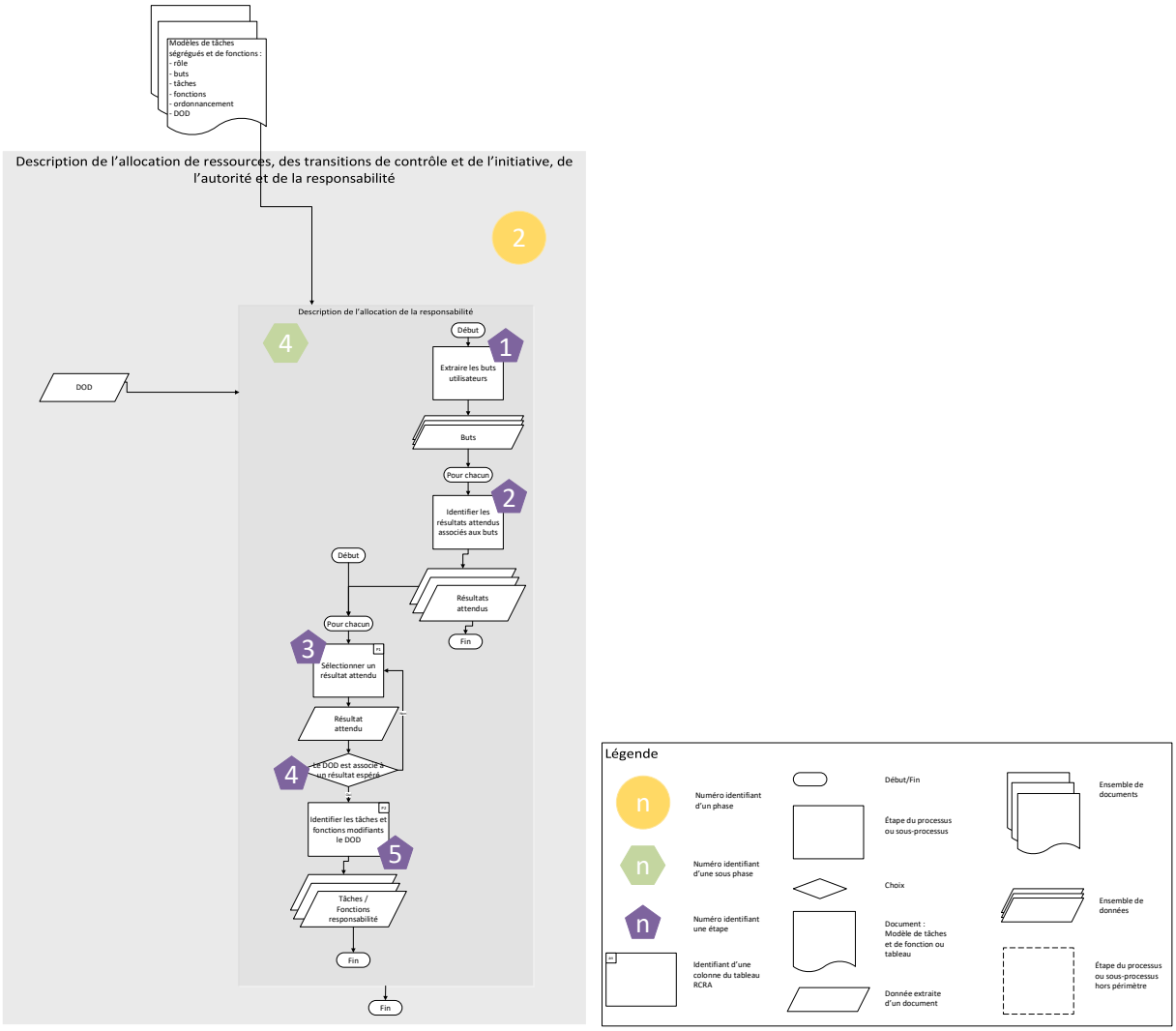


Figure 67. Vue détaillée de la sous-phase de description de l'allocation de la responsabilité.

La sous-phase 4 permet de décrire l'allocation de la responsabilité (défini en Chapitre 4.6) dans les colonnes du tableau RCRA présentées dans le Tableau 23.

La Figure 67 présente la vue détaillée de la sous-phase 4 de la phase 2. La sous-phase 4 est composée des étapes suivantes :

- Les étapes 1 et 2 permettent d'identifier les résultats attendus :
  - 1 L'étape numérotée 1 consiste à extraire les buts utilisateurs modélisés dans les modèles de tâches ségrégués et comment les rôles attribués à des acteurs systèmes soutiennent ces buts.
  - 2 Pour chacun des buts, l'étape numérotée 2 consiste à identifier le ou les résultats attendus permettant de valider ou non pour atteindre les buts visés par les rôles.

Par exemple, « Posséder 3 jetons dont la somme est égale à 15 » pour le but « gagner le jeu des 15 ».

- Les étapes suivantes permettent d'identifier si une modification du DOD peut faire dévier un résultat attendu :
  3. L'étape numérotée 3 consiste à sélectionner un résultat attendu parmi les résultats attendus identifiés en étape 2. Cette étape permet de remplir la colonne P2 du Tableau 23.
  4. Le choix numéroté 4 consiste à identifier si le DOD est associé au résultat attendu. Cette étape propose de répondre à la question : est-ce que la valeur du DOD à une influence sur le résultat attendu ? Si c'est le cas, la sous-phase passe à l'étape 6. Sinon, la sous-phase itère sur l'étape 3.
  5. L'étape numérotée 5 consiste à identifier toutes les tâches et fonctions qui modifient, créent ou suppriment le DOD. Cette étape permet de remplir la colonne P1 du Tableau 23.

Tableau 23. Colonnes du tableau RCRA décrivant l'allocation de la responsabilité.

Allocation de la responsabilité	
Tâche/ Fonction P1	Résultat attendu possiblement impacté P2

1.2.2.5 Description de l'allocation de l'autorité

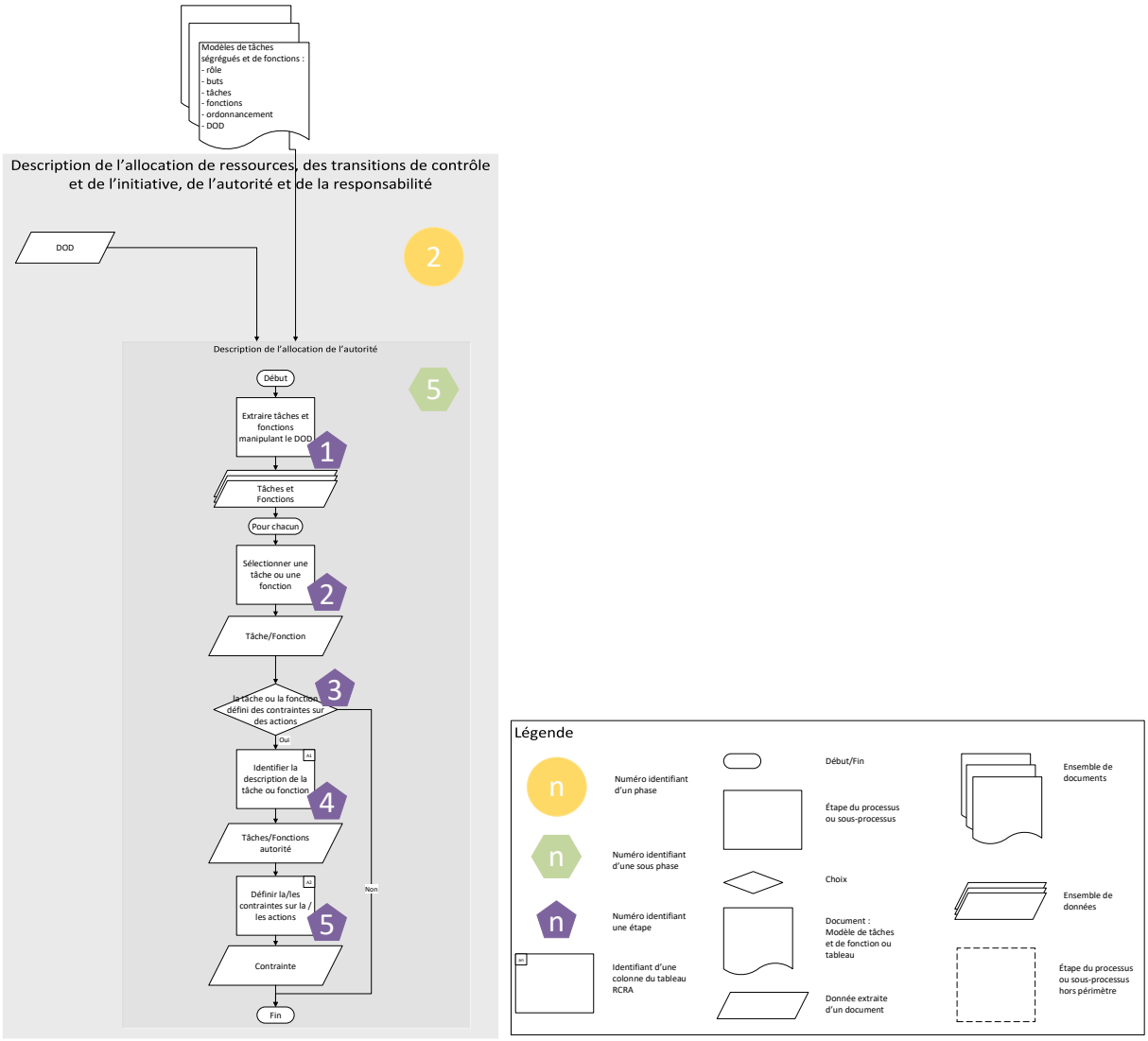


Figure 68. Vue détaillée de la sous-phase de description de l'allocation de l'autorité

La sous-phase 5 permet de décrire l'allocation de l'autorité (défini en Chapitre 4.4) dans les colonnes du tableau RCRA présentées dans le Tableau 24.

La Figure 68 présente la vue détaillée de la sous-phase 5 de la phase 2. La sous-phase 5 est composée des étapes suivantes :

1. L'étape numérotée 1 consiste à identifier les tâches et les fonctions manipulant le DOD. Les étapes suivantes sont appliquées à chacune des tâches et fonctions identifiées.
2. L'étape numérotée 2 consiste à sélectionner une tâche ou fonction parmi les tâches et fonctions identifiés.
3. Le choix numérotée 3 consiste à identifier si la tâche ou fonction sélectionnée défini une contrainte sur une tâche ou fonction par le biais du DOD ou réalise une tâche ou fonctions changeant l'état du système, d'un objet ou de l'environnement pour se rapprocher du but. Cette étape propose de répondre aux questions : est-ce que la tâche ou fonction restreint la liberté d'action d'un autre rôle ? Est-ce que la tâche ou la

fonction contrôle la réalisation de la tâche ou fonction ? Si c'est le cas, la sous-phase passe à l'étape 4. Sinon la sous-phase se termine pour la tâche ou fonction sélectionnée et la sous-phase itère sur l'étape 2.

4 L'étape 4 consiste à identifier la description de la tâche ou fonction à autorité. Cette étape permet de remplir la colonne notée A1 dans le Tableau 24.

5 L'étape 5 consiste à identifier la contrainte définie ou ce qui est réalisé par la tâche ou fonction à autorité. Cette étape permet de remplir la colonne notée A2 dans le Tableau 24.

Tableau 24. Colonnes du tableau RCRA décrivant l'allocation de l'autorité

Allocation de l'autorité	
Tâche/Fonction A1	Contrainte A2

1.2.3 Phase de Modélisation et d'Analyse RCRAFT

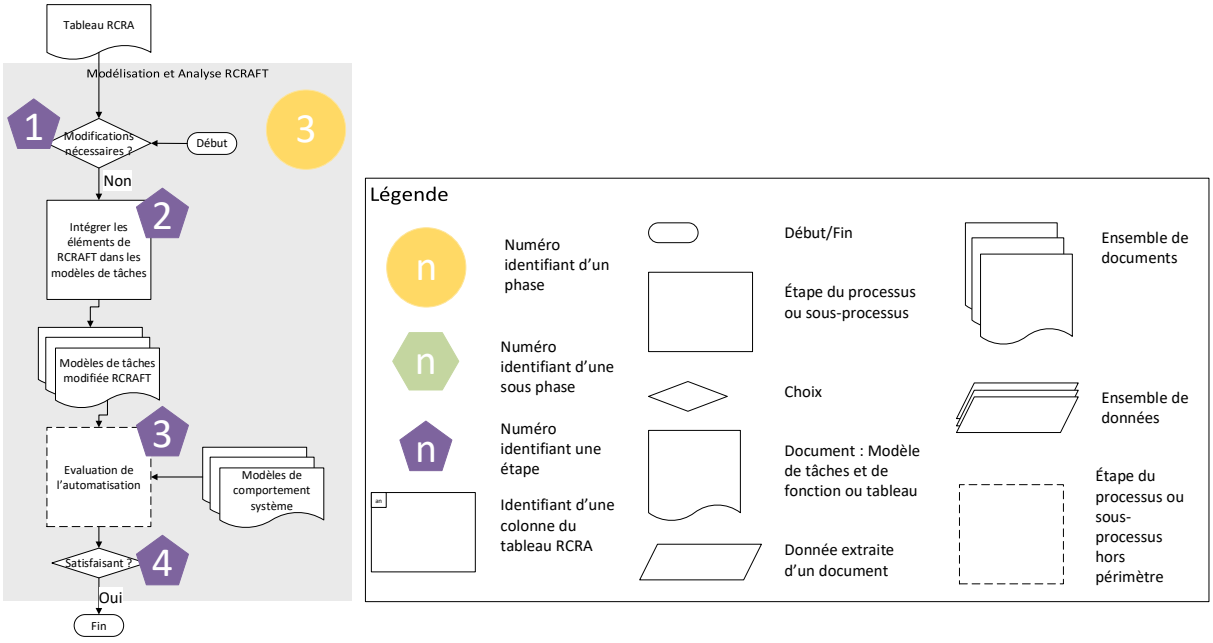


Figure 69. Vue détaillée de la phase 3 du processus RCRAFT.

La Figure 69 décrit en détail la phase 3 intitulée « Modélisation RCRAFT ». Cette phase nous permet d'intégrer les éléments RACRFT dans les modèles de tâches ségrégués et de fonctions ainsi que d'évaluer l'automatisation. Elle prend en entrée le tableau RCRA produit lors de la phase 2.

1 Le choix numéroté 1 consiste à déterminer si l'analyse réalisée en phase 2 a soulevé des problèmes de modélisation. Si des modifications des modèles sont nécessaires, le processus itère en phase 1. Sinon, la phase 3 passe à l'étape 2.

2 L'étape 2 consiste à utiliser les éléments de notation RCRA proposé par l'outil HAMSTERS présenté au Chapitre 5.3 pour intégrer les éléments RCRA identifié dans le tableau RCRA dans les modèles de tâches ségrégués et de fonctions.

3 L'étape numérotée 3 consiste à évaluer l'automatisation par rapport aux objectifs de conception. Au cours de cette étape des experts peuvent utiliser les artefacts de ce processus pour l'évaluation de la *situation awareness*, de la transparence ou de la confiance (*trust*) par exemple (voir Chapitre 2.2). Cette étape ne fait pas partie du périmètre de la thèse.

4 Si le résultat de l'évaluation est satisfaisant, le processus RCRAFT est terminé.

2 Application du processus RCRAFT à l'exemple du jeu des 15

Cette section présente l'application du processus RCRAFT à l'exemple du Jeu des 15. Les artefacts produits sont : les modèles de tâches ségrégués et de fonctions pour jouer au jeu des 15 et la première ligne du tableau RCRA du Jeu des 15.

2.1 Phase de modélisation des tâches explicitant l'allocation de fonctions et de tâches

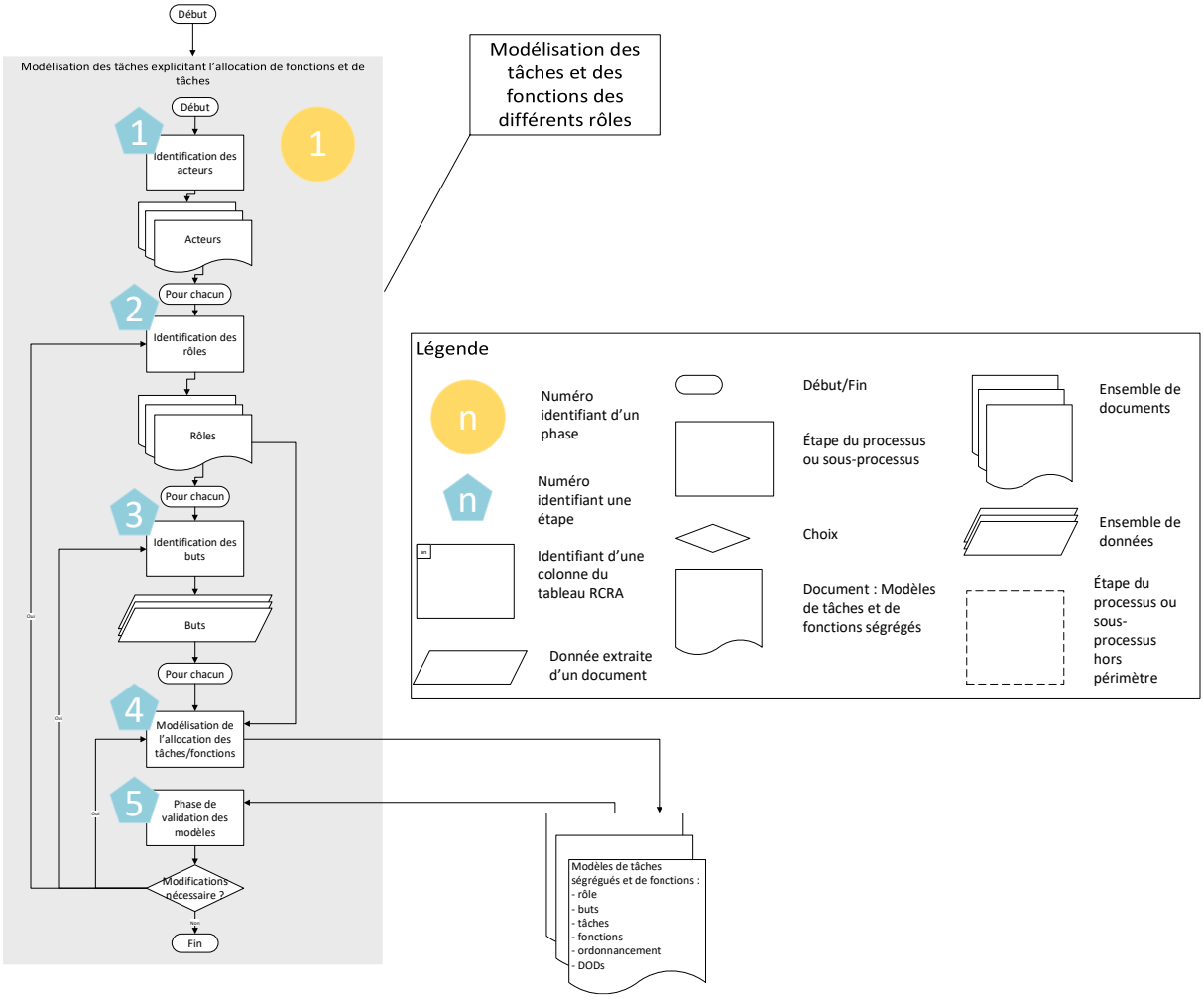


Figure 70. Vue détaillée du sous-processus de modélisation des tâches explicitant l'allocation de fonctions et de tâches avec les étapes explicitées numérotées de 1 à 7 dans des pastilles bleues.

La Figure 70 présente la vue détaillée de la phase 1 du processus qui permet d'expliciter l'allocation de fonctions et de tâches. Les étapes parcourues durant notre exemple sont détaillées ci-après et numérotées de 1 à 5 sur la Figure 70.

1. Dans un premier temps nous devons identifier les acteurs. Nous identifions deux acteurs utilisateurs (humains) et un acteur système.
2. Ensuite, nous identifions les rôles de chacun de ces acteurs. Les acteurs utilisateurs ont le rôle de Joueur, puis de *Leader* ou de *Challenger* selon l'allocation des rôles affichée. L'acteur système a le rôle d'arbitre.
3. Nous identifions les buts pour chacun des rôles pour notre exemple. Le rôle Joueur a pour but de jouer une partie du jeu des 15. Nous considérons la situation où les deux joueurs jouent pour gagner. Le rôle *Leader* a pour but de jouer pour gagner la partie en tant que Leader que nous notons « Gagner le jeu des 15 ». Le rôle *Challenger* a pour but de jouer pour gagner la partie en tant que Challenger que nous notons « Gagner le jeu des 15 ». Le rôle Arbitre soutient les buts des utilisateurs en soutenant la partie dans le respect des règles du jeu des 15.
4. Nous modélisons les tâches permettant d'atteindre chacun des buts des joueurs identifiés. Également, nous modélisons les fonctions de l'arbitre permettant de soutenir la partie dans le respect des règles du jeu des 15. Les modèles de tâches ségrégués et de fonctions correspondants sont consultables ci-après (voir Figure 71, Figure 72, Figure 73 et Figure 74).
5. Ensuite, nous testons les modèles de tâches ségrégués et de fonctions produits en testant différents scénarios sur les modèles de tâches ségrégués et de fonctions des différents rôles. S'il n'y a pas de modification nécessaire identifiée, cette première phase est terminée.

Les Figure 71, Figure 72, Figure 73 et Figure 74 sont les modèles de tâches ségrégués et de fonctions produits lors de la phase 1 :

- La Figure 71 présente la modélisation de tâches ségréguée HAMSTERS du rôle Joueur pour le but « Jouer une partie du jeu des 15 ». Après l'allocation des rôles *Leader* et *Challenger*, le joueur réalise les tâches de la *sub-routine* correspondante au rôle alloué.
- La Figure 72 présente la modélisation de tâches ségréguée HAMSTERS du rôle *Leader* pour le but « Jouer pour gagner la partie en tant que Leader ». Nous notons le but « Jouer pour gagner la partie en tant que Leader » *Gagner le jeu des 15* dans la suite de cette thèse.
- La présente Figure 73 la modélisation de tâches ségréguée HAMSTERS du rôle Challenger pour le but « Jouer pour gagner la partie en tant que Challenger ». Nous notons le but « Jouer pour gagner la partie en tant que Challenger » *Gagner le jeu des 15* dans la suite de cette thèse.
- La Figure 74 présente la modélisation de fonctions HAMSTERS du rôle Arbitre pour le but « Soutenir la partie dans le respect des règles du jeu des 15 ».

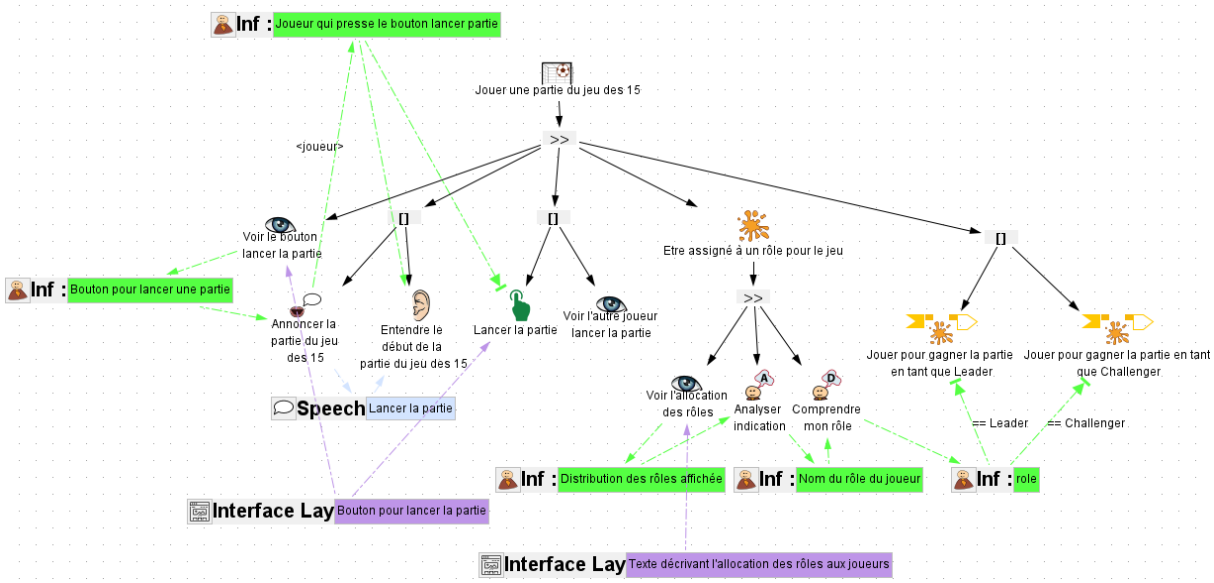


Figure 71. Modèle de tâches ségrégué du rôle Joueur pour Jouer au jeu des 15

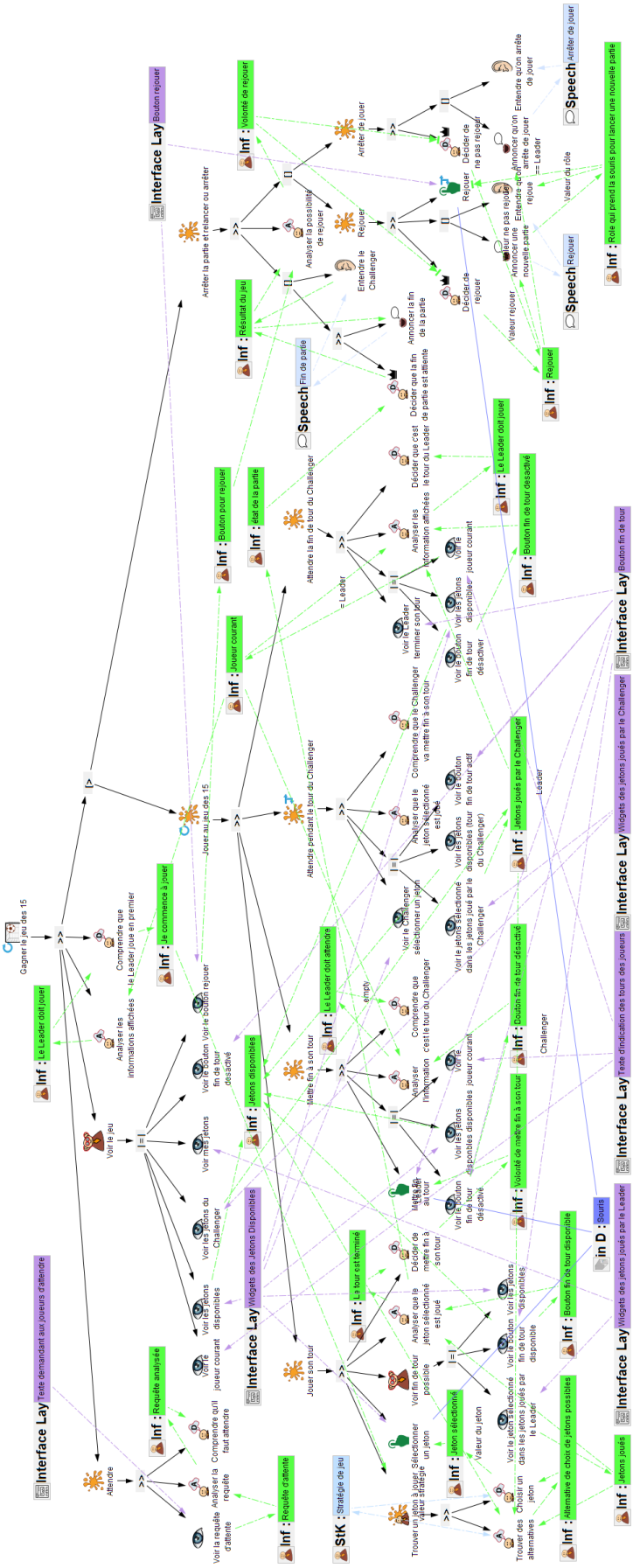


Figure 72. Modèle de tâches ségrégué du rôle Leader pour gagner le jeu

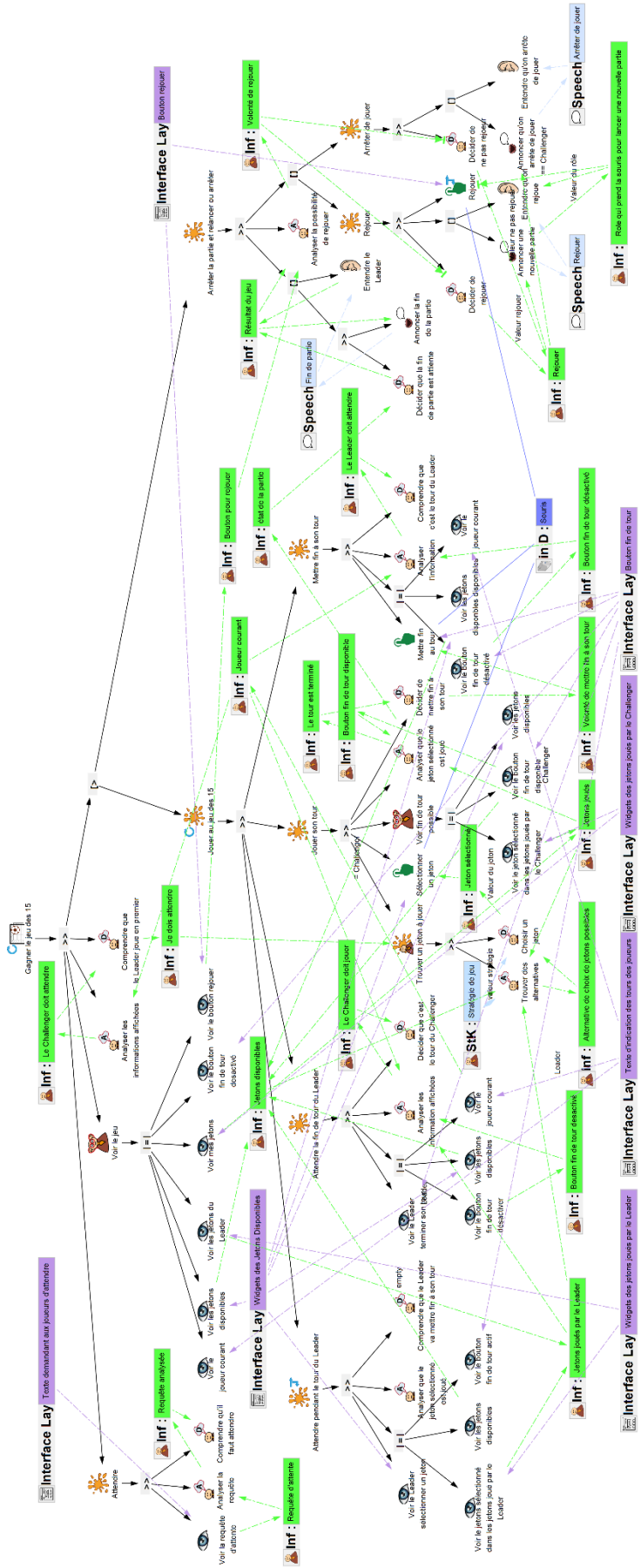


Figure 73. Modèle de tâches ségrégué du rôle Challenger pour gagner le jeu

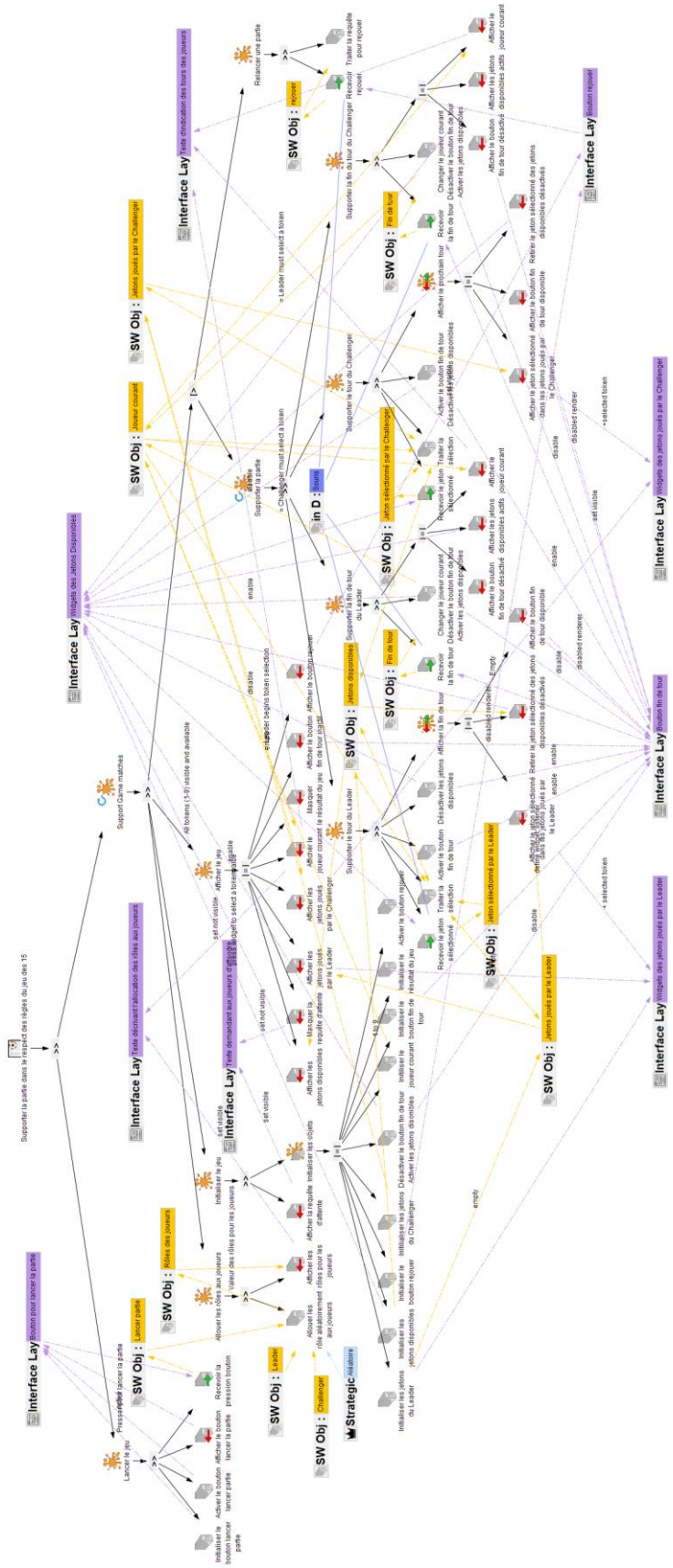


Figure 74. Modèle de fonctions du rôle Arbitre pour faire respecter les règles du jeu des 15

Les modèles de tâches ségrégués et de fonctions des rôles Joueur, Arbitre, *Leader* et *Challenger* sont utilisés dans la prochaine phase du processus.

2.2 Phase de description de l'allocation de ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité

La phase 2 du processus nous permet d'analyser et décrire l'allocation de ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité entre les rôles Joueur, *Leader*, *Challenger* et Arbitre dans le tableau RCRA. Seule une itération de cette phase est décrite sur l'exemple du Jeu des 15. L'artefact produit lors de cette phase est la première ligne du tableau RCRA du Jeu des 15.

Après avoir produit une première version des modèles de tâches ségrégués et de fonctions pour les rôles d'Arbitre, de Joueur, de *Leader* et de *Challenger* lors de la phase 1, nous utilisons ces artefacts pour effectuer la phase 2.

2.2.1 Extraction des DODs

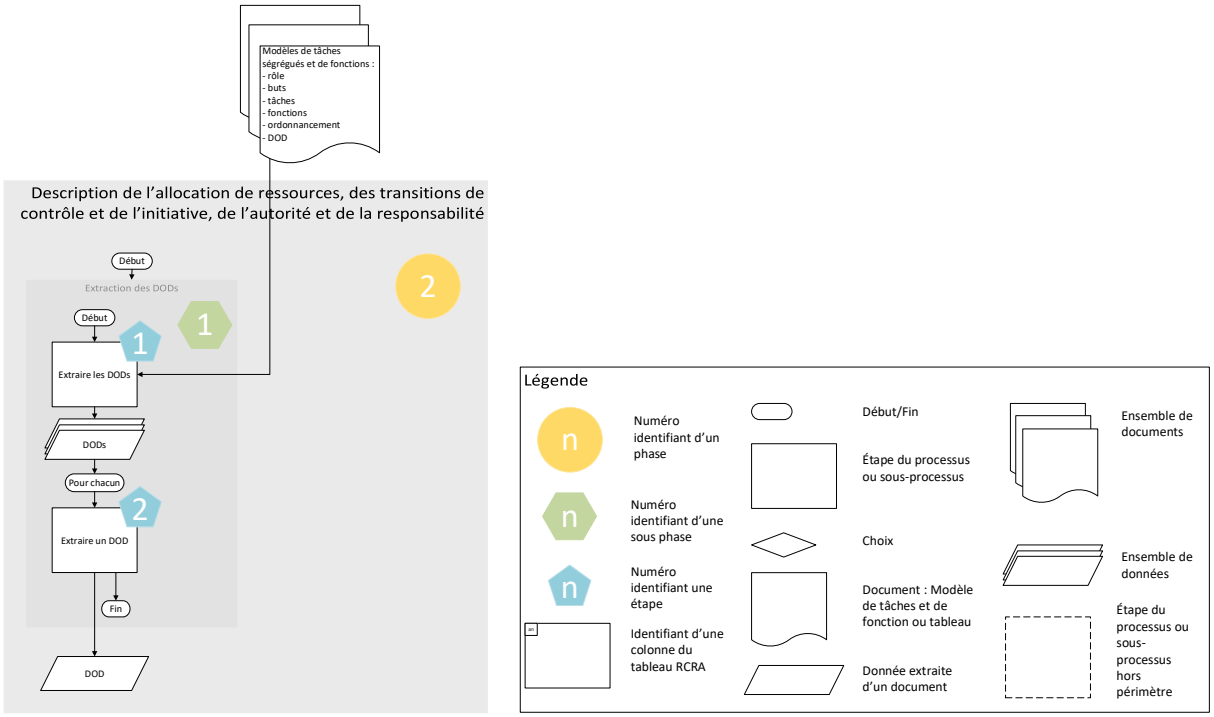


Figure 75. Vue détaillée de la sous-phase d'extraction des DODs avec les étapes explicitées numérotées de 1 et 2 dans des pastilles bleues.

La Figure 75 présente la vue détaillée de la sous-phase 1 de la phase 2. Les étapes parcourues durant notre exemple sont détaillées ci-après et numérotées de 1 à 2 sur la Figure 75 dans les pastilles bleues.

1. D'abord, nous extrayons les DODs décrits dans les modèles de tâches ségrégués et de fonctions des rôles Arbitre, Joueur, *Leader* et *Challenger* afin de les traiter un par un.

2. Parmi tous les DODs des modèles, nous sélectionnons le premier DOD de la liste décrivant un élément de la couche interface « Widgets des Jetons disponibles » visible sur la Figure 76. Les DODs décrivant des éléments de la couche interface sont appelés « Interface Lay. » pour *interface layer* dans les modèles de tâches ségrégués et les modèles de fonctions.

Puis, nous appliquons la suite du processus comme suit.

Figure 76. Couche interface "Widget des jetons disponibles"

2.2.2 Description de l'allocation de ressources

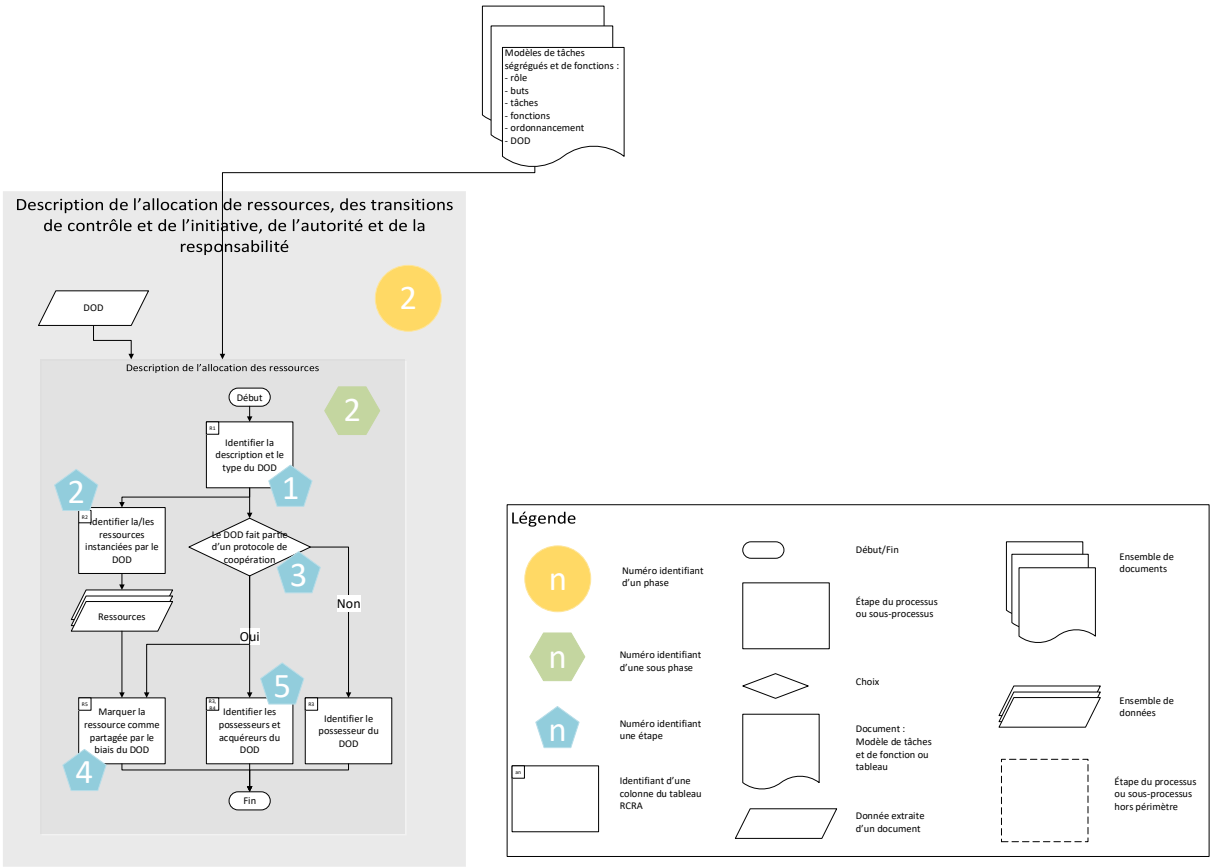


Figure 77. Vue détaillée de la sous-phase de description de l'allocation des ressources avec les étapes explicitées numérotées de 1 à 5 dans des pastilles bleues.

Afin de décrire l'allocation de ressources sur notre exemple, nous appliquons la sous-phase « Description de l'allocation de ressources » numéroté 2 dans la pastille verte de la Figure 77. Les étapes parcourues dans notre exemple sont numérotées dans les pastilles bleues de la Figure 77.

1. Nous identifions la description du DOD : « Widgets des Jetons disponibles ». Nous identifions le type de DOD : élément de la couche interface (noté « Interface Lay. » dans la Figure 76). Nous créons une nouvelle ligne dans le tableau RCRA et nous remplissons les colonnes notées R1 dans le Tableau 25 avec la description et le type de DOD.

2. Nous identifions que ce DOD permet de communiquer quels sont les jetons disponibles et quel est le jeton sélectionné à chaque tour entre les rôles lors de travails coopératifs. Le DOD est une instance de deux ressources nécessaires pour le jeu des 15 : les jetons disponibles et le jeton sélectionné à chaque tour. Nous remplissons la colonne « ressource » notée R2 dans le Tableau 25 avec ces deux ressources.
3. Nous identifions que ce DOD est utilisé lors de la coopération. Ce DOD est décrit dans le modèle « Soutenir la partie dans le respect des règles du jeu des 15 » du rôle Arbitre, dans le modèle « Gagner le jeu des 15 » décrivant un but du rôle *Leader* et dans le modèle « Gagner le jeu des 15 » décrivant un but du rôle *Challenger*. Nous suivons la flèche « Oui » pour les prochaines étapes.
4. Nous marquons les ressources « Jetons disponibles » et « Jeton sélectionné » comme étant partagées. Nous ajoutons [P] dans la colonne notée R5 du Tableau 25 pour indiquer que les ressources « Jetons disponibles » et le « Jeton sélectionné » sont partagées par le biais du DOD « Widgets des Jetons disponibles ».
5. Nous identifions le système ayant le rôle d'arbitre comme étant le possesseur du DOD et le *Leader* et le *Challenger* comme les rôles acquérant ce widget : ils peuvent le voir et cliquer dessus. Nous remplissons la colonne du Tableau 25 notée R3 avec l'Arbitre comme étant le possesseur du DOD. Nous remplissons la colonne notée R4 du Tableau 25 avec le *Leader* et le *Challenger* comme acquéreur du DOD.

Le résultat de ces étapes est la partie du tableau RCRA présentée dans la première ligne surlignée en bleu du Tableau 25. Le Tableau 25 présente le résultat pour la description de l'allocation des ressources à la fin de toutes les itérations de la phase 2.

Tableau 25. Colonnes de la partie « Allocation des ressources » du tableau RCRA rempli après les étapes décrites dans cette section.

Allocation des ressources				
Ressource R2 [Partagée] R5	Description DOD R1	Type DOD R1	Possesseur R3	Acquéreur R4
Jetons disponibles **[P]** Jeton Sélectionné **[P]**	Widgets des Jetons disponibles	Couche interface	Arbitre	*Leader*, *Challenger*
Jeton sélectionné **[P]**	Jeton sélectionné par le Leader	Objet Logiciel	Arbitre	
	Jeton sélectionné par le Challenger	Objet Logiciel	Arbitre	
	Jeton sélectionné	Information	*Leader*, *Challenger*	
	Alternatives de choix de jetons possibles	Information	*Leader*, *Challenger*	
Jetons disponibles **[P]**	Jetons disponibles	Objet Logiciel	Arbitre	
	Jetons disponibles	Information	*Leader*, *Challenger*	
Jetons du Leader **[P]**	Widgets des jetons joués par le Leader	Couche Interface	Arbitre	*Leader*, *Challenger*
	Jetons joués par le Leader	Objet logiciel	Arbitre	
	Jetons joués	Information	*Leader*, *Challenger*	

	Jetons joués par le Leader	Information	*Challenger*	
Jetons du Challenger **[P]**	Widgets des jetons joués par le Challenger	Couche Interface	Arbitre	*Leader*, *Challenger*
	Jetons joués par le Challenger	Objet logiciel	Arbitre	
	Jetons joués	Information	*Challenger*	
	Jetons joués par le Challenger	Information	*Leader*	
Fin de tour **[P]**	Bouton fin de tour	Couche Interface	Arbitre	*Leader*, *Challenger*
	Fin de tour	Objet Logiciel	Arbitre	
	Bouton fin de tour disponible	Information	*Leader*, *Challenger*	
	Volonté de mettre fin à son tour	Information	*Leader*, *Challenger*	
	Bouton fin de tour désactivé	Information	*Leader*, *Challenger*	
	Le tour est terminé	Information	*Leader*, *Challenger*	
Joueur Courant **[P]**	Texte d'indication des tours des joueurs	Couche Interface	Arbitre	*Leader*, *Challenger*
	Texte demandant aux joueurs d'attendre	Couche Interface	Arbitre	*Leader*, *Challenger*
	Joueur courant	Objet Logiciel	Arbitre	
	Je commence à jouer	Information	*Leader*	
	Je dois attendre	Information	*Challenger*	
	Le Leader doit attendre	Information	*Leader*	
	Le Challenger doit jouer	Information	*Challenger*	
	Le Leader doit jouer	Information	*Leader*	
	Le Challenger doit attendre	Information	*Challenger*	
	Joueur courant	Information	*Leader*, *Challenger*	
	Requête d'attente	Information	*Leader*, *Challenger*	
	Requête analysée	Information	*Leader*, *Challenger*	
Résultat de la partie **[P]**	Fin de partie	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Etat de la partie	Information	*Leader*, *Challenger*	
	Résultat du jeu	Information	*Leader*, *Challenger*	
Rejouer **[P]**	Bouton Rejouer	Couche Interface	Arbitre	*Leader*, *Challenger*
	Rejouer	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Arrêter de jouer	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Rejouer	Objet Logiciel	Arbitre	
	Volonté de rejouer	Information	*Leader*, *Challenger*	

	Bouton pour rejouer	Information	*Leader, Challenger*	
	Rejouer	Information	*Leader, Challenger*	
	Rôle qui prend la souris pour lancer une nouvelle partie	Information	*Leader, Challenger*	
Stratégie de jeu	Stratégie de jeu	Information Stratégique	*Leader, Challenger*	
Stratégie d'attribution des rôles	Aléatoire	Données Stratégique	Arbitre	
Lancer partie **[P]**	Bouton pour lancer la partie	Couche Interface	Arbitre	Joueur
	Lancer partie	Objet Logiciel	Arbitre	
	Lancer la partie	Parole	*Leader, Challenger*	*Leader, Challenger*
	Bouton pour lancer une partie	Information	Joueur	
	Joueur qui presse le bouton lancer la partie	Information	Joueur	
Rôle **[P]**	Texte décrivant l'allocation des rôles aux joueurs	Couche Interface	Arbitre	Joueur
	Rôles des joueurs	Objet Logiciel	Arbitre	
	Challenger	Objet Logiciel	Arbitre	
	Leader	Objet Logiciel	Arbitre	
	Distribution des rôles affichée	Information	Joueur	
	Rôle	Information	Joueur	

2.2.3 Description de l'allocation des transitions de contrôle et de l'initiative

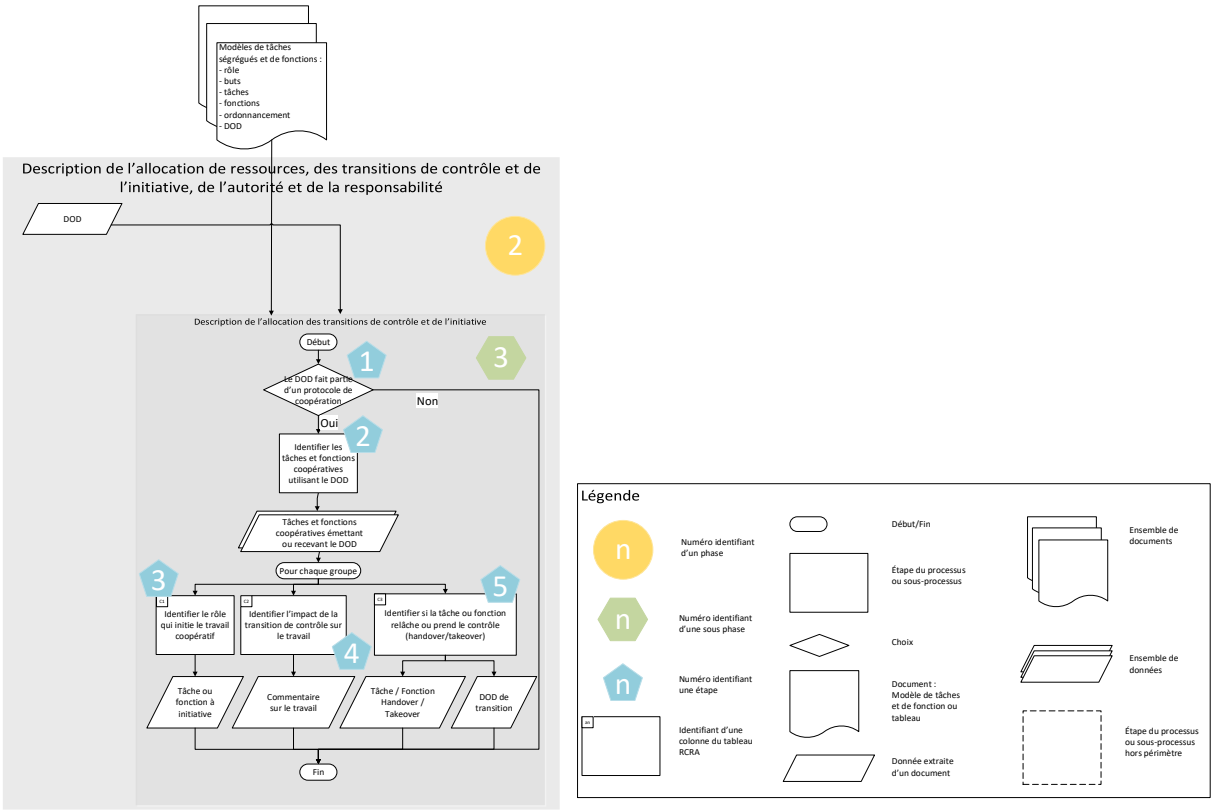


Figure 78. Vue détaillée de la sous-phase l'allocation des transitions de contrôle et de l'initiative avec les étapes explicitées numérotées de 1 à 4 dans des pastilles bleues.

Afin de décrire l'allocation des transitons de contrôles et de l'initiative sur notre exemple, nous allons appliquer la sous-phase « Description des transitions de contrôle et de l'allocation de l'initiative » numéroté 3 dans la pastille verte de la Figure 78. Nous appliquons la sous-phase comme suit. Les étapes parcourues dans notre exemple sont numérotées dans les pastilles bleues de la Figure 78.

1. Tout d'abord, nous identifions si le DOD « Widgets des Jetons disponibles » est une instance d'une ressource d'un protocole de coopération. Le DOD « Widgets des Jetons disponibles » permet de communiquer les jetons disponibles et le jeton sélectionné à chaque tour entre les rôles. Nous passons à l'étape 2.
2. Parmi toutes les coopérations, nous identifions les tâches et fonctions coopératives manipulant le DOD « Widgets des Jetons disponibles ». Dans cet exemple, nous détaillons seulement une itération en sélectionnant la coopération permettant de communiquer les jetons disponibles en début de partie. Parmi les tâches et fonctions coopératives composant cette coopération, nous sélectionnons la fonction de sortie coopérative « Afficher les jetons disponibles » de l'arbitre et la tâche perceptive coopérative « Voir les jetons disponibles » du *Leader*.
3. Nous identifions que la fonction de sortie coopérative « Afficher les jetons disponibles » de l'arbitre initie la coopération. Nous indiquons le rôle d'arbitre dans la colonne notée C1 du Tableau 26.

4. Nous identifions que cette coopération interrompt la tâche d'attente du *Leader* et du *Challenger*. Nous l'indiquons dans la colonne notée C2 du Tableau 26.
5. Nous identifions alors que l'arbitre relâche le contrôle sur les « Widgets des Jetons disponibles » : ils sont disponibles pour le *Leader*. Nous remplissons les colonnes notées C3 du Tableau 26. Nous indiquons la fonction de sortie coopérative « Afficher les jetons disponibles » de l'arbitre dans la colonne « *Handover* / émetteur » en indiquant la mention HANDOVER. Nous indiquons par une coche dans la colonne « transition » que le DOD permet d'effectuer une transition de contrôle sur ce même DOD.

Le résultat de ces étapes est la partie du tableau RCRA présentée dans la première ligne surlignée en bleu du Tableau 26. Le Tableau 26 présente le résultat pour la description de l'allocation des transitions de contrôle et de l'initiative à la fin de toutes les itérations de la phase 2.

Tableau 26. Colonnes de la partie « Allocation des transitions de contrôle et de l'initiative » du tableau RCRA complété durant les étapes 1 à 4 décrites ci-dessus.

Allocation des transitions de contrôle et de l'initiative				
Transition C3	Initiative C1	*Handover* ou émetteur C3	*Takeover* ou receveur C3	Tâches et Fonctions C2
X	Arbitre	Arbitre : « Afficher les jetons disponibles » **HANDOVER**	Leader : « Voir les jetons disponibles »	Fin attente pour le *Leader* et le *Challenger* *Leader* joue son tour *Challenger* attend l'arbitre finit d'afficher le jet et démarre le supporte du tour du *Leader*
X	Arbitre	Arbitre : « Afficher les jetons disponibles » **HANDOVER**	Challenger : « Voir les jetons disponibles »	
X	Arbitre	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Arbitre	
X	Leader	Leader : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »	
X	Challenger	Challenger : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Leader »	Leader : « Voir mes jetons »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Leader »	Challenger : « Voir les jetons du Leader »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Challenger »	Leader : « Voir les jetons du Challenger »	
	Arbitre	Arbitre : « Afficher les jetons joués	*Challenger :* « « Voir mes jetons »	

		par le Challenger »		
x	Arbitre	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Leader : « Voir le bouton fin de tour activé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Challenger : « Voir le bouton fin de tour activé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Arbitre	
x	Leader	Leader : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »	
x	Challenger	Challenger : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »	
	Arbitre	Arbitre : « Afficher le joueur courant »	Leader : « Voir le joueur courant »	
	Arbitre	Arbitre : « Afficher le joueur courant »	Challenger : « Voir le joueur courant »	
	Arbitre	Arbitre ; « Afficher la requête d'attente »	Leader : « Voir la requête d'attente »	
	Arbitre	Arbitre ; « Afficher la requête d'attente »	Challenger : « Voir la requête d'attente »	
	Leader	Leader : « Annoncer la fin de la partie »	Challenger : « Entendre le Leader »	Fin des tours de jeu
	Challenger	Challenger : « Annoncer la fin de la partie »	Leader : « Entendre le Challenger »	Fin des tours de jeu
X	Arbitre	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Leader : « Voir le bouton rejouer »	
X	Arbitre	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Challenger : « Voir le bouton rejouer »	
	Leader	Leader : « Rejouer »	Arbitre : « Recevoir rejouer »	
	Challenger	Challenger : « Rejouer »	Arbitre : « Recevoir rejouer »	
	Leader	Leader : « Annoncer une nouvelle partie »	Challenger : « Entendre qu'on rejoue »	
	Challenger	Challenger : « Annoncer une nouvelle partie »	Leader : « Entendre qu'on rejoue »	
	Leader	Leader : « Annoncer qu'on arrête de jouer »	Challenger : « Entendre qu'on arrête de jouer »	Fin du jeu des 15
	Challenger	Challenger : « Annoncer qu'on arrête de jouer »	Leader : « Entendre qu'on arrête de jouer »	Fin du jeu des 15
x	Arbitre	Arbitre : « Afficher le bouton lancer la partie » **HANDOVER**	Joueur : « Voir le bouton lancer la partie »	
x	Joueur	Joueur : « Lancer la partie » **HANDOVER**	Arbitre : « Recevoir la pression bouton »	
	Joueur	Joueur : « Annoncer la partie du jeu des 15 »	Joueur : « Entendre le début de la partie du jeu des 15 »	
	Arbitre	Arbitre : « Afficher les rôles pour les joueurs »	Joueur : « Voir l'allocation	

			des rôles »	

2.2.4 Description de l'allocation de la responsabilité

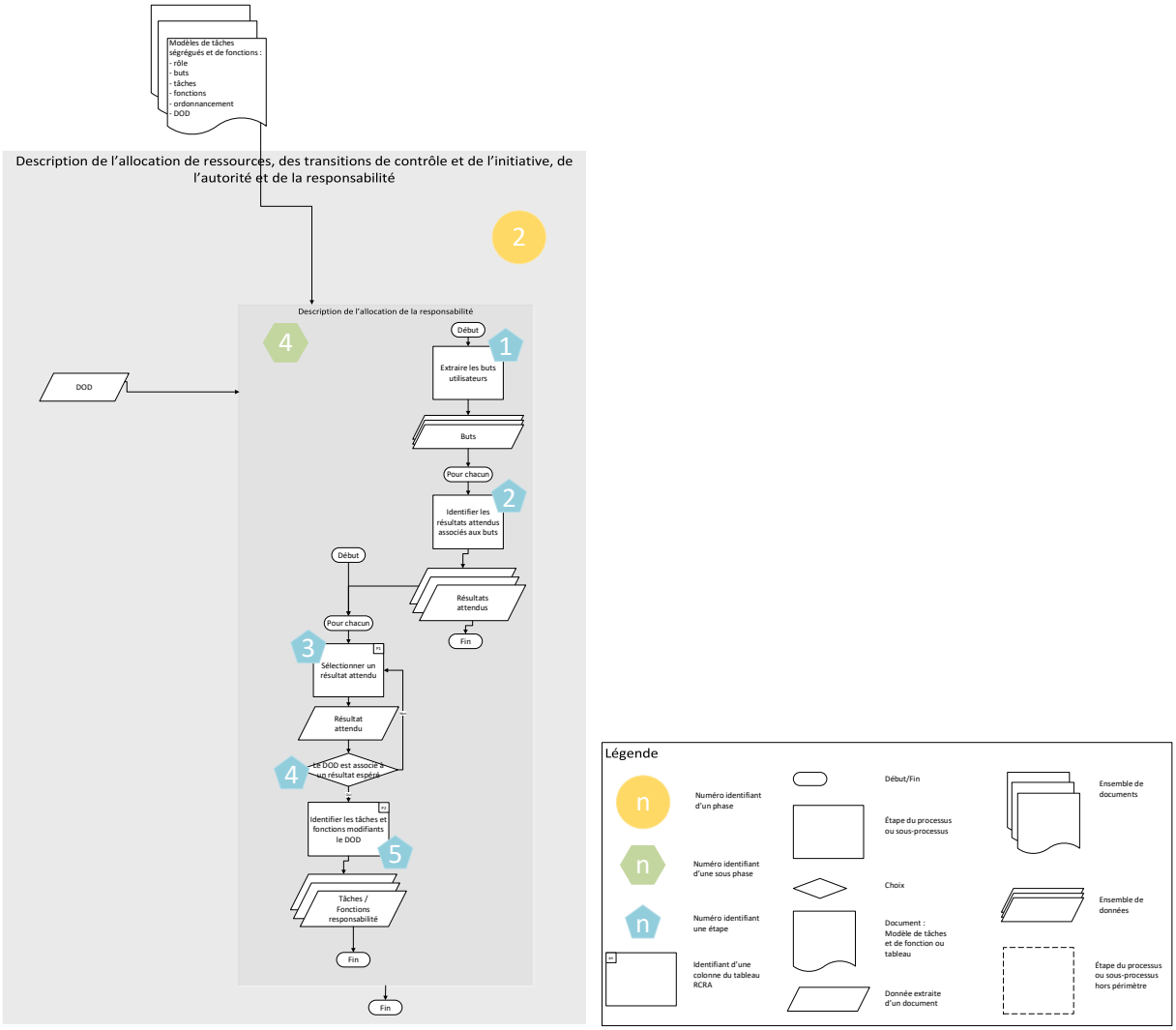


Figure 79. Vue détaillée de la sous-phase de description de la responsabilité avec les étapes explicitées numérotées de 1 à 6 dans des pastilles bleues.

Afin de décrire l'allocation de la responsabilité sur notre exemple, nous allons appliquer le processus « Description de l'allocation de la responsabilité » numéroté 4 dans la pastille verte de la Figure 79. Les étapes décrites sont numérotées dans les pastilles bleues de la Figure 79.

1. Dans un premier temps, nous extrayons les buts décrits dans les modèles de tâches ségrégués et de fonctions des différents rôles : « Démarrer le jeu » (Joueur), « Gagner le jeu des 15 » (*Leader*), « Gagner le jeu des 15 » (*Challenger*) et « Soutenir la partie dans le respect des règles du jeu des 15 » (Arbitre) pour soutenir les buts des utilisateurs.
2. Nous identifions les résultats attendus associés à ces buts :
    a. « Posséder 3 jetons dont la somme est égale à 15 » pour le *Leader* et le *Challenger*,
    b. « Deux joueurs jouent au jeu des 15 », « un seul jeton *Leader* est sélectionné pendant le tour du *Leader* », « un seul jeton *Challenger* est sélectionné pendant

le tour du *Challenger* », « le *Challenger* et le *Leader* joue chacun leur tour » pour l'arbitre.

3. Parmi tous les résultats attendus identifiés, nous sélectionnons le premier de la liste « Posséder 3 jetons dont la somme est égale à 15 » du *Leader* associé au but « Gagner le jeu des 15 » du *Leader*. Nous remplissons la colonne P2 avec le résultat attendu « Posséder 3 jetons dont la somme est égale à 15 » du *Leader*.
4. Nous identifions que le DOD décrivant un élément de la couche interface « Widgets des Jetons disponibles » est associé à ce résultat attendu : la sélection des valeurs des jetons joués par le Leader et le Challenger s'effectue grâce à ces widgets.
5. Nous identifions que le *Leader* manipule le DOD « Widgets des Jetons disponibles » lorsqu'il réalise la tâche motrice coopérative « Sélection d'un jeton ». Le Leader sélectionne une valeur qui peut faire dévier le résultat attendu « Posséder 3 jetons dont la somme est égale à 15 ». La tâche motrice coopérative « Sélection d'un jeton » du *Leader* est à responsabilité et est indiquée dans la colonne P1 du Tableau 27.

Le résultat de ces étapes est la partie du tableau RCRA présentée dans la première ligne surlignée en bleu du Tableau 27. Le Tableau 27 présente le résultat pour la description de l'allocation de la responsabilité à la fin de toutes les itérations de la phase 2.

Tableau 27. Colonnes de la partie « Allocation de la responsabilité » du tableau RCRA complétées durant les étapes décrites ci-dessus.

Allocation de la responsabilité	
Tâche/ Fonction P1	Résultat attendu possiblement impacté P2
Leader : Sélection d'un jeton	Posséder 3 jetons dont la somme est égale à 15 (*Leader*)
Tâche motrice Leader : « Sélectionner un jeton »	Leader : « Posséder 3 jetons dont la somme est égale à 15 »
Tâche motrice Challenger : « Sélectionner un jeton »	Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Initialiser les jetons disponibles »	Arbitre : « Un seul jeton est sélection chaque tour »
Fonction de sortie de l'arbitre : « Afficher les jetons disponibles activés »	
Fonction de l'arbitre : « Désactiver les jetons disponibles »	
Fonction de sortie de l'arbitre : « Retirer le jeton sélectionné des jetons disponibles et afficher les jetons disponibles désactivés »	
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »

Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Leader »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Challenger »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Arbitre : « Les joueurs jouent chacun leur tours »
Fonction de sortie de l'arbitre : « Afficher le bouton fin de tour désactivé »	
Fonction de l'arbitre : « Activer le bouton fin de tour »	
Fonction de sortie : « Afficher le bouton fin de tour disponible »	
Tâche motrice du Leader : « Mettre fin au tour »	
Tâche motrice du Challenger : « Mettre fin au tour »	
Fonction de sortie de l'arbitre : « Afficher le joueur courant »	
Fonction de l'arbitre : « Initialiser le joueur courant »	
Fonction de l'arbitre : « Changer le joueur courant »	
Fonction de l'arbitre : « Allouer les rôles aléatoirement »	Arbitre : « Le jeu est joué avec 2 joueurs »
Fonction de sortie de l'arbitre : « Afficher les rôles des joueurs »	

2.2.1 Description de l'allocation de l'autorité

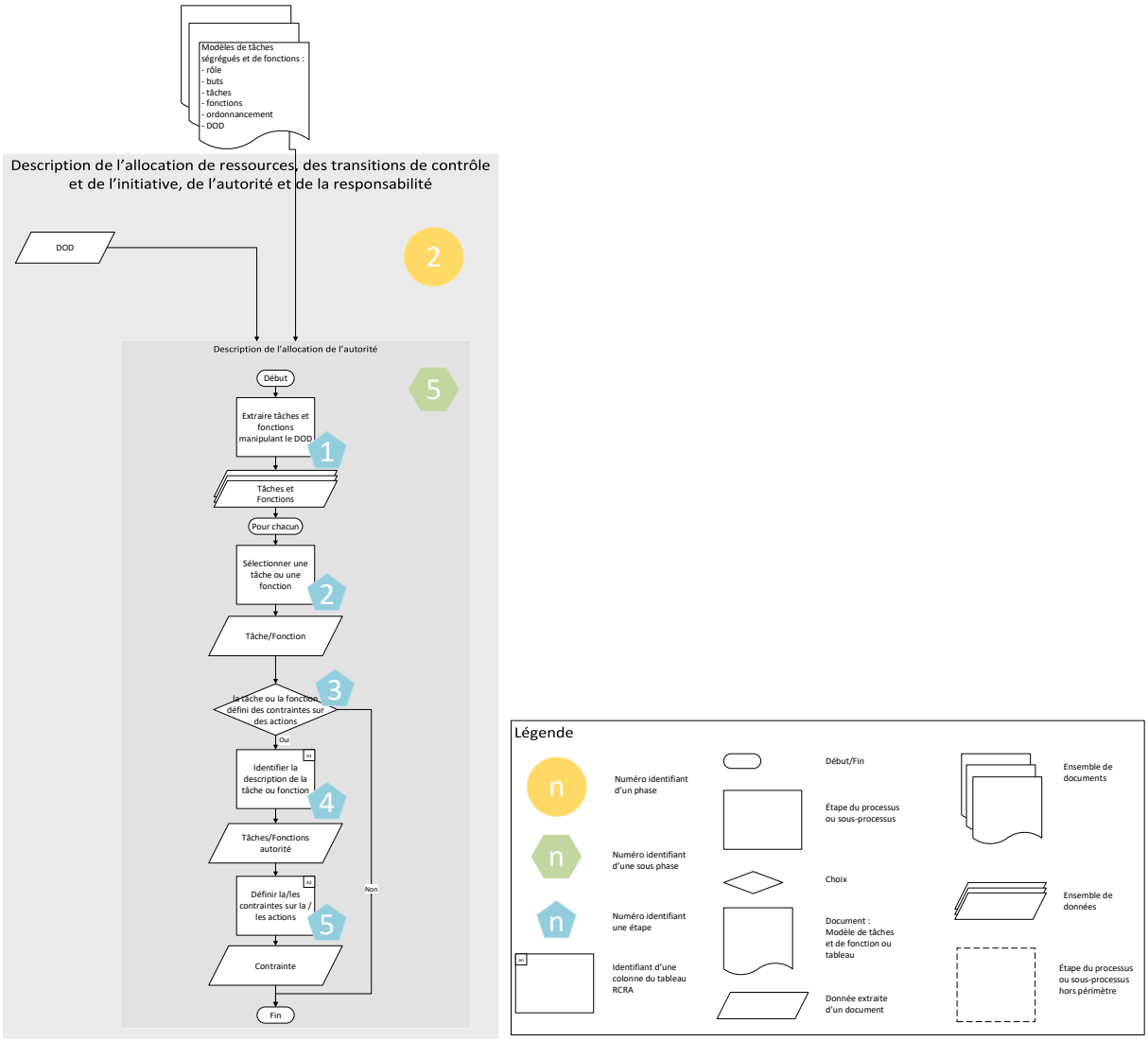


Figure 80. Vue détaillée de la sous-phase de description de l'autorité avec les étapes explicitées numérotées de 1 à 5 dans des pastilles bleues.

Afin de décrire l'allocation de l'autorité sur notre exemple, nous allons appliquer la sous-phase « Description de l'allocation de l'autorité » numéroté 5 dans la pastille verte de la Figure 80. Les étapes parcourues dans notre exemple sont numérotées dans les pastilles bleues de la Figure 80.

1. Parmi toutes les tâches et les fonctions des modèles de tâches ségrégués et de fonctions, nous extrayons les tâches et fonctions manipulant le DOD décrivant un élément de la couche interface « Widgets des Jetons disponibles ».
2. Parmi toutes les tâches et fonctions extraites, nous sélectionnons la fonction de l'arbitre « désactiver les jetons disponibles ». Nous décrirons les prochaines étapes avec cette fonction.
3. Nous identifions que cette fonction pose des contraintes pour la sélection des jetons par les joueurs.

4. Nous remplissons la colonne notée A1 du Tableau 28 avec la fonction de l'arbitre « Désactiver les jetons disponibles ».
5. Nous définissons que la fonction de l'arbitre « Désactiver les jetons disponibles » défini que la sélection d'un jeton n'est plus possible à partir de l'exécution de cette fonction. Nous remplissons la colonne notée A2 du Tableau 28 avec la contrainte posée par la fonction de l'arbitre « Désactiver les jetons disponibles » : « la sélection d'un jeton n'est plus possible à partir de l'exécution de cette fonction ».

Le résultat de ces étapes est la partie du tableau RCRA présentée dans la première ligne surlignée en bleu du Tableau 28. Le Tableau 28 présente le résultat pour la description de l'allocation de la responsabilité à la fin de toutes les itérations de la phase 2.

Tableau 28. Colonnes de la partie « Allocation des ressources » du tableau RCRA complétées durant les étapes décrites ci-dessus.

Allocation de l'autorité	
Tâche/Fonction A1	Contrainte A2
Arbitre : Désactiver les jetons disponibles	La sélection d'un jeton n'est plus possible à partir de l'exécution de cette fonction
Fonction de l'arbitre : « Activer les jetons disponibles »	Autoriser la sélection d'un jeton
Fonction de l'arbitre : Initialiser les jetons disponibles	Presser le widget pour sélectionner la valeur du jeton sélectionné
Tâche motrice du Leader : « Sélectionner le jeton »	Effectuer la sélection
Tâche motrice du Challenger : « Sélectionner le jeton »	Effectuer la sélection
Tâche de décision du Leader : « Choisir un jeton »	Valeur du jeton à sélectionner
Tâche de décision du Challenger : « Choisir un jeton »	Valeur du jeton à sélectionner
Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Empêcher la pression du bouton fin de tour
Fonction de l'arbitre : « Activer le bouton fin de tour »	Autoriser la pression du bouton fin de tour
Fonction de l'arbitre : « Initialiser le bouton fin de tour »	Bouton à presser pour mettre fin au tour
Tâche motrice du Leader : « Mettre fin au tour »	Mettre fin au tour
Tâche motrice du Challenger : « Mettre fin au tour »	Mettre fin au tour
Fonction de l'arbitre : « Initialiser le joueur courant »	Définir valeur premier joueur
Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Challenger
Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Leader
Tâche cognitive de décision du Leader : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Tâche cognitive de décision du Challenger : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Initialiser le bouton rejouer	Presser le bouton pour rejouer

Activer le bouton rejouer	Autoriser pression bouton rejouer
Tâche motrice du Leader : « Rejouer »	Lancer une nouvelle partie
Tâche motrice du Challenger : « Rejouer »	Lancer une nouvelle partie
Tâche cognitive de décision du Leader : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche cognitive de décision du Challenger : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche de groupe du Leader et du Challenger : « Les joueurs se mettent d'accord pour rejouer »	Défini qui presse le bouton pour rejouer
Tâche cognitive d'analyse du Leader : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Tâche cognitive d'analyse du Challenger : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Fonction de l'Arbitre : « Initialiser le bouton lancer partie »	Presser le bouton lancer partie pour lancer la partie
Fonction de l'Arbitre : « Activer le bouton lancer partie »	Activer bouton lancer partie
Tâche motrice du Joueur : « Lancer la partie »	Lancer la partie
Tâche motrice parler du joueur : « Annoncer la partie du jeu des 15 »	Définir quel joueur lance la partie
Fonction de l'arbitre : « Allouer les rôles aléatoirement aux joueurs »	Définir la valeur du rôle des joueurs

2.3 Phase de modélisation et d'analyse RCRAFT

A la fin des étapes décrites, nous avons produit la première ligne du tableau RCRA pour l'exemple du jeu des 15. Dans chaque sous-phases, nous n'avons pas décrit toutes les itérations nécessaires pour le DOD « Widgets des Jetons Disponibles ». L'application complète du processus permet de produire le tableau RCRA du jeu des 15 dont les différentes colonnes sont présentées dans la section ci-avant.

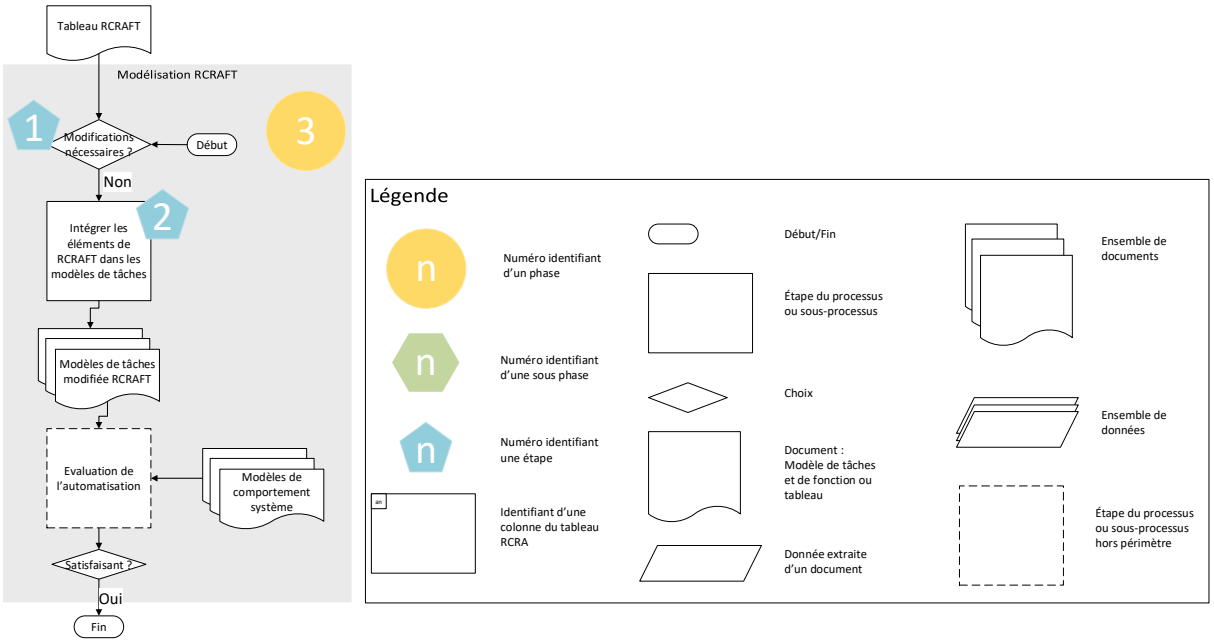


Figure 81. Vue détaillée de la phase de la modélisation RCRAFT avec les étapes explicitées numérotées de 1 à 2 dans des pastilles bleues

La phase 3 du processus nous permet de modéliser de manière explicite dans les modèles de tâches ségrégués et de fonctions l'allocation de ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre les rôles Joueur, *Leader*, *Challenger* et Arbitre décrite dans le tableau RCRA à l'issue de la phase 2. La Figure 81 présente la vue détaillée de la phase 3. Les étapes parcourues sont numérotées dans les pastilles bleues :

1. Lors de la phase 2, nous n'avons pas relevé de problème de modélisation. Il n'y a pas de modification des modèles nécessaire.
2. Nous intégrons les éléments RCRAFT dans les modèles de tâches et de fonctions grâce aux éléments de notation présentés au Chapitre 5.3 :
    - Nous indiquons que le DOD « Widget des Jetons disponibles » permet de partager les ressources « Jetons disponibles » et « Jeton sélectionné ».
    - Nous indiquons le l'arbitre initie une transition de contrôle : l'arbitre relâche le contrôle pour la sélection des jetons avec la fonction « Afficher les jetons disponibles » qui manipule le DOD « Widgets des jetons disponibles ».
    - Nous indiquons que la fonction « Désactiver les jetons disponibles » exprime l'autorité de l'arbitre de pouvoir empêcher les joueurs de sélectionner un jeton.
    - Nous indiquons que la tâche motrice coopérative « Sélection d'un jeton » est à responsabilité et peut faire dévier le résultat attendu du *Leader* « Posséder 3 jetons dont la somme est égale à 15 ».

Les Figure 82, Figure 83, Figure 84 et Figure 85 sont les modèles de tâches ségrégués et de fonctions respectivement du Joueur, du Leader, du Challenger et de l'Arbitre produits lors de la phase 1 enrichis avec les éléments de notation RCRAFT.

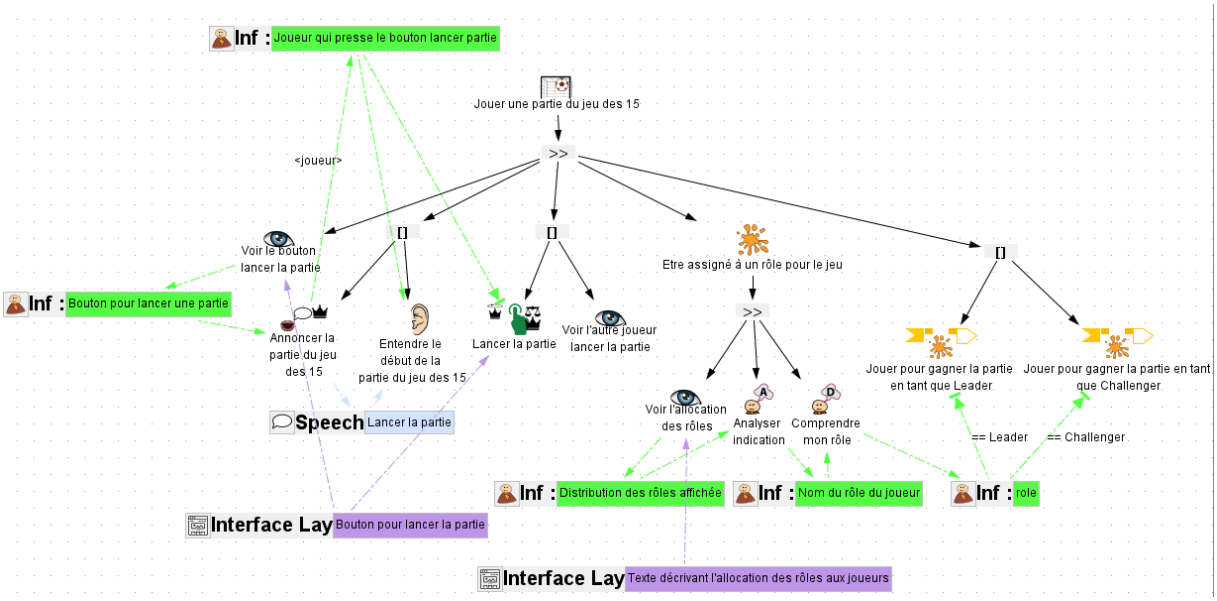


Figure 82. Modèle de tâches ségrégué du rôle Joueur pour jouer une partie du jeu des 15.

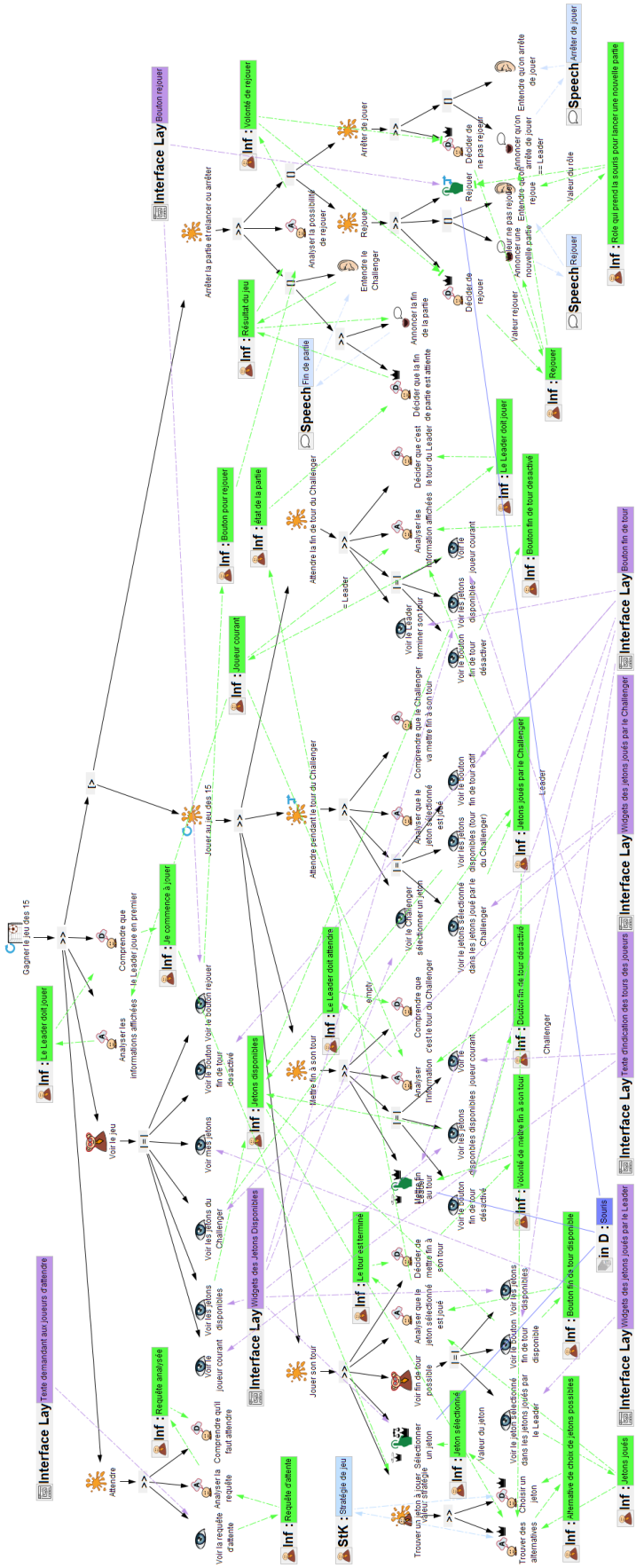


Figure 83. Modèle de tâches ségrégué du Leader pour gagner le jeu des 15.

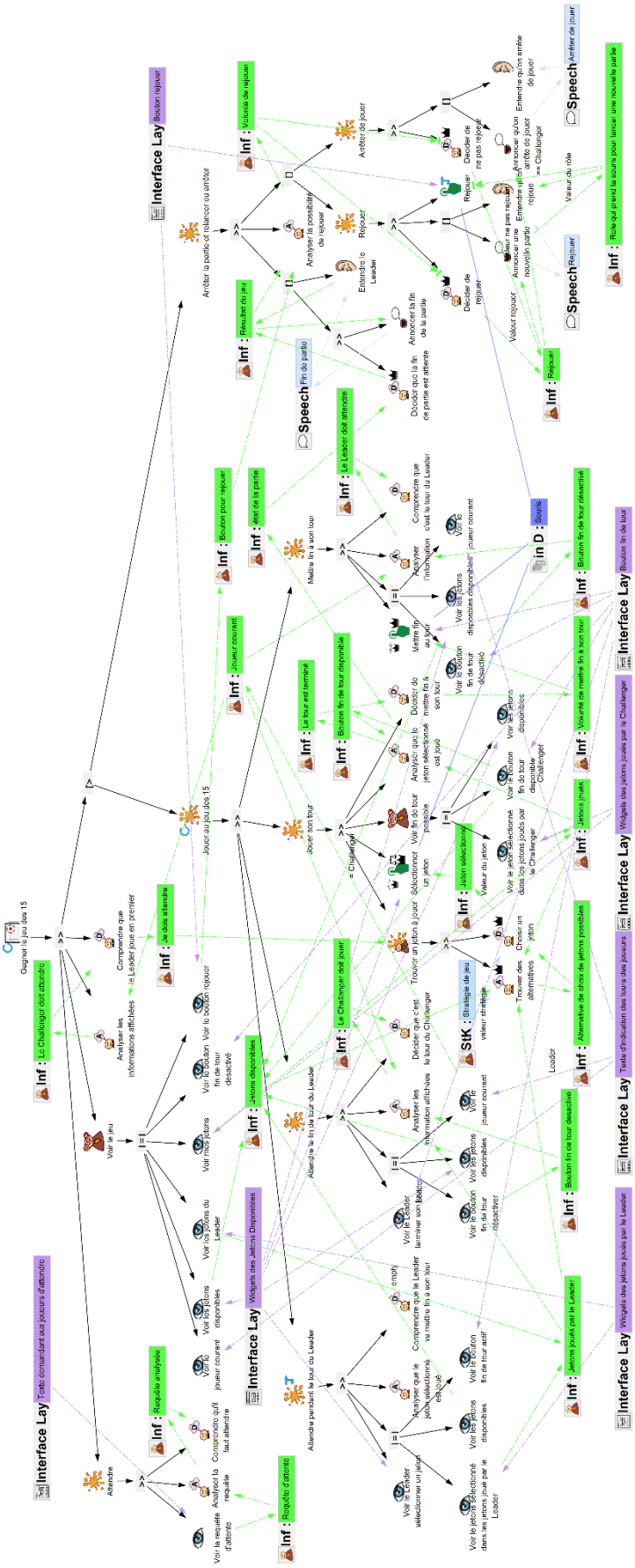


Figure 84. Modèle de tâches ségrégué du Challenger pour gagner le jeu des 15.

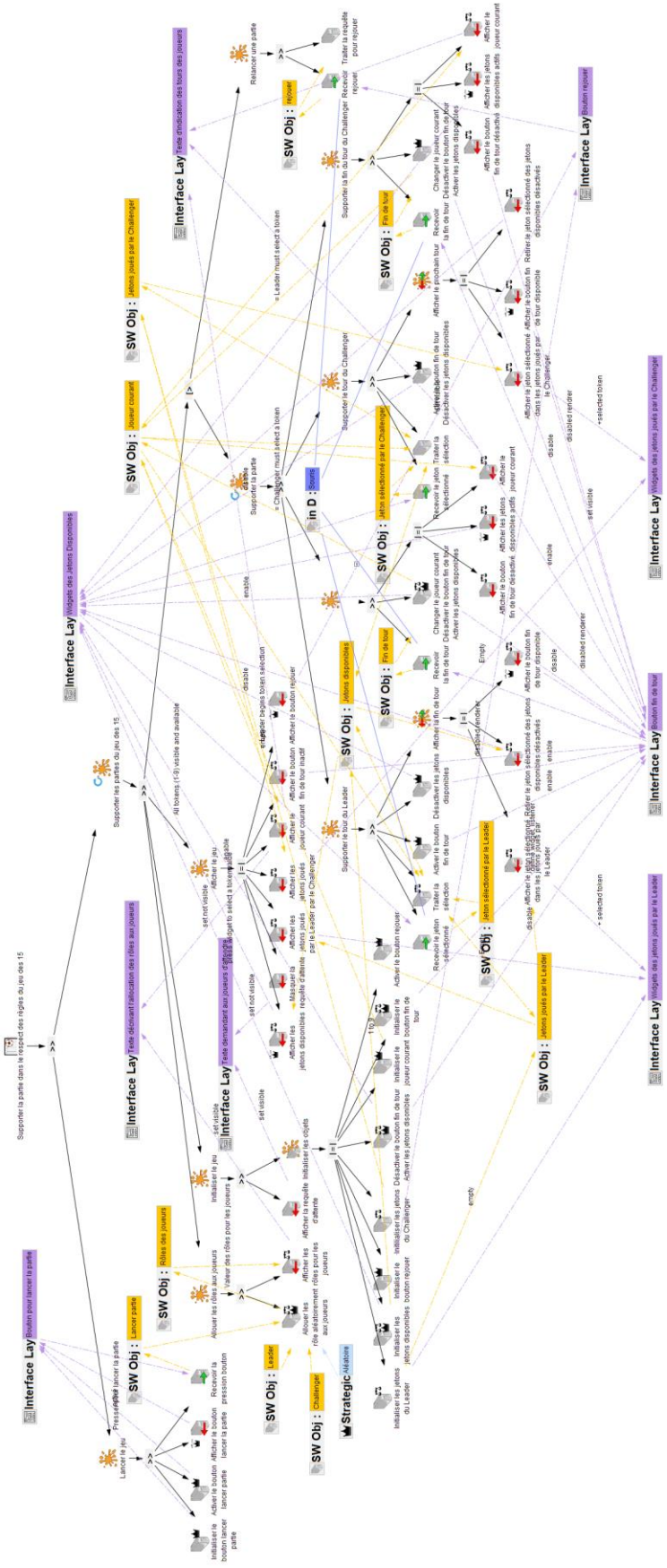


Figure 85. Modèle de fonctions de l'arbitre pour soutenir la partie dans le respect des règles du jeu des 15.

3 Conclusion

Ce chapitre décrit le processus RCRAFT d'analyse systématique de l'allocation RCRAFT. Dans ce chapitre nous avons présenté le processus RCRAFT de manière abstraite à travers la vue haut-niveau du processus présenté en Figure 62. Ensuite, nous avons détaillé chaque étape des trois phases du processus. Ce processus permet de produire et d'analyser les modèles de tâches ségrégués et de fonctions produits lors d'une première phase les parcourant DOD par DOD. Pour chacun des DODs, la ressource que le DOD représente est identifiée, ainsi que les tâches et fonctions qui les manipulent. L'identification et la représentation des DODs est une étape essentielle car elle fournit un support à l'identification et à l'analyse de l'allocation transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité.

Lors de la phase 3 du processus RCRAFT, les allocations RCRAFT identifiées sont représentées dans des modèles de tâches ségrégués et de fonctions. Nous proposons lors de cette phase 3 du processus RCRAFT une étape d'évaluation de l'automatisation par rapport aux objectifs de conception. Nous proposons de fournir en entrée de cette étape d'évaluation des propriétés de l'automatisation la description explicite de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre les utilisateurs et les systèmes contenant de l'automatisation. Nous montrons dans le Chapitre 8 et le Chapitre 9 qu'une telle description permet d'identifier précisément, au niveau des tâches et des fonctions, les différences entre plusieurs solutions de conception de l'automatisation en termes d'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité.

L'outil HAMSTERS a été étendu pour mettre en œuvre ce processus. Ces extensions sont présentées au chapitre suivant.

Chapitre 7. Extension de l'outil Hamsters pour l'édition des modèles tâches dans le cadre de l'analyse de l'automatisation

Ce chapitre présente les extensions apportées à l'outil HAMSTERS afin de soutenir le processus RCRAFT. Les extensions apportées permettent la description des protocoles de coopération et de chacun des éléments RCRAFT.

La première section présente les extensions apportées à l'outil HAMSTERS pour la représentation des protocoles de coopération et de ses caractéristiques : les ressources partagées, la temporalité, la cardinalité et la localisation de la coopération.

La seconde section présente comment nous exploitons les modèles de tâches ségrégués et les modèles de fonctions pour la représentation de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité.

La troisième section conclut ce chapitre.

1 Extension de l'outil HAMSTERS pour la représentation de la coopération

Cette section présente les extensions apportées à l'outil HAMSTERS afin de représenter la coopération tel que décrit au Chapitre 5.2. L'identifiant unique est automatiquement généré par l'outil.

1.1 Ressources du protocole de coopération

L'outil HAMSTERS permet de représenter les instances de ressources émises ou reçues lors de la coopération. Dans Figure 86 et Figure 87, l'objet physique « Carte de crédit », le périphérique d'entrée « Fente carte » et la couche interface « Texte demandant de récupérer la carte » représente les moyens par lesquels le client et le distributeur coopèrent.

Nous reprenons l'exemple du retrait d'argent avec le distributeur de billet présenté au Chapitre 5.1.1.1 utilisé pour l'illustration des modèles de tâches ségrégués et de fonctions et de la description de la coopération. Pour le but consistant à retirer de l'argent au distributeur automatique de billets, lorsque le client récupère sa carte de crédit (tâche motrice coopérative « Prendre la carte » de la Figure 86) et le distributeur relâche la carte de crédit (fonction d'entrée/sortie coopérative « Relâcher carte » de la Figure 87), les instances de ressources émises et reçues lors de la coopération sont : l'objet physique carte de crédit et le périphérique d'entrée/sortie fente du distributeur de billet.

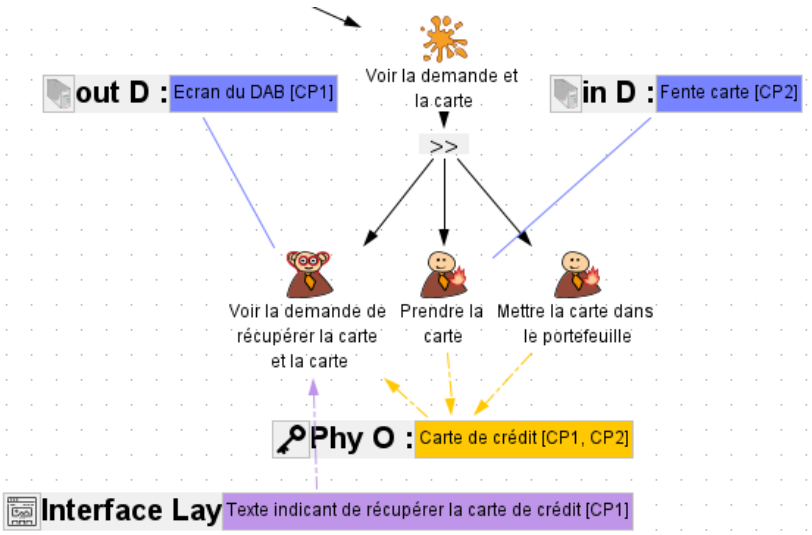


Figure 86. Tâche "Finaliser de retirer l'argent" du rôle Client.

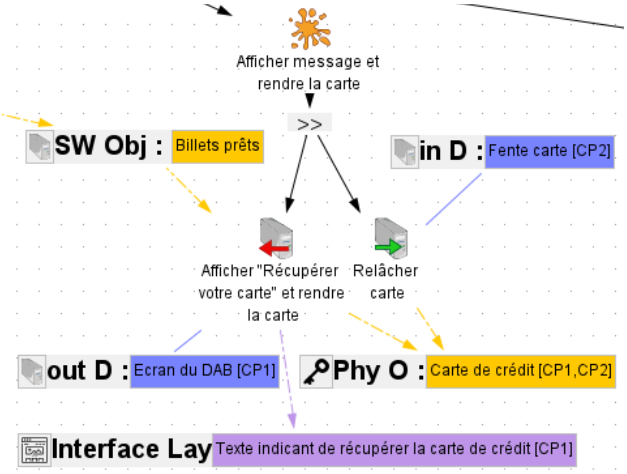


Figure 87. Fonction "Finaliser la transaction" du rôle distributeur de billet.

1.2 Protocole de coopération synchrone ou asynchrone

L'outil HAMSTERS permet de décrire si un protocole de coopération est synchrone ou asynchrone (voir Chapitre 5.2.2.2) dans les propriétés du protocole de coopération sélectionné (voir Figure 88).

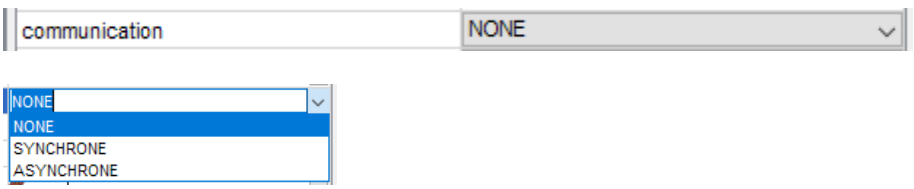


Figure 88. Propriété de communication du protocole de coopération.

1.3 Localisation de la coopération

L'outil HAMSTERS permet de décrire si un protocole de coopération décrit une coopération locale ou distante (voir Chapitre 5.2.2.4) dans les propriétés du protocole de coopération sélectionné (voir Figure 89).

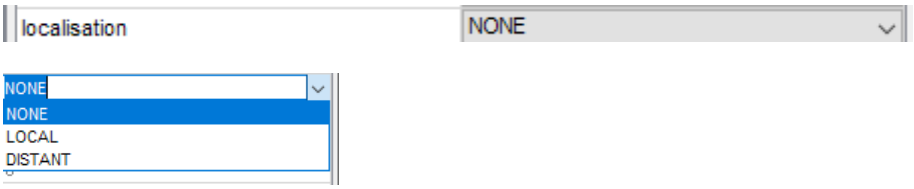


Figure 89. Propriété de localisation du protocole de coopération.

1.4 Cardinalité du protocole de coopération

L'outil HAMSTERS permet de décrire la cardinalité du protocole de coopération (élément présenté dans le Chapitre 5.2.2.3). La cardinalité peut être :

- Point à point (entre 2 instances de rôle : unicast)
- Broadcast (entre tous rôles ou instances d'un même rôle)
- Multicast (entre un groupe restreint d'acteurs)
- Anycast (entre n'importe quel acteur).

Le panneau d'édition des propriétés permet de modifier la cardinalité du protocole de coopération sélectionné. Également, le nombre d'acteurs impliqués dans la coopération peut être spécifié dans les propriétés de cardinalité du protocole de coopération sélectionné. Cette propriété s'édite de la même manière que l'édition de la localisation de la coopération (voir Figure 89).

1.5 Type de coopération

L'outil HAMSTERS permet de décrire le type de coopération du protocole de coopération (élément présenté dans le Chapitre 5.2.2.1). Le type de coopération peut être :

- Communication
- Production
- Coordination

Cette propriété s'édite de la même manière que l'édition de la localisation de la coopération (voir Figure 89).

2 Extension de l'outil HAMSTERS pour la représentation des éléments RCRAFT

Cette section présente les extensions apportées à l'outil HAMSTERS afin de représenter les éléments RCRAFT dans les modèles de tâches ségrégués et de fonctions. La description de l'allocation des ressources est déjà possible avec l'outil HAMSTERS par la représentation de DOD : les informations, données, périphériques et objets manipulés par les utilisateurs et les systèmes contenant de l'automatisation. Nous présentons les extensions apportées pour la représentation des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité.

2.1 Exploitation des modèles pour la représentation de l'Allocation des transitions de contrôle et de l'initiative

Lorsque les tâches et les fonctions des rôles des différents acteurs initiant une transition de contrôle ont été identifiées (voir Chapitre 4.5), il est possible d'enrichir les modèles de fonctions et de tâches HAMSTERS afin de le décrire.

Par exemple, la Figure 91 présente l'édition de la tâche motrice du Leader « Sélectionner un jeton ». Lorsque le Leader sélectionne un jeton (tâche motrice « Sélectionner un jeton »), il initie une transition de contrôle. L'arbitre reprend le contrôle des jetons disponible : il prend en compte sa sélection, retire le jeton des jetons disponibles et l'affiche dans les jetons joués par le Leader. Il est possible d'indiquer directement dans le modèle de tâches ségrégué du Leader qu'il initie une transition de contrôle par cette tâche en sélectionnant le bouton radio *transition de contrôle* dans les propriétés de la tâche. Les propriétés de la tâche sont accessibles en cliquant sur la tâche concernée dans le modèle de tâches ségrégué.

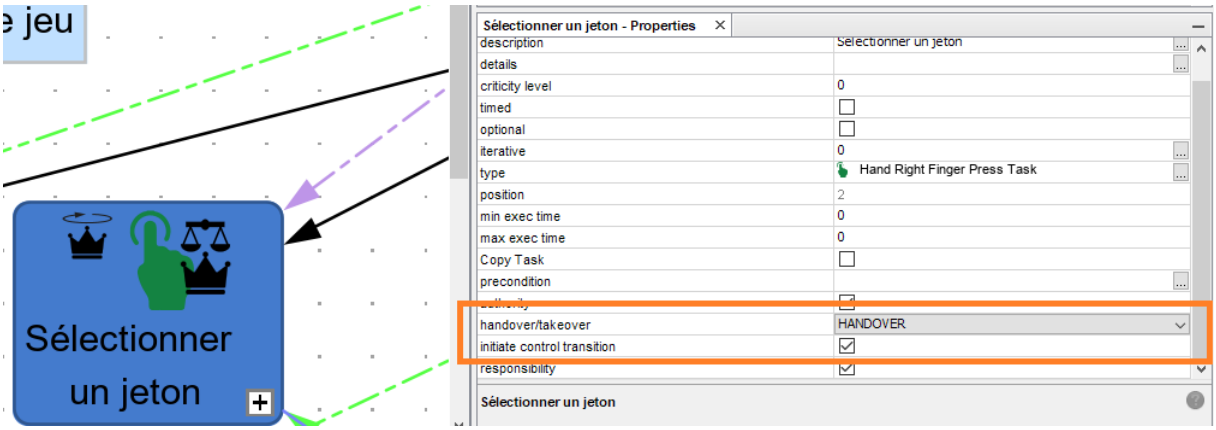


Figure 90. Edition de la transition de contrôle initiée par la tâche motrice "Sélectionner un jeton" du Leader.

2.2 Exploitation des modèles pour la représentation de l'allocation de la Responsabilité

Lorsque les tâches et les fonctions des rôles des différents acteurs pouvant faire dévier un des résultats attendus ont été identifiées (voir Chapitre 4.6), il est possible d'enrichir les modèles de fonctions et de tâches ségrégués HAMSTERS afin de le décrire.

Par exemple, la Figure 91 présente l'édition de la tâche motrice du Leader « Sélectionner un jeton ». Le leader a pour but de gagner la partie du jeu des 15. Pour cela, son résultat attendu est de « Posséder 3 jetons dont la somme est égale à 15 » à la fin de la partie. La tâche motrice « Sélectionner un jeton » définie la valeur du jeton sélectionné pour le tour par le biais de l'objet de coopération couche d'interface « Widget des Jetons Disponibles » du protocole de coopération pour sélectionner un jeton. L'utilisation de cet objet de coopération peut faire dévier le résultat attendu car les jetons joués par le Leader sont définis en sélectionnant un jeton parmi ceux présenté dans la couche d'interface « Widget des Jetons Disponibles ». La tâche motrice « Sélectionner un jeton » est à responsabilité comme défini au Chapitre 4.6. Il est possible de l'indiquer directement dans le modèle de tâches ségrégué en sélectionnant le bouton

radio *responsability* dans les propriétés de la tâche. Les propriétés de la tâche sont accessibles en cliquant sur la tâche dans le modèle de tâches ségrégué.

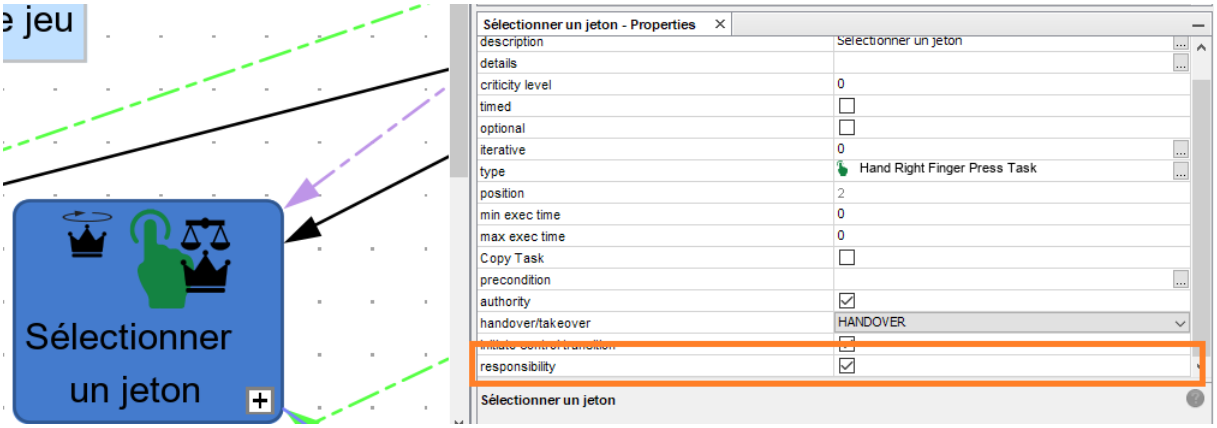


Figure 91. Edition de la responsabilité de la tâche motrice "Sélectionner un jeton" du Leader.

2.3 Exploitation des modèles pour la représentation de l'allocation de l'Autorité

Lorsque les tâches et les fonctions des rôles des différents acteurs contraignant les tâches et fonctions ou réalisant les tâches et fonctions ont été identifiées (voir Chapitre 4.4), il est possible d'enrichir les modèles de fonctions et de tâches ségrégués HAMSTERS afin de le décrire.

Par exemple, la Figure 92 présente l'édition de la tâche cognitive « Trouver des alternatives » du Leader. La tâche cognitive de décision « Choisir un jeton » défini la valeur du jeton à sélectionner pour le tour, c.-à-d. la tâche contraint la valeur du jeton à sélectionner. Cette tâche est à autorité comme défini au Chapitre 4.4. Il est possible de l'indiquer directement dans le modèle de tâches en sélectionnant le bouton radio *authority* dans les propriétés de la tâche. Les propriétés de la tâche sont accessibles en cliquant sur la tâche dans le modèle de tâche.

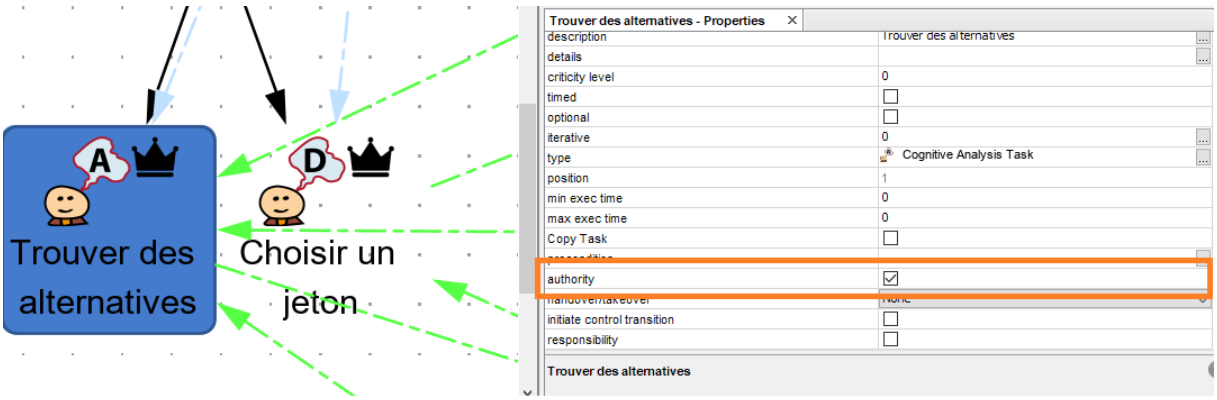


Figure 92. Edition de l'autorité de la tâche cognitive "Trouver des alternatives" du Leader.

Il est à noter que la représentation de l'autorité peut être automatisée pour certaines tâches et fonctions. Notamment, les fonctions systèmes activant, désactivant et définissant les interactions à réaliser sur les éléments de l'interface contraindront toujours les tâches et fonctions de l'utilisateur et sont donc à autorité.

3 Conclusion

Dans cette section, nous avons présentés les extensions apportées à l'outil HAMSTERS pour soutenir le processus RCRAFT. Ces extensions permettent la description de la coopération à travers les protocoles de coopération, de l'allocation des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Ces extensions permettent de soutenir les phases 1 (modélisation de tâches explicitant l'allocation des fonctions et des tâches) et 3 (modélisation et analyse RCRAFT) du processus RCRAFT.

Partie III. Validation de la contribution

Chapitre 8. Validation de l'expressivité de RCRAFT sur des variantes du jeu des 15

Dans ce chapitre, nous décrivons et comparons différentes versions du jeu des 15 en termes d'allocation RCRAFT avec la version du jeu des 15 utilisée à des fins d'illustration dans cette thèse. Ces descriptions et analyses nous permettent de caractériser en termes d'allocation RCRAFT les différences introduites par différents types d'automatisation du système permettant de jouer au jeu des 15.

Dans la première section, nous présentons la comparaison en termes d'allocation RCRAFT d'une version du jeu des 15 avec un temps limité pour jouer son tour avec la version du jeu des 15 présentée au Chapitre 4.1.

Dans la seconde section, nous présentons la comparaison en termes d'allocation RCRAFT d'une version du jeu des 15 proposant des recommandations aux joueurs avec la version du jeu des 15 présentée au Chapitre 4.1.

Dans la troisième section, nous concluons ce chapitre en présentant une synthèse des comparaisons réalisées dans ce chapitre.

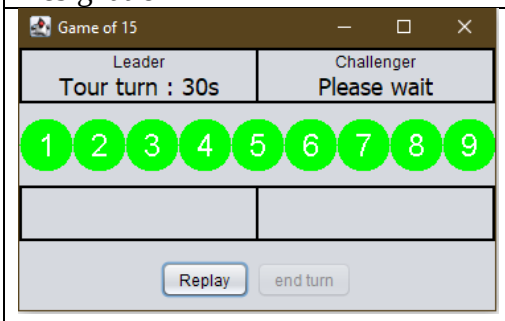
1 Version du jeu des 15 avec un temps limité pour jouer son tour

Dans cette première section, nous présentons la description de l'allocation RCRAFT sur une version du jeu des 15 avec un temps limité pour jouer son tour. Dans la première sous-section, nous présentons la version du jeu des 15 avec un temps limité pour jouer son tour. Dans la seconde sous-section, nous décrivons l'allocation RCRAFT produite suite à l'application du processus RCRAFT pour la version du jeu des 15 avec un temps limité pour jouer son tour. Enfin, nous présentons la comparaison en termes d'allocation RCRAFT du jeu des 15 avec un temps limité pour jouer son tour et la version du jeu des 15 présentée au Chapitre 4.1.

1.1 Présentation du jeu des 15 avec un temps limité pour jouer son tour

Cette version du jeu des 15 se joue à deux joueurs. Ils disposent de jetons représentant les chiffres de 1 à 9. Chacun leur tour, ils doivent choisir un jeton en moins de 30 secondes, qui devient indisponible pour le joueur adverse. Si le joueur ne sélectionne pas un jeton avant que les 30 secondes ne soient écoulées, le système sélectionne un jeton au hasard pour le joueur. Lorsque l'un des deux joueurs possède trois jetons dont la somme est égale à 15, la partie est terminée et ce joueur gagne. Ici, nous considérons que deux joueurs humains jouent l'un contre l'autre à travers un système permettant de jouer au jeu des 15. Le système indique les tours de jeux, permet de mettre fin à un tour de jeu, affiche les jetons disponibles et joués par les deux joueurs et affiche le temps restant pour le tour. Les joueurs peuvent choisir de rejouer une partie à tout moment. Le Tableau 29 résume cette version du jeu des 15.

Tableau 29. Présentation du jeu des 15 avec un temps limité pour jouer son tour.

Désignation	Description
 Jeu des 15 avec un temps limité pour jouer son tour	Les jetons disponibles, les jetons joués par le système et les jetons joués par le joueur humain sont affichés par le système. Le tour de jeu se termine au bout de 30 secondes pour le joueur. Un jeton est sélectionné par le système si les joueurs n'en ont pas sélectionné au bout de 30 secondes. Les joueurs peuvent choisir de rejouer une partie à tout moment.

1.2 Description de l'allocation RCRAFT pour la version du jeu des 15 avec un temps limité pour jouer son tour

Dans cette sous-section, nous décrivons l'allocation RCRAFT produite suite à l'application du processus RCRAFT pour la version du jeu des 15 avec un temps limité pour jouer son tour. Nous ne présentons seulement les artefacts produits après l'application du processus : les modèles de tâches ségrégués et les modèles de fonctions, le tableau RCRA et les modèles de tâches ségrégués et les modèles de fonctions intégrant la description de l'allocation RCRA.

1.2.1 Description de l'allocation des tâches et des fonctions pour la version du jeu des 15 avec un temps limité pour jouer son tour

La Figure 93 présente un aperçu du modèle de tâches ségrégué du rôle Leader. La Figure 94 présente un aperçu du modèle de fonctions du rôle Arbitre. La description sous formes de modèles de tâches de l'allocation des tâches et des fonctions de chacun des rôles sont consultables en Annexes. Les principaux changements de cette version en termes d'allocation de fonctions et de tâches concernent l'interruption des tours de jeu et la définition d'un *timer* pour chaque tour de jeu.

Les tâches du rôle Joueur sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.

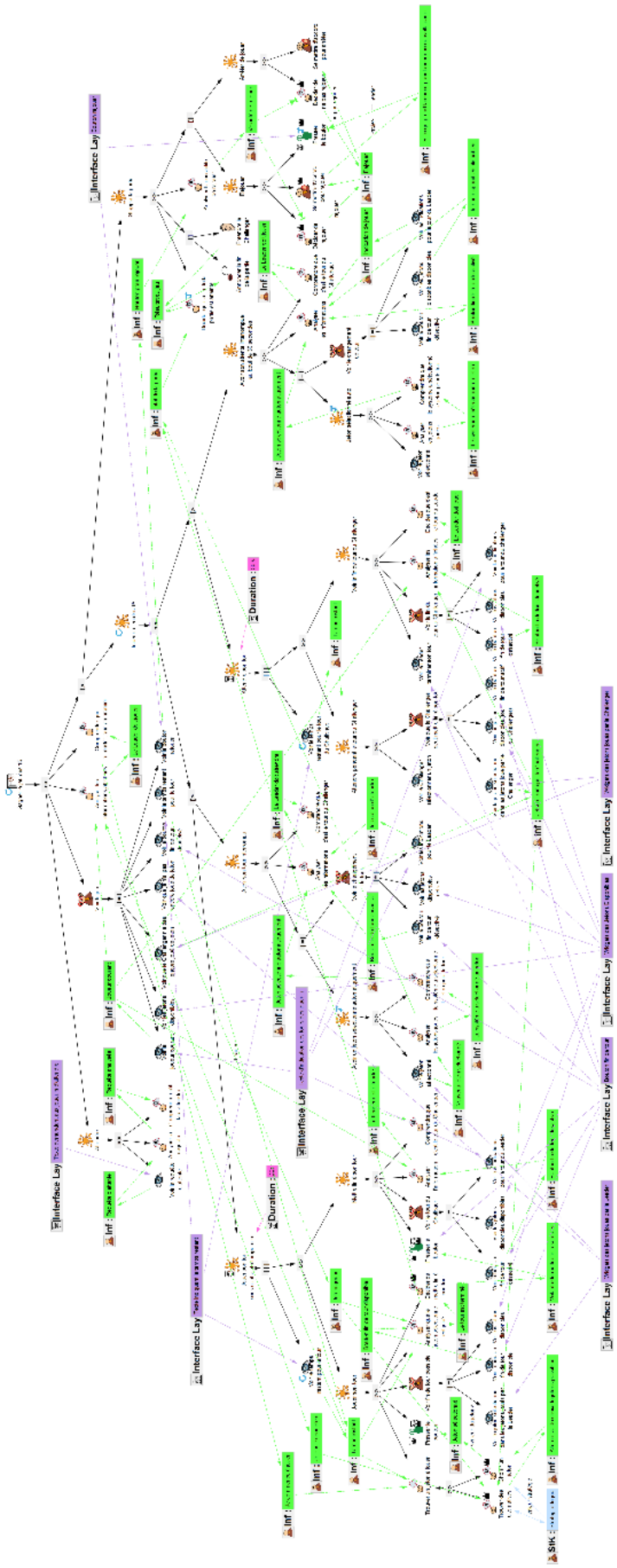


Figure 93. Aperçu du modèle de tâches du Leader pour gagner le jeu des 15 avec un temps limité pour jouer son tour

Les tâches allouées au Leader décrites dans la Figure 93 sont :

- Attendre (le début de la partie) dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
- Voir le jeu (dont une tâche perceptive coopérative « Voir le temps restant pour le tour »), analyser les informations affichées et comprendre que le Leader joue en premier.
- Jouer au jeu des 15 (tâche abstraite itérative)
    - Jouer son tour sous contrainte temporelle (tâche abstraite avec une durée maximale de 30 secondes)
        - Voir le temps restant pour le tour (tâche perceptive itérative)
        - Jouer son tour dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
        - Mettre fin à son tour dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.

      Cette tâche est interrompue si le Leader ne l'a pas terminée en 30 secondes.
    - Avoir son tour interrompu (tâche abstraite)
        - Avoir un jeton sélectionné automatiquement (tâche abstraite optionnelle)
            - Voir le jeton sélectionné (tâche perceptive coopérative)
            - Analyser les jetons joués (tâche cognitive d'analyse)
            - Comprendre que le système a sélectionné le jeton pour le tour (tâche cognitive de décision)
        - Voir le changement de tour (tâche perceptive)
            - Voir : le bouton fin de tour désactivé, les jetons disponibles activés, les indications pour le Leader (tâches perceptives coopératives)
        - Analyser les informations
        - Comprendre que c'est le tour du Challenger
    - Attendre son tour (tâche abstraite avec une durée maximale de 30 secondes)
        - Voir le temps restant pour le tour du Challenger (tâche perceptive itérative)
        - Attendre pendant le tour du Challenger (tâche abstraite)
        - Voir la fin de tour du Challenger (tâche abstraite)

      Cette tâche peut être interrompue si le Challenger n'a pas mis fin à son tour avant la fin des 30 secondes.
    - Avoir son attente interrompue au bout de 30 secondes (tâche abstraite)
- Stopper la partie dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.

Les tâches du Leader et du Challenger sont identiques à la différence que le Leader est le premier à jouer son tour.

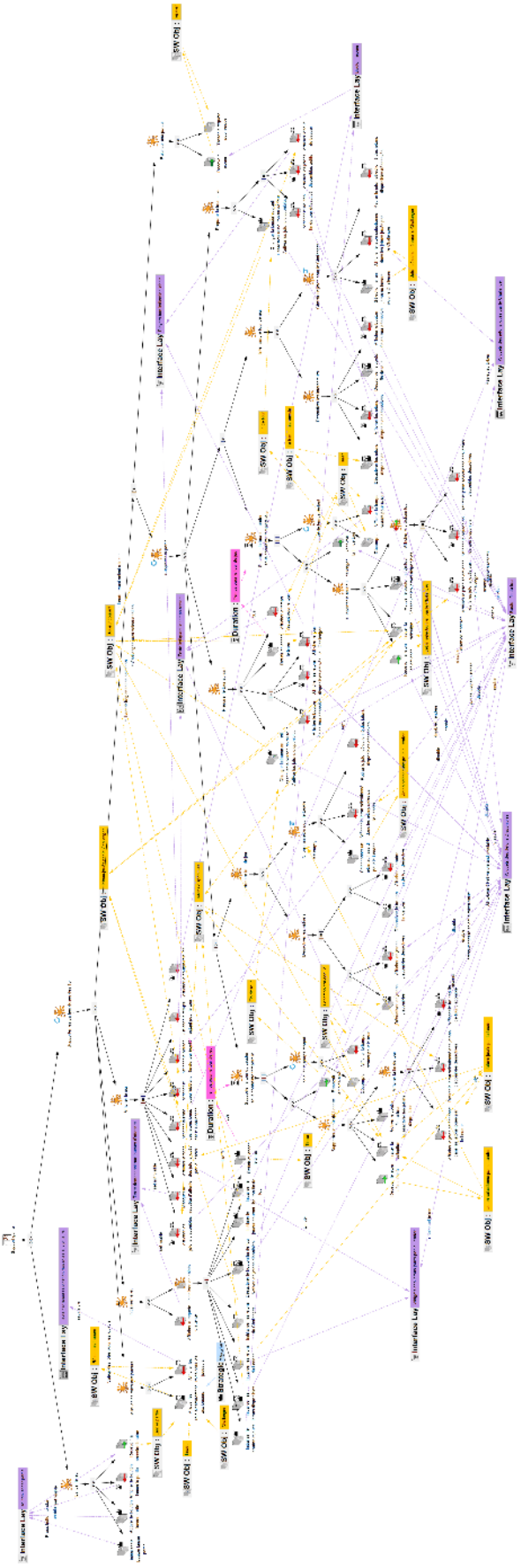


Figure 94. Aperçu du modèle de fonctions de l'Arbitre pour soutenir la partie du jeu des 15 avec un temps limité pour le tour des joueurs.

Les fonctions allouées à l'Arbitre décrites dans la Figure 94 sont :

- Lancer le jeu dont les fonctions sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
- Soutenir les parties du jeu des 15 (fonction abstraite itérative)
    - Allouer les rôles aux joueurs dont les fonctions sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
    - Initialiser le jeu (fonction abstraite)
        - Définir et lancer le timer (fonction)
        - Les autres fonctions sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
    - Afficher le jeu (fonction abstraite)
        - Afficher le temps restant pour le tour (fonction de sortie coopérative)
        - Les autres fonctions sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
    - Soutenir la partie
        - Soutenir le tour du Leader sous contrainte temporelle (fonction abstraite avec une durée maximale de 30 secondes)
            - Soutenir le tour du Leader (fonction abstraite)
                - Recevoir le jeton sélectionné (fonction d'entrée coopérative)
                - Traiter la sélection (fonction)
                - Activer le bouton fin de tour et désactiver les jetons disponibles (fonction)
                - Afficher la fin de tour (fonction abstraite)
                    - Afficher : le jeton sélectionné dans les jetons joués par le Leader, retirer le jeton sélectionné des jetons disponibles désactivés, le bouton fin de tour disponible (fonction de sortie coopératives)
            - Recevoir la fin de tour (tâche d'entrée coopérative)
            - Indiquer le temps restant (fonction abstraite itérative)
                - Décompter (fonction)
                - Afficher le temps restant pour le tour (fonction de sortie coopérative)
        - Interrompre le tour de jeu (fonction abstraite)
            - Désactiver les widgets (fonction abstraite)
                - Désactiver les jetons disponibles (fonction)
                - Afficher les jetons disponibles désactivés (fonction de sortie coopérative)
                - Désactiver le bouton fin de tour (fonction)

        - Afficher le bouton fin de tour désactivé (fonctions de sortie coopérative)
    - Choisir un jeton pour le Leader (fonctions abstraite optionnelle)
      Si l'objet logiciel « Jeton sélectionné par le Leader » est vide :
        - Sélectionner un jeton au hasard pour le Leader (fonction)
        - Afficher le jeton sélectionné dans les jetons joués par le Leader (fonction de sortie coopérative)
        - Retirer le jeton sélectionné des jetons disponibles désactivés (fonction de sortie coopérative)
  - Préparer le tour suivant (fonction abstraite)
    - Changer le joueur courant, désactiver le bouton fin de tour, activer les jetons disponibles (fonctions)
    - Afficher : le bouton fin de tour désactivé, les jetons disponibles activés, le tour du Challenger (fonctions de sortie coopératives)
    - Définir et lancer le timer (fonction)
    - Afficher le temps restant pour le tour (fonction de sortie coopérative)
  - Les mêmes fonctions sont exécutées par l'Arbitre pour le tour du Challenger
- Relancer une partie (fonction abstraite) dont les fonctions sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.

1.2.2 Description de l'allocation des ressources pour la version du jeu des 15 avec un temps limité pour jouer son tour

Le Tableau 30 décrit l'allocation des ressources pour le jeu des 15 avec un temps limité pour jouer son tour entre les rôles des acteurs. Le principal changement pour cette version du jeu des 15 est le partage du temps restant avant la fin de tour du Leader ou du Challenger.

Le temps pour le tour de jeu est limité pour le Leader et le Challenger. L'Arbitre présente le temps restant pour le tour par le biais d'un élément graphique de la couche interface « Texte indiquant le temps restant » visible par le Leader et le Challenger. La ressource « fin de tour » est partagée entre les rôles Arbitre, Leader et Challenger par l'affichage du temps restant en plus du bouton permettant de mettre fin au tour. Chacun des rôles Arbitre, Leader et Challenger possède ou connaisse les durées correspondant au temps restant pour les tours de jeu (Durée : « Temps pour le tour de jeu »). L'arbitre manipule un timer (Objet Logiciel : « Timer ») pour décompter le temps restant pour le tour de chaque joueur. Lorsque le Leader ou le Challenger ne sélectionne pas de jeton dans les 30 secondes de leur tour, l'arbitre sélectionne un jeton à leur place. L'arbitre affiche ce jeton sélectionné dans les « Widgets des jetons joués par le Challenger » ou « par le Leader » et le retire des « Widgets des Jetons Disponibles ». Le jeton sélectionné pour le tour est partagé entre les rôles Arbitre, Leader et Challenger par une modification du rendu graphique des « Widgets des jetons joués par le Challenger » ou « par le Leader » et des « Widgets des Jetons Disponibles »

Tableau 30. Description de l'allocation des ressources pour le jeu des 15 avec un temps limité pour jouer son tour

Allocation des ressources				
Ressource [Partagée]	Description DOD	Type DOD	Possesseur	Acquéreur
Jetons disponibles [**P**] Jeton Sélectionné **[P]**	Widgets des Jetons disponibles	Couche interface	Arbitre	*Leader*, *Challenger*
Jeton sélectionné **[P]**	Widgets des jetons joués par le Leader	Couche interface	Arbitre	*Leader*, *Challenger*
	Widgets des jetons joués par le Challenger	Couche interface	Arbitre	*Leader*, *Challenger*
	Jeton sélectionné par le Leader	Objet Logiciel	Arbitre	
	Jeton sélectionné par le Challenger	Objet Logiciel	Arbitre	
	Jeton sélectionné pour le Leader	Objet Logiciel	Arbitre	
	Jeton sélectionné pour le Challenger	Objet Logiciel	Arbitre	
	Nouveau jeton sélectionné	Information	*Leader*, *Challenger*	
	Le système a sélectionné un jeton	Information	*Leader*, *Challenger*	
	Jeton sélectionné automatiquement	Information	*Leader*, *Challenger*	
	Jeton sélectionné	Information	*Leader*, *Challenger*	
	Alternatives de choix de jetons possibles	Information	*Leader*, *Challenger*	
Jetons disponibles **[P]**	Jetons disponibles	Objet Logiciel	Arbitre	
	Jetons disponibles	Information	*Leader*, *Challenger*	
Jetons du Leader **[P]**	Widgets des jetons joués par le Leader	Couche Interface	Arbitre	*Leader*, *Challenger*
	Jetons joués par le Leader	Objet logiciel	Arbitre	
	Jetons joués	Information	*Leader*, *Challenger*	
	Jetons joués par le Leader	Information	*Challenger*	
Jetons du Challenger **[P]**	Widgets des jetons joués par le Challenger	Couche Interface	Arbitre	*Leader*, *Challenger*
	Jetons joués par le Challenger	Objet logiciel	Arbitre	
	Jetons joués	Information	*Challenger*	
	Jetons joués par le Challenger	Information	*Leader*	
Fin de tour **[P]**	Bouton fin de tour	Couche Interface	Arbitre	*Leader*, *Challenger*
	Texte indiquant le temps restant	Couche Interface	Arbitre	*Leader*, *Challenger*
	Fin de tour	Objet Logiciel	Arbitre	
	Timer	Objet Logiciel	Arbitre	
	Temps pour le tour de jeu	Durée	Arbitre	

	Temps pour le tour de jeu	Durée	Leader	
	Temps pour le tour de jeu	Durée	Challenger	
	Bouton fin de tour disponible	Information	*Leader*, *Challenger*	
	Volonté de mettre fin à son tour	Information	*Leader*, *Challenger*	
	Bouton fin de tour désactivé	Information	*Leader*, *Challenger*	
	Le tour est terminé	Information	*Leader*, *Challenger*	
Joueur Courant **[P]**	Texte d'indication des tours des joueurs	Couche Interface	Arbitre	*Leader*, *Challenger*
	Texte demandant aux joueurs d'attendre	Couche Interface	Arbitre	*Leader*, *Challenger*
	Joueur courant	Objet Logiciel	Arbitre	
	Je commence à jouer	Information	*Leader*	
	Je dois attendre	Information	*Challenger*	
	Le Leader doit attendre	Information	*Leader*	
	Le Challenger doit jouer	Information	*Challenger*	
	Le Leader doit jouer	Information	*Leader*	
	Le Challenger doit attendre	Information	*Challenger*	
	Joueur courant	Information	*Leader*, *Challenger*	
	Requête d'attente	Information	*Leader*, *Challenger*	
	Requête analysée	Information	*Leader*, *Challenger*	
Résultat de la partie **[P]**	Fin de partie	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Etat de la partie	Information	*Leader*, *Challenger*	
	Résultat du jeu	Information	*Leader*, *Challenger*	
Rejouer **[P]**	Bouton Rejouer	Couche Interface	Arbitre	*Leader*, *Challenger*
	Rejouer	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Arrêter de jouer	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Rejouer	Objet Logiciel	Arbitre	
	Volonté de rejouer	Information	*Leader*, *Challenger*	
	Bouton pour rejouer	Information	*Leader*, *Challenger*	
	Rejouer	Information	*Leader*, *Challenger*	
	Rôle qui prend la souris pour lancer une nouvelle partie	Information	*Leader*, *Challenger*	
Stratégie de jeu	Stratégie de jeu	Information Stratégique	*Leader*, *Challenger*	

Stratégie d'attribution des rôles	Aléatoire	Données Stratégique	Arbitre	
Lancer partie **[P]**	Bouton pour lancer la partie	Couche Interface	Arbitre	Joueur
	Lancer partie	Objet Logiciel	Arbitre	
	Lancer la partie	Parole	*Leader, Challenger*	*Leader, Challenger*
	Bouton pour lancer une partie	Information	Joueur	
	Joueur qui presse le bouton lancer la partie	Information	Joueur	
Rôle **[P]**	Texte décrivant l'allocation des rôles aux joueurs	Couche Interface	Arbitre	Joueur
	Rôles des joueurs	Objet Logiciel	Arbitre	
	Challenger	Objet Logiciel	Arbitre	
	Leader	Objet Logiciel	Arbitre	
	Distribution des rôles affichée	Information	Joueur	
	Rôle	Information	Joueur	

1.2.3 Description de l'allocation des transitions de contrôle et de l'initiative pour la version du jeu des 15 avec un temps limité pour jouer son tour

Le Tableau 31 décrit l'allocation des transitions de contrôle de l'initiative pour le jeu des 15 avec un temps limité pour jouer son tour entre les rôles des acteurs selon leurs tâches ou fonctions. Le principal changement en termes d'allocation des transitions de contrôle et de l'initiative pour cette version est une possible initiation d'une transition de contrôle par l'arbitre pour mettre fin au tour de jeu des joueurs au bout de 30 secondes.

Au bout de 30 secondes de tour de jeu, l'Arbitre interrompt le tour. Pour interrompre le tour, l'Arbitre empêche la sélection d'un jeton et de mettre fin au tour de jeu en pressant le bouton « fin de tour ». Les fonctions de sortie coopérative « Afficher les jetons disponibles » et « Afficher le bouton fin de tour désactivé » exécutée à la fin des 30 secondes initient des transitions de contrôle. L'arbitre réalise un *takeover* du contrôle des jetons disponibles et du bouton fin de tour : l'arbitre empêche tout autre rôle de manipuler les DODs « Jetons Disponibles » et « Bouton fin de tour ». Ensuite, l'Arbitre sélectionne un jeton parmi les jetons disponibles pour le Leader ou le Challenger si le joueur n'avait pas sélectionné de jeton. Puis, l'arbitre affiche le tour suivant.

Tableau 31. Description de l'allocation des transitions de Contrôle et de l'initiative avec un temps limité pour jouer son tour

Allocation des transitions de contrôle et de l'initiative				
Transition	Initiative	*Handover* ou émetteur	*Takeover* ou receveur	Tâches et Fonctions

X	Arbitre	Arbitre : « Afficher les jetons disponibles » **HANDOVER**	Leader : « Voir les jetons disponibles »	Fin attente pour le *Leader* et le *Challenger* *Leader* joue son tour *Challenger* attend l'arbitre finit d'afficher le jet et démarre le supporte du tour du *Leader*
X	Arbitre	Arbitre : « Afficher les jetons disponibles » **HANDOVER**	Challenger : « Voir les jetons disponibles »	
X	Arbitre	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Leader : « Voir les jetons disponibles désactivés »	
X	Arbitre	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Challenger : « Voir les jetons disponibles désactivés »	
X	Leader	Leader : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »	
X	Challenger	Challenger : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »	
x	Arbitre	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Leader : « Voir les jetons disponibles désactivés »	Interrompt le tour de jeu du Leader
x	Arbitre	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Challenger : « Voir les jetons disponibles désactivés »	Interrompt le tour de jeu du Challenger
	Arbitre	Arbitre : « Afficher les jetons joués par le Leader »	Leader : « Voir mes jetons »	
	Arbitre	Arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Leader »	Leader : « Voir le jeton Sélectionné »	
	Arbitre	Arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Leader »	Challenger : « Voir le jeton sélectionné »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Leader »	Challenger : « Voir les jetons du Leader »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Challenger »	Leader : « Voir les jetons du Challenger »	
	Arbitre	Afficher le jeton sélectionné dans les jetons joués par le Challenger	Challenger : « Voir le jeton Sélectionné »	
	Arbitre	Arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Leader »	Leader : « Voir le jeton sélectionné »	
	Arbitre	Arbitre : « Afficher les jetons joués	*Challenger :* « « Voir mes jetons »	

		par le Challenger »		
x	Arbitre	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Leader : « Voir le bouton fin de tour activé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Challenger : « Voir le bouton fin de tour activé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Leader : « Voir le bouton fin de tour désactivé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Challenger : « Voir le bouton fin de tour désactivé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Challenger : « Voir le bouton fin de tour désactivé »	Interrompt le tour de jeu du Leader
x	Arbitre	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Leader : « Voir le bouton fin de tour désactivé »	Interrompt le tour de jeu du Challenger
x	Leader	Leader : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »	
x	Challenger	Challenger : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »	
	Arbitre	Arbitre : « Afficher le joueur courant »	Leader : « Voir le joueur courant »	
	Arbitre	Arbitre : « Afficher le joueur courant »	Challenger : « Voir le joueur courant »	
	Arbitre	Arbitre ; « Afficher la requête d'attente »	Leader : « Voir la requête d'attente »	
	Arbitre	Arbitre ; « Afficher la requête d'attente »	Challenger : « Voir la requête d'attente »	
	Leader	Leader : « Annoncer la fin de la partie »	Challenger : « Entendre le Leader »	Fin des tours de jeu
	Challenger	Challenger : « Annoncer la fin de la partie »	Leader : « Entendre le Challenger »	Fin des tours de jeu
X	Arbitre	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Leader : « Voir le bouton rejouer »	
X	Arbitre	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Challenger : « Voir le bouton rejouer »	
	Leader	Leader : « Rejouer »	Arbitre : « Recevoir rejouer »	
	Challenger	Challenger : « Rejouer »	Arbitre : « Recevoir rejouer »	
	Leader	Leader : « Annoncer une nouvelle partie »	Challenger : « Entendre qu'on rejoue »	

	Challenger	Challenger : « Annoncer une nouvelle partie »	Leader : « Entendre qu'on rejoue »	
	Leader	Leader : « Annoncer qu'on arrête de jouer »	Challenger : « Entendre qu'on arrête de jouer »	Fin du jeu des 15
	Challenger	Challenger : « Annoncer qu'on arrête de jouer »	Leader : « Entendre qu'on arrête de jouer »	Fin du jeu des 15
x	Arbitre	Arbitre : « Afficher le bouton lancer la partie » **HANDOVER**	Joueur : « Voir le bouton lancer la partie »	
x	Joueur	Joueur : « Lancer la partie » **HANDOVER**	Arbitre : « Recevoir la pression bouton »	
	Joueur	Joueur : « Annoncer la partie du jeu des 15 »	Joueur : « Entendre le début de la partie du jeu des 15 »	
	Arbitre	Arbitre : « Afficher les rôles pour les joueurs »	Joueur : « Voir l'allocation des rôles »	

1.2.4 Description de l'allocation de la responsabilité pour la version du jeu des 15 avec un temps limité pour jouer son tour

Le Tableau 32 décrit l'allocation de la responsabilité pour le jeu des 15 avec un temps limité pour jouer son tour entre les rôles des acteurs selon leurs tâches ou fonctions. Le principal changement en termes d'allocation de la responsabilité pour cette version est que l'Arbitre peut sélectionner un jeton pour le Leader ou le Challenger.

Lorsque le Leader ou le Challenger ne choisit pas un jeton pendant les 30 secondes de son tour de jeu, l'Arbitre sélectionne un jeton à sa place. Cette fonction peut faire dévier les résultats attendus du Leader et du Challenger de « Posséder 3 jetons dont la somme est égale à 15. Les fonctions « Sélectionner un jeton au hasard pour le Leader » et « Sélectionner un jeton au hasard pour le Leader » sont à responsabilité.

Tableau 32. Description de l'allocation de la responsabilité pour le jeu des 15 avec un temps limité pour jouer son tour

Allocation de la responsabilité	
Tâche/ Fonction	Résultat attendu possiblement impacté
Leader : « Sélection d'un jeton »	*Leader* : « Posséder 3 jetons dont la somme est égale à 15 »
Tâche motrice Leader : « Sélectionner un jeton »	Leader : « Posséder 3 jetons dont la somme est égale à 15 »
Tâche motrice Challenger : « Sélectionner un jeton »	Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Initialiser les jetons disponibles »	Arbitre : « Un seul jeton est sélection chaque tour »
Fonction de sortie de l'arbitre : « Afficher les jetons disponibles activés »	
Fonction de l'arbitre : « Désactiver les jetons disponibles »	
Fonction de sortie de l'arbitre : « Retirer le jeton sélectionné des jetons disponibles et afficher les jetons disponibles désactivés »	
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour »

	Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Sélectionner un jeton au hasard pour le Leader »	Leader : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Sélectionner un jeton au hasard pour le Challenger »	Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Leader »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Challenger »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Arbitre : « Les joueurs jouent chacun leur tours »
Fonction de sortie de l'arbitre : « Afficher le bouton fin de tour désactivé »	
Fonction de l'arbitre : « Activer le bouton fin de tour »	
Fonction de sortie : « Afficher le bouton fin de tour disponible »	
Tâche motrice du Leader : « Mettre fin au tour »	
Tâche motrice du Challenger : « Mettre fin au tour »	
Fonction de sortie de l'arbitre : « Afficher le joueur courant »	
Fonction de l'arbitre : « Initialiser le joueur courant »	
Fonction de l'arbitre : « Changer le joueur courant »	
Fonction de l'arbitre : « Allouer les rôles aléatoirement »	Arbitre : « Le jeu est joué avec 2 joueurs »
Fonction de sortie de l'arbitre : « Afficher les rôles des joueurs »	

1.2.5 Description de l'allocation de l'autorité pour la version du jeu des 15 avec un temps limité pour jouer son tour

Le Tableau 33 décrit l'allocation de l'autorité pour le jeu des 15 avec un temps limité pour jouer son tour entre les rôles des acteurs selon leurs tâches ou fonctions. Le principal changement en termes d'allocation de l'autorité de cette version du jeu est que l'Arbitre définit une contrainte de temps pour le tour de jeu du Leader et du Challenger.

Dans cette version du jeu des 15, l'Arbitre définit une durée pour les tours de jeu du Leader et du Challenger. La fonction « Définir et lancer le *timer* » est à autorité car elle définit une contrainte de temporelle sur les tâches et fonctions du Leader et du Challenger : sélectionner un jeton. L'arbitre peut sélectionner un jeton à la place du Leader ou du Challenger si un jeton n'est pas sélectionné pendant les 30 secondes du tour de jeu. Les fonctions « Sélectionner un jeton au hasard pour le Leader » et « Sélectionner un jeton au hasard pour le Challenger » sont à autorité car elles définissent une valeur pour le jeton sélectionné pour le tour du Leader ou du Challenger.

Tableau 33. Description de l'allocation de l'autorité avec un temps limité pour jouer son tour

Allocation de l'autorité	
Tâche/Fonction	Contrainte
Arbitre : Désactiver les jetons disponibles	La sélection d'un jeton n'est plus possible à partir de l'exécution de cette fonction
Fonction de l'arbitre : « Activer les jetons disponibles »	Autoriser la sélection d'un jeton
Fonction de l'arbitre : Initialiser les jetons disponibles	Presser le widget pour sélectionner la valeur du jeton sélectionné
Tâche motrice du Leader : « Sélectionner le jeton »	Effectuer la sélection
Tâche motrice du Challenger : « Sélectionner le jeton »	Effectuer la sélection
Fonction de l'arbitre : « Sélectionner un jeton au hasard pour le Leader »	Définir la valeur du jeton pour le Leader
Fonction de l'arbitre : « Sélectionner un jeton au hasard pour le Challenger »	Définir la valeur du jeton pour le Challenger
Tâche de décision du Leader : « Choisir un jeton »	Valeur du jeton à sélectionner
Tâche de décision du Challenger : « Choisir un jeton »	Valeur du jeton à sélectionner
Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Empêcher la pression du bouton fin de tour
Fonction de l'arbitre : « Activer le bouton fin de tour »	Autoriser la pression du bouton fin de tour
Fonction de l'arbitre : « Initialiser le bouton fin de tour »	Bouton à presser pour mettre fin au tour
Tâche motrice du Leader : « Mettre fin au tour »	Mettre fin au tour
Tâche motrice du Challenger : « Mettre fin au tour »	Mettre fin au tour
Fonction de l'arbitre : « Définir et lancer le timer »	Définir le temps de jeu pour les joueurs
Fonction de l'arbitre : « Initialiser le joueur courant »	Définir valeur premier joueur
Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Challenger

Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Leader
Tâche cognitive de décision du Leader : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Tâche cognitive de décision du Challenger : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Fonction de l'arbitre : Initialiser le bouton rejouer	Presser le bouton pour rejouer
Fonction de l'arbitre : Activer le bouton rejouer	Autoriser pression bouton rejouer
Tâche motrice du Leader : « Rejouer »	Lancer une nouvelle partie
Tâche motrice du Challenger : « Rejouer »	Lancer une nouvelle partie
Tâche cognitive de décision du Leader : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche cognitive de décision du Challenger : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche de groupe du Leader et du Challenger : « Les joueurs se mettent d'accord pour rejouer »	Défini qui presse le bouton pour rejouer
Tâche cognitive d'analyse du Leader : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Tâche cognitive d'analyse du Challenger : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Fonction de l'Arbitre : « Initialiser le bouton lancer partie »	Presser le bouton lancer partie pour lancer la partie
Fonction de l'Arbitre : « Activer le bouton lancer partie »	Activer bouton lancer partie
Tâche motrice du Joueur : « Lancer la partie »	Lancer la partie
Tâche motrice parler du joueur : « Annoncer la partie du jeu des 15 »	Définir quel joueur lance la partie
Fonction de l'arbitre : « Allouer les rôles aléatoirement aux joueurs »	Définir la valeur du rôle des joueurs

1.2.1 Description de l'allocation RCRA dans les modèles de tâches ségrégués et de fonctions

La Figure 95 et Figure 96 présentent le modèle de tâches ségrégué du Leader et le modèle de fonctions de l'arbitre. Le modèle de tâches ségrégué du Challenger décrit les mêmes tâches que pour le Leader sauf que le Challenger joue son tour en second. L'allocation RCRA du rôle Joueur n'évolue pas dans cette version du jeu des 15 par rapport à la version précédente du jeu des 15 présentée au Chapitre 4.1.

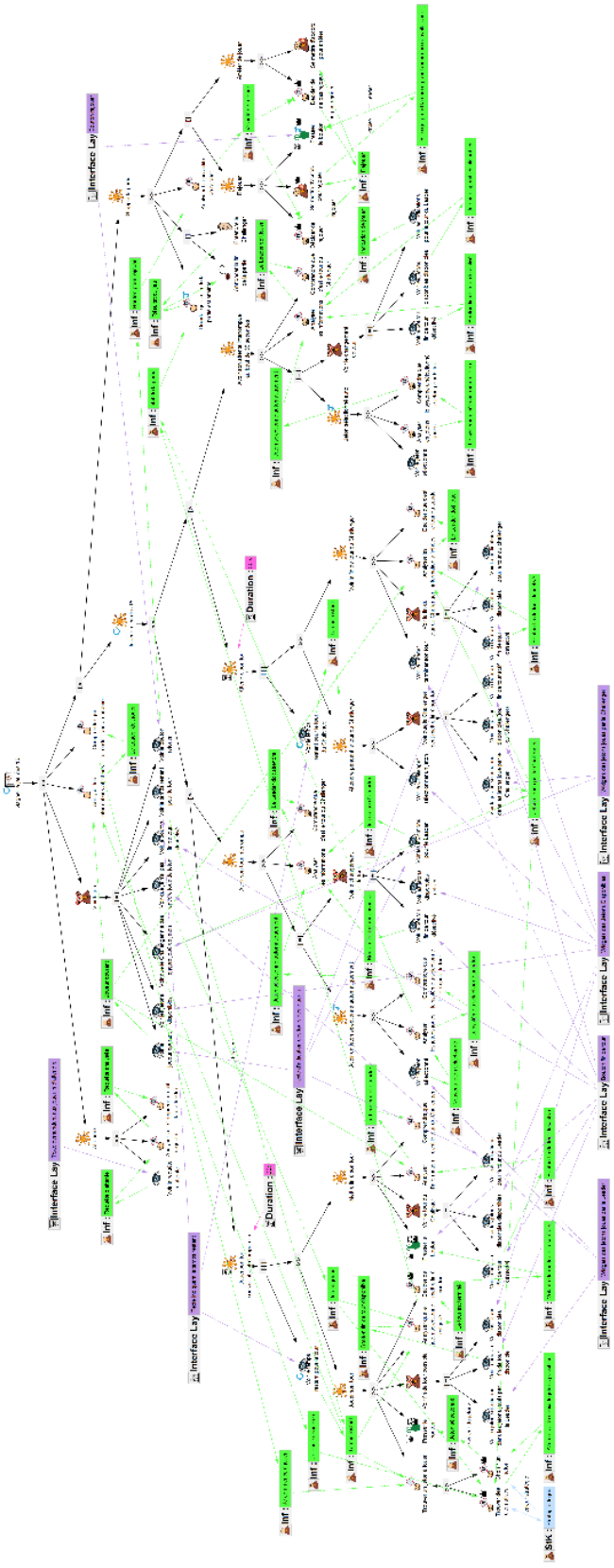


Figure 95. Modèle de tâches ségrégués du Leader pour gagner le jeu des 15.

Figure 96. Modèle de fonctions de l'arbitre pour soutenir la partie dans le respect des règles du jeu des 15

1.3 Comparaison des versions du jeu des 15

Dans cette section, nous présentons les différences en termes d'allocation RCRAFT entre le jeu des 15 qui a été présenté au Chapitre 4.1 et la version du jeu des 15 avec un temps limité pour jouer son tour.

1.3.1 Comparaison de l'allocation des fonctions et des tâches

Dans cette sous-section, nous présentons la comparaison de l'allocation des tâches et des fonctions entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du jeu des 15 avec un temps limité pour jouer son tour.

1.3.1.1 Identification des différences

Dans la version du jeu des 15 avec un temps limité, des fonctions supplémentaires sont allouées à l'Arbitre. Ce sont les fonctions relatives à la gestion du temps du tour de jeu et à l'interruption d'un tour du jeu lorsque la durée de 30 secondes est écoulée. Ces fonctions supplémentaires sont :

- Définir et lancer le *timer* (fonction)
- Afficher le temps restant pour le tour (fonction de sortie coopérative)
- Indiquer le temps restant (fonction abstraite itérative)
    - Décompter (fonction)
    - Afficher le temps restant pour le tour (fonction de sortie coopérative)
- Interrompre le tour de jeu (fonction abstraite)
    - Désactiver les widgets (fonction abstraite)
    - Désactiver les jetons disponibles (fonction)
    - Afficher les jetons disponibles désactivés (fonction de sortie coopérative)
    - Désactiver le bouton fin de tour (fonction)
    - Afficher le bouton fin de tour désactivé (fonctions de sortie coopérative)
- Choisir un jeton pour le Leader (fonctions abstraite optionnelle)
    - Si l'objet logiciel « Jeton sélectionné par le Leader » est vide :
    - Sélectionner un jeton au hasard pour le Leader (fonction)
    - Afficher le jeton sélectionné dans les jetons joués par le Leader (fonction de sortie coopérative)
    - Retirer le jeton sélectionné des jetons disponibles désactivés (fonction de sortie coopérative)

L'ordonnancement des tâches et des fonctions est également modifié afin de pouvoir interrompre les tours de jeu et gérer le changement de tour après l'interruption.

Des tâches supplémentaires sont également allouées au Leader et le Challenger pour gérer la sélection d'un jeton sous contrainte temporelle et la gestion de l'interruption d'un tour de jeu. Ces tâches supplémentaires sont :

- Voir le temps restant pour le tour (tâche perceptive coopérative)
- Jouer son tour sous contrainte temporelle (tâche abstraite avec une durée maximale de 30 secondes)

    - Voir le temps restant pour le tour (tâche perceptive itérative)
- Avoir son tour interrompu (tâche abstraite)
    - Avoir un jeton sélectionné automatiquement (tâche abstraite optionnelle)
        - Voir le jeton sélectionné (tâche perceptive coopérative)
        - Analyser les jetons joués (tâche cognitive d'analyse)
        - Comprendre que le système a sélectionné le jeton pour le tour (tâche cognitive de décision)
- Attendre son tour (tâche abstraite avec une durée maximale de 30 secondes)
    - Voir le temps restant pour le tour du Challenger (tâche perceptive itérative)
- Avoir son attente interrompue au bout de 30 secondes (tâche abstraite)

1.3.1.2 Impact

Bien que le décompte du temps restant pour le tour, l'interruption du tour et la sélection automatique d'un jeton soient des fonctions allouées au rôle arbitre, le Leader et le Challenger ont également de nouvelles tâches allouées dans cette version du jeu des 15. Ces tâches permettent de prendre connaissance du temps restant pour le tour et de gérer une interruption de tour.

1.3.2 Comparaison de l'allocation des ressources

Dans cette sous-section, nous présentons la comparaison de l'allocation des ressources entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du jeu des 15 avec un temps limité pour jouer son tour.

1.3.2.1 Identification des différences

Dans la version du jeu des 15 avec un temps limité, l'arbitre manipule de nouveaux DODs :

- Objet logiciel : « Timer »
- Durée : « Temps pour le tour de jeu »
- Objet logiciel : « Le temps est écoulé »
- Objet logiciel : « Jeton sélectionner pour le Leader »
- Objet logiciel : « Jeton sélectionner pour le Challenger »
- Elément de la couche interface : « Texte indiquant le temps restant »

Également, dans cette version du jeu des 15, le Leader et le Challenger manipulent de nouveaux DODs :

- Elément de la couche interface « Texte indiquant le temps restant »
- Information : Temps restant
- Durée : 30 secondes
- Information : Nouveau Jeton sélectionné
- Information : Le système a sélectionné un jeton
- Information : Jeton sélectionné automatiquement

Dans cette version du jeu des 15, l'arbitre affiche un nouvel élément le « Texte indiquant le temps restant » par rapport à la version du jeu des 15 précédente. La ressource « fin de tour »

est partagée entre les rôles Arbitre, Leader et Challenger par l'affichage de ce nouvel élément graphique en plus du bouton permettant de mettre fin au tour.

Dans la précédente version du jeu des 15, les « Widgets des Jetons Disponibles » étaient utilisés par le Leader et le Challenger pour communiquer le jeton sélectionné pour le tour. Dans la version avec un temps limité pour les tours de jeu, l'arbitre peut sélectionner un jeton pour le Leader ou le Challenger. L'arbitre communique le jeton sélectionné en retirant la valeur du jeton des « Widgets des Jetons Disponibles » et en l'ajoutant aux « Widgets des jetons joués par le Challenger » ou « par le Leader ».

1.3.2.2 Impact

En affichant le temps restant pour le tour, l'arbitre partage le temps restant pour le tour. Le Leader et le Challenger peuvent savoir que le tour de jeu est sous contrainte temporelle. A moins qu'un des deux joueurs ne finisse pas son tour avant la fin du temps imparti, le Leader et le Challenger ne peuvent pas anticiper le comportement du système à la fin du temps imparti. Si un des deux joueurs ne finit pas son tour avant la fin du temps imparti, le Leader ou le Challenger peut savoir quel jeton a été sélectionné en localisant le jeton manquant des jetons disponibles et le nouveau jeton dans ses jetons joués.

1.3.3 Comparaison de l'allocation des transitions de contrôle et de l'initiative

Dans cette sous-section, nous présentons la comparaison de l'allocation des transitions de contrôle et de l'initiative entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du jeu des 15 avec un temps limité pour jouer son tour.

1.3.3.1 Identification des différences

Dans la version du jeu des 15 avec un temps limité, l'arbitre peut exécuter un *takeover* lors de la transition de contrôle interrompant le tour de jeu du Leader ou du Challenger au bout de 30 secondes. L'arbitre empêche tout autre rôle de manipuler les DODs « Jetons Disponibles » et « Bouton fin de tour » alors toutes les tâches et fonctions du tour de jeu n'ont pas été réalisées.

1.3.3.1 Impact

Dans la version précédente du jeu des 15, l'arbitre ne pouvait pas reprendre le contrôle des éléments du jeu de lui-même avant que le Leader ou le Challenger aient mis fin à leur tour.

1.3.4 Comparaison de l'allocation de la responsabilité

Dans cette sous-section, nous présentons la comparaison de l'allocation de la responsabilité entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du jeu des 15 avec un temps limité pour jouer son tour.

1.3.4.1 Identification des différences

Dans la version du jeu des 15 avec un temps limité, l'arbitre peut sélectionner un jeton à la place du Leader ou du Challenger. Cette fonction manipule les jetons joués par le Leader et le Challenger et peut donc faire dévier les résultats attendus du Leader et du Challenger de « Posséder 3 jetons dont la somme est égale à 15.

1.3.4.1 Impact

L'arbitre peut être responsable si le Leader ou le Challenger ne gagne pas. Dans la version précédente du jeu, seuls le Leader et le Challenger réalisaient des tâches et fonctions susceptibles de faire dévier leurs résultats attendus.

1.3.5 Comparaison de l'allocation de l'autorité

Dans cette sous-section, nous présentons la comparaison de l'allocation de l'autorité entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du jeu des 15 avec un temps limité pour jouer son tour.

1.3.5.1 Identification des différences

Dans la version du jeu des 15 avec un temps limité, l'arbitre définit une contrainte sur l'action de sélection d'un jeton et de mettre fin à un tour de jeu en définissant une durée pour les tours de jeu du Leader et du Challenger. Également, l'arbitre peut contraindre la valeur du jeton sélectionné pour le tour du Leader ou du Challenger si un jeton n'est pas sélectionné pendant les 30 secondes du tour de jeu. Les Joueurs, le Leader et le Challenger n'ont pas de tâche leur permettant de corriger la sélection du système ou de modifier la durée d'un tour.

1.3.5.1 Impact

Le rôle Arbitre a plus d'autorité dans cette version que dans la version précédente du jeu. Les rôles Joueurs, le Leader et le Challenger ont autant d'autorité que dans la version précédente du jeu.

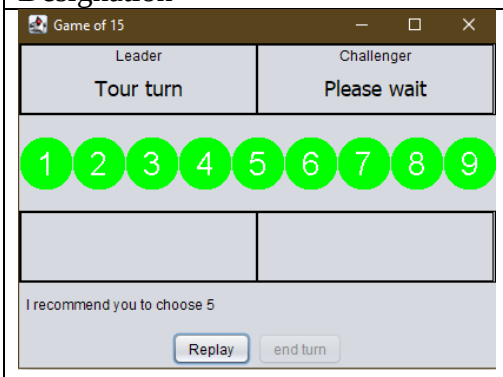
2 Version du jeu des 15 proposant des recommandations aux joueurs

Dans cette première section, nous présentons la description de l'allocation RCRAFT sur une version du jeu des 15 proposant des recommandations aux joueurs. Dans la première sous-section, nous présentons la version du jeu des 15 proposant des recommandations aux joueurs. Dans la seconde sous-section, nous décrivons l'allocation RCRAFT produite suite à l'application du processus RCRAFT pour la version du jeu des 15 proposant des recommandations aux joueurs. Enfin, nous présentons la comparaison en termes d'allocation RCRAFT du jeu des 15 proposant des recommandations aux joueurs et la version du jeu des 15 présenté au Chapitre 4.1.

2.1 Présentation du jeu des 15 proposant des recommandations aux joueurs

Cette version du jeu des 15 se joue à deux joueurs. Ils disposent de jetons représentant les chiffres de 1 à 9 et de recommandations pour leurs tours de jeu. Chacun leur tour, ils doivent choisir un jeton qui devient indisponible pour le joueur adverse. Lorsque l'un des deux joueurs possède trois jetons dont la somme est égale à 15, la partie est terminée et ce joueur gagne. Ici, nous considérons que deux joueurs jouent l'un contre l'autre grâce à un système permettant de jouer au jeu des 15. Le système indique les tours de jeux, permet de mettre fin à un tour de jeu, affiche les jetons disponibles et joués par les deux joueurs et affiche des recommandations pour

le tour. Les joueurs peuvent choisir de rejouer une partie à tout moment. Le résume cette version du jeu des 15.

Désignation	Description
 Le jeu des 15 Recommandations	Les jetons disponibles, les jetons joués par le système et les jetons joués par le joueur sont affichés par le système. Lors du tour du joueur, le système indique les meilleurs jetons à choisir pour gagner ou les jetons à ne pas jouer pour ce tour. Lorsqu'un joueur sélectionne un jeton, le bouton fin de tour devient actif. Lorsque le joueur clique sur le bouton fin de tour, le tour de jeu se termine et c'est à l'adversaire de jouer. Les joueurs peuvent choisir de rejouer une partie à tout moment.

2.2 Description de l'allocation RCRAFT pour la version du jeu des 15 proposant des recommandations aux joueurs

Dans cette sous-section, nous décrivons l'allocation RCRAFT produite suite à l'application du processus RCRAFT pour la version du jeu des 15 proposant des recommandations aux joueurs. Nous ne présentons seulement les artefacts produits après l'application du processus : les modèles de tâches ségrégués et les modèles de fonctions, le tableau RCRA et les modèles de tâches ségrégués et les modèles de fonctions intégrant la description de l'allocation RCRA.

2.2.1 Description de l'allocation des tâches et des fonctions pour la version du jeu des 15 proposant des recommandations aux joueurs

La Figure 93 présente un aperçu du modèle de tâches ségrégué du rôle Leader. La Figure 94 présente un aperçu du modèle de fonctions du rôle Arbitre. La description sous formes de modèles de tâches de l'allocation des tâches et des fonctions de chacun des rôles sont consultables en Annexes. Le principal changement de cette version en termes d'allocation de tâches et de fonctions est qu'il existe un nouveau rôle pour cette version : le recommandeur qui recommande des jetons aux joueurs.

Les tâches du rôle Joueur pour lancer la partie sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.

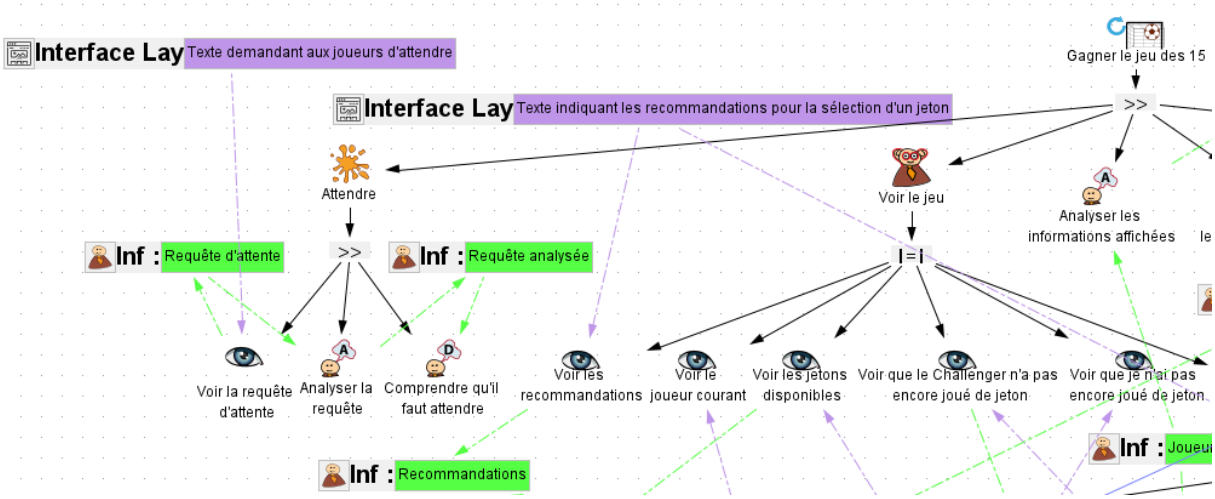


Figure 97. Tâche « Voir le jeu » du modèle de tâches ségrégué du Leader pour gagner le jeu des 15 modifiée dans cette version du jeu des 15.

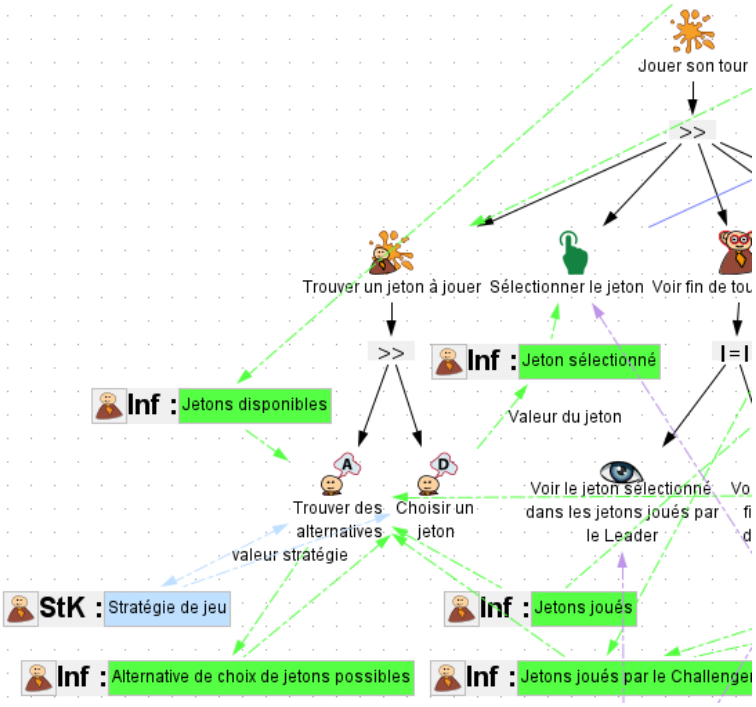


Figure 98. Tâche « Trouver un jeton à jouer » du modèle de tâches ségrégué du Leader pour gagner le jeu des 15 modifiée dans cette version du jeu des 15.

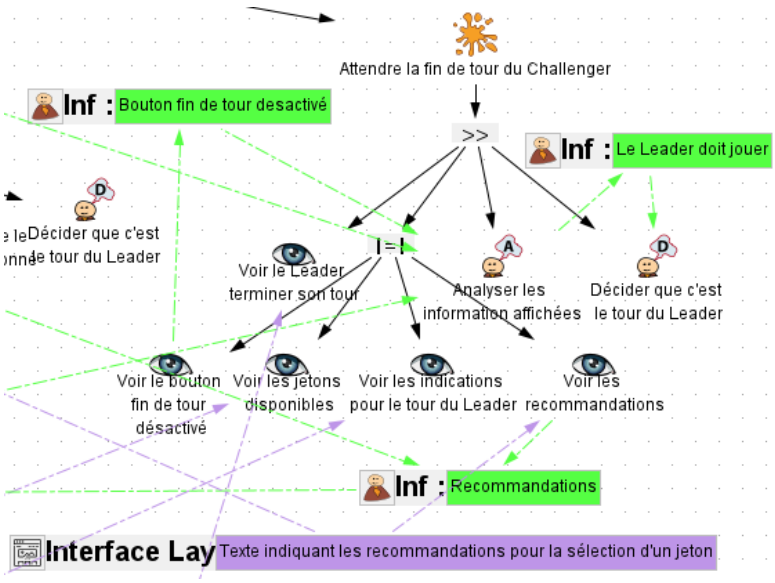


Figure 99. Tâche « Attendre la fin de tour du Challenger » du modèle de tâches ségrégué du Leader pour gagner le jeu des 15 modifiée dans cette version du jeu des 15.

Dans la version du jeu des 15 proposant des recommandations, les tâches allouées au Leader sont :

- Attendre (le début de la partie) dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
- Voir le jeu visible dans la Figure 97 dont une tâche perceptive coopérative « Voir les recommandations »
- Analyser les informations affichées et comprendre que le Leader joue en premier
- Jouer au jeu des 15 (tâche abstraite itérative)
    - Jouer son tour
        - Trouver un jeton à jouer visible dans la Figure 98 où la tâche cognitive d'analyse « Trouver des alternatives » a en entrée les recommandations sous forme d'information.
        - Le reste des tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
    - Mettre fin à son tour dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
    - Attendre pendant le tour du Challenger dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
    - Attendre la fin de tour du Challenger
        - Voir les recommandations (tâche perceptive coopérative)
        - Le reste des tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.
- Stopper la partie dont les tâches sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.

Les tâches du Leader et du Challenger sont identiques à la différence que le Leader est le premier à jouer son tour.

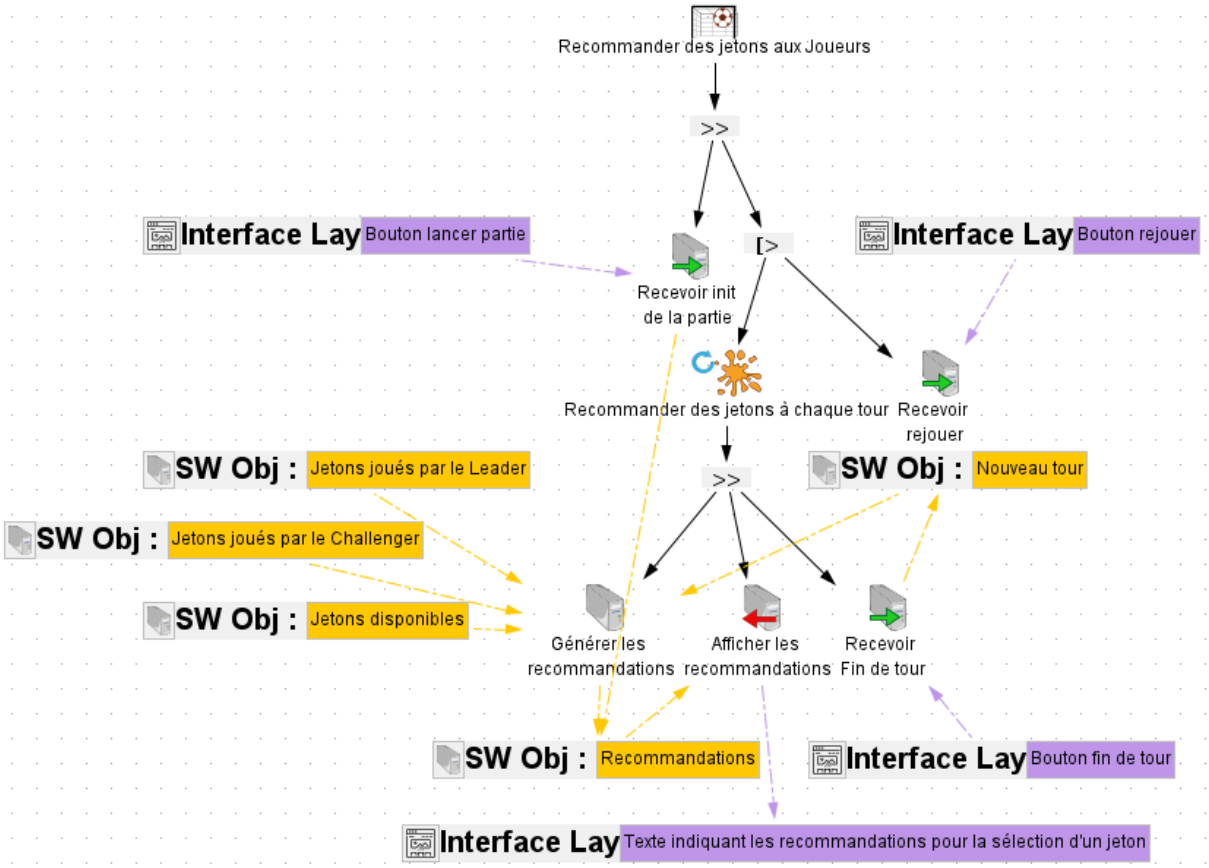


Figure 100. Modèle de fonctions du recommandeur pour recommander des jetons aux joueurs.

Les fonctions allouées à l'Arbitre sont identiques à la version du jeu des 15 présentée au Chapitre 4.1.

Un rôle supplémentaire est attribué à l'acteur système. Ce rôle est recommandeur. Il soutient les buts des utilisateurs en recommandant des jetons aux joueurs. Le modèle de fonctions de ce rôle est visible en Figure 100.

Les fonctions allouées au recommandeur présentée en Figure 100 sont :

- Recevoir l'initialisation de la partie (fonction d'entrée coopérative)
- Recommander des jetons à chaque tour (fonction abstraite itérative)
  - Générer les recommandations (fonction)
  - Afficher les recommandations (fonction de sortie coopérative)
  - Recevoir la fin de tour (fonction d'entrée coopérative)
- Recevoir rejouer (fonction d'entrée coopérative)

2.2.2 Description de l'allocation des ressources pour la version du jeu des 15 proposant des recommandations aux joueurs

Le Tableau 34 décrit l'allocation des ressources entre les rôles des acteurs pour la version du jeu des 15 proposant des recommandations aux joueurs. Le principal changement en termes d'allocation des ressources pour cette version concerne la ressource « jeton sélectionné » pour laquelle des recommandations sont partagées par le recommandeur aux joueurs.

Des recommandations de valeurs de jetons à sélectionner ou à ne pas sélectionner sont affichées aux Leaders et au Challenger pour leur tour de jeu. Le rôle recommandeur génère des recommandations (objet logiciel : recommandations). Ces recommandations sont affichées par le biais d'un élément graphique de la couche interface « Texte indiquant les recommandations pour la sélection d'un jeton » visible par le Leader et le Challenger. Le recommandeur partage des recommandations de valeurs pour la ressource « jeton sélectionné » au Leader et au Challenger. Ces recommandations sont perçues, stockées et manipulées par le Leader et le Challenger sous forme d'information.

Tableau 34. Description de l'allocation des Ressources pour le jeu des 15 proposant des recommandations aux joueurs

Allocation des ressources				
Ressource [Partagée]	Description DOD	Type DOD	Possesseur	Acquéreur
Jetons disponibles [**P**] Jeton Sélectionné **[P]**	Widgets des Jetons disponibles	Couche interface	Arbitre	*Leader*, *Challenger*
Jeton sélectionné **[P]**	Jeton sélectionné par le Leader	Objet Logiciel	Arbitre	
	Jeton sélectionné par le Challenger	Objet Logiciel	Arbitre	
	Jeton sélectionné	Information	*Leader*, *Challenger*	
	Texte indiquant les recommandations pour la sélection d'un jeton	Couche interface	Recommandeur	*Leader*, *Challenger*
	Recommandations	Information	*Leader*, *Challenger*	
	Recommandations	Objet Logiciel	Recommandeur	
	Alternatives de choix de jetons possibles	Information	*Leader*, *Challenger*	
Jetons disponibles **[P]**	Jetons disponibles	Objet Logiciel	Arbitre, Recommandeur	
	Jetons disponibles	Information	*Leader*, *Challenger*	
Jetons du Leader **[P]**	Widgets des jetons joués par le Leader	Couche Interface	Arbitre	*Leader*, *Challenger*
	Jetons joués par le Leader	Objet logiciel	Arbitre, Recommandeur	
	Jetons joués	Information	*Leader*, *Challenger*	
	Jetons joués par le Leader	Information	*Challenger*	
Jetons du Challenger **[P]**	Widgets des jetons joués par le Challenger	Couche Interface	Arbitre	*Leader*, *Challenger*
	Jetons joués par le Challenger	Objet logiciel	Arbitre, Recommandeur	
	Jetons joués	Information	*Challenger*	
	Jetons joués par le Challenger	Information	*Leader*	
Fin de tour **[P]**	Bouton fin de tour	Couche Interface	Arbitre	*Leader*, *Challenger*, Recommandeur
	Fin de tour	Objet Logiciel	Arbitre	

	Nouveau tour	Objet Logiciel	Recommandeur	
	Bouton fin de tour disponible	Information	*Leader*, *Challenger*	
	Volonté de mettre fin à son tour	Information	*Leader*, *Challenger*	
	Bouton fin de tour désactivé	Information	*Leader*, *Challenger*	
	Le tour est terminé	Information	*Leader*, *Challenger*	
Joueur Courant **[P]**	Texte d'indication des tours des joueurs	Couche Interface	Arbitre	*Leader*, *Challenger*
	Texte demandant aux joueurs d'attendre	Couche Interface	Arbitre	*Leader*, *Challenger*
	Joueur courant	Objet Logiciel	Arbitre	
	Je commence à jouer	Information	*Leader*	
	Je dois attendre	Information	*Challenger*	
	Le Leader doit attendre	Information	*Leader*	
	Le Challenger doit jouer	Information	*Challenger*	
	Le Leader doit jouer	Information	*Leader*	
	Le Challenger doit attendre	Information	*Challenger*	
	Joueur courant	Information	*Leader*, *Challenger*	
	Requête d'attente	Information	*Leader*, *Challenger*	
	Requête analysée	Information	*Leader*, *Challenger*	
Résultat de la partie **[P]**	Fin de partie	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Etat de la partie	Information	*Leader*, *Challenger*	
	Résultat du jeu	Information	*Leader*, *Challenger*	
Rejouer **[P]**	Bouton Rejouer	Couche Interface	Arbitre	*Leader*, *Challenger*, Recommandeur
	Rejouer	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Arrêter de jouer	Parole	*Leader*, *Challenger*	*Leader*, *Challenger*
	Rejouer	Objet Logiciel	Arbitre	
	Volonté de rejouer	Information	*Leader*, *Challenger*	
	Bouton pour rejouer	Information	*Leader*, *Challenger*	
	Rejouer	Information	*Leader*, *Challenger*	
	Rôle qui prend la souris pour lancer une nouvelle partie	Information	*Leader*, *Challenger*	
Stratégie de jeu	Stratégie de jeu	Information Stratégique	*Leader*, *Challenger*	
Stratégie d'attribution des rôles	Aléatoire	Données Stratégique	Arbitre	

Lancer partie **[P]**	Bouton pour lancer la partie	Couche Interface	Arbitre	Joueur, Recommandeur
	Lancer partie	Objet Logiciel	Arbitre	
	Lancer la partie	Parole	*Leader, Challenger*	*Leader, Challenger*
	Bouton pour lancer une partie	Information	Joueur	
	Joueur qui presse le bouton lancer la partie	Information	Joueur	
Rôle **[P]**	Texte décrivant l'allocation des rôles aux joueurs	Couche Interface	Arbitre	Joueur
	Rôles des joueurs	Objet Logiciel	Arbitre	
	Challenger	Objet Logiciel	Arbitre	
	Leader	Objet Logiciel	Arbitre	
	Distribution des rôles affichée	Information	Joueur	
	Rôle	Information	Joueur	

2.2.3 Description de l'allocation des transitions de contrôle et de l'initiative pour la version du jeu des 15 proposant des recommandations aux joueurs

Le Tableau 35 décrit l'allocation des transitions de contrôles et de l'initiative entre les rôles des acteurs pour la version du jeu des 15 proposant des recommandations aux joueurs. Il n'y a pas de nouvelle transition de contrôle dans cette version du jeu, seulement de nouvelles fonctions coopératives pour le rôle Recommandeur.

Le recommandeur exécute des fonctions coopératives afin de se coordonner avec les autres rôles. Le recommandeur reçoit :

- Le lancement de la partie lorsqu'un des joueurs clique sur le bouton pour lancer la partie
- La fin de tour lorsque le Challenger ou le Leader clique sur le bouton fin de tour
- Le lancement d'une nouvelle partie lorsque le Challenger ou le Leader clique sur le bouton rejouer.

Le recommandeur affiche au Leader et au Challenger les recommandations à chaque tour. Le recommandeur n'est pas à l'initiative de transition de contrôle.

Tableau 35. Description de l'allocation des Transitions de Contrôle et de l'initiative proposant des recommandations aux joueurs

Allocation des transitions de contrôle et de l'initiative				
Transition	Initiative	*Handover* ou émetteur	*Takeover* ou receveur	Tâches et Fonctions

X	Arbitre	Arbitre : « Afficher les jetons disponibles » **HANDOVER**	Leader : « Voir les jetons disponibles »	Fin attente pour le *Leader* et le *Challenger* *Leader* joue son tour *Challenger* attend l'arbitre finit d'afficher le jet et démarre le supporte du tour du *Leader*
X	Arbitre	Arbitre : « Afficher les jetons disponibles » **HANDOVER**	Challenger : « Voir les jetons disponibles »	
X	Arbitre	Arbitre : « Afficher les jetons disponibles désactivés » **TAKEOVER**	Arbitre	
X	Leader	Leader : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »	
X	Challenger	Challenger : « Sélectionner un jeton » **HANDOVER**	Arbitre : « Recevoir le jeton sélectionné »	
	Recommandeur	Recommandeur : « Afficher les recommandations »	Leader : « Voir les recommandations »	
	Recommandeur	Recommandeur : « Afficher les recommandations »	Leader : « Voir les recommandations »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Leader »	Leader : « Voir mes jetons »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Leader »	Challenger : « Voir les jetons du Leader »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Challenger »	Leader : « Voir les jetons du Challenger »	
	Arbitre	Arbitre : « Afficher les jetons joués par le Challenger »	*Challenger : «* « Voir mes jetons »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Leader : « Voir le bouton fin de tour activé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour activé » **HANDOVER**	Challenger : « Voir le bouton fin de tour activé »	
x	Arbitre	Arbitre : « Afficher le bouton fin de tour désactivé » **TAKEOVER**	Arbitre	
x	Leader	Leader : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »	
x	Challenger	Challenger : « Mettre fin au tour » **HANDOVER**	Arbitre : « Recevoir la fin de tour »	
	Leader	Leader : « Mettre fin au tour »	Recommandeur : « Recevoir la fin de tour »	
	Challenger	Challenger : « Mettre fin au tour »	Recommandeur : « Recevoir la fin de tour »	
	Arbitre	Arbitre : « Afficher le joueur courant »	Leader : « Voir le joueur courant »	

	Arbitre	Arbitre : « Afficher le joueur courant »	Challenger : « Voir le joueur courant »	
	Arbitre	Arbitre ; « Afficher la requête d'attente »	Leader : « Voir la requête d'attente »	
	Arbitre	Arbitre ; « Afficher la requête d'attente »	Challenger : « Voir la requête d'attente »	
	Leader	Leader : « Annoncer la fin de la partie »	Challenger : « Entendre le Leader »	Fin des tours de jeu
	Challenger	Challenger : « Annoncer la fin de la partie »	Leader : « Entendre le Challenger »	Fin des tours de jeu
X	Arbitre	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Leader : « Voir le bouton rejouer »	
X	Arbitre	Arbitre : « Afficher le bouton rejouer » **HANDOVER**	Challenger : « Voir le bouton rejouer »	
	Leader	Leader : « Rejouer »	Arbitre : « Recevoir rejouer »	
	Challenger	Challenger : « Rejouer »	Arbitre : « Recevoir rejouer »	
	Leader	Leader : « Rejouer »	Recommandeur : « Recevoir rejouer »	
	Challenger	Challenger : « Rejouer »	Recommandeur : « Recevoir rejouer »	
	Leader	Leader : « Annoncer une nouvelle partie »	Challenger : « Entendre qu'on rejoue »	
	Challenger	Challenger : « Annoncer une nouvelle partie »	Leader : « Entendre qu'on rejoue »	
	Leader	Leader : « Annoncer qu'on arrête de jouer »	Challenger : « Entendre qu'on arrête de jouer »	Fin du jeu des 15
	Challenger	Challenger : « Annoncer qu'on arrête de jouer »	Leader : « Entendre qu'on arrête de jouer »	Fin du jeu des 15
x	Arbitre	Arbitre : « Afficher le bouton lancer la partie » **HANDOVER**	Joueur : « Voir le bouton lancer la partie »	
x	Joueur	Joueur : « Lancer la partie » **HANDOVER**	Arbitre : « Recevoir la pression bouton »	
	Joueur	Joueur : « Lancer la partie »	Recommandeur : « Recevoir init de la partie »	
	Joueur	Joueur : « Annoncer la partie du jeu des 15 »	Joueur : « Entendre le début de la partie du jeu des 15 »	
	Arbitre	Arbitre : « Afficher les rôles pour les joueurs »	Joueur : « Voir l'allocation des rôles »	

2.2.4 Description de l'allocation de la responsabilité pour la version du jeu des 15 proposant des recommandations aux joueurs

Le Tableau 36 décrit l'allocation de la responsabilité entre les rôles des acteurs pour la version du jeu des 15 proposant des recommandations aux joueurs. Le principal changement dans cette version du jeu concernant l'allocation de la responsabilité concerne les recommandations proposées par le recommandeur et leur possible impact sur le choix des joueurs.

Le recommandeur définit des valeurs pour le jeton à sélectionner pour le tour à privilégier et à éviter (fonction : générer les recommandations) et les affichent au Leader et au Challenger

(fonction de sortie coopérative : afficher les recommandations). Ces recommandations peuvent faire dévier le résultat attendu du Leader et du Challenger. Les fonctions « générer les recommandations » et « afficher les recommandations » du recommandeur sont à responsabilité.

Tableau 36. Description de l'allocation de la responsabilité pour le jeu des 15 proposant des recommandations aux joueurs

Allocation de la responsabilité	
Tâche/ Fonction	Résultat attendu possiblement impacté
Leader : « Sélection d'un jeton »	*Leader* : « Posséder 3 jetons dont la somme est égale à 15 »
Tâche motrice Leader : « Sélectionner un jeton »	Leader : « Posséder 3 jetons dont la somme est égale à 15 »
Tâche motrice Challenger : « Sélectionner un jeton »	Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Initialiser les jetons disponibles »	Arbitre : « Un seul jeton est sélection chaque tour »
Fonction de sortie de l'arbitre : « Afficher les jetons disponibles activés »	
Fonction de l'arbitre : « Désactiver les jetons disponibles »	
Fonction de sortie de l'arbitre : « Retirer le jeton sélectionné des jetons disponibles et afficher les jetons disponibles désactivés »	
Fonction du recommandeur : « Générer les recommandations »	Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction du recommandeur : « Afficher les recommandations »	Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Leader »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de sortie de l'arbitre : « Afficher le jeton sélectionné dans les jetons joués par le Challenger »	Arbitre : « Un seul jeton est sélection chaque tour »

	Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Traiter la sélection »	Arbitre : « Un seul jeton est sélection chaque tour » Leader : « Posséder 3 jetons dont la somme est égale à 15 » Challenger : « Posséder 3 jetons dont la somme est égale à 15 »
Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Arbitre : « Les joueurs jouent chacun leur tours »
Fonction de sortie de l'arbitre : « Afficher le bouton fin de tour désactivé »	
Fonction de l'arbitre : « Activer le bouton fin de tour »	
Fonction de sortie : « Afficher le bouton fin de tour disponible »	
Tâche motrice du Leader : « Mettre fin au tour »	
Tâche motrice du Challenger : « Mettre fin au tour »	
Fonction de sortie de l'arbitre : « Afficher le joueur courant »	
Fonction de l'arbitre : « Initialiser le joueur courant »	
Fonction de l'arbitre : « Changer le joueur courant »	
Fonction de l'arbitre : « Allouer les rôles aléatoirement »	Arbitre : « Le jeu est joué avec 2 joueurs »
Fonction de sortie de l'arbitre : « Afficher les rôles des joueurs »	

2.2.5 Description de l'allocation de l'autorité pour la version du jeu des 15 proposant des recommandations aux joueurs

Le Tableau 37 décrit l'allocation de l'autorité entre les rôles des acteurs pour la version du jeu des 15 proposant des recommandations aux joueurs. Le principal changement pour cette version du jeu est que le recommandeur à l'autorité de recommander des jetons au Leader au Challenger.

En effet, le recommandeur à l'autorité de définir des valeurs à privilégier et à éviter pour le jeton à sélectionner pour le tour (fonction : générer les recommandations). Ensuite, le Leader et le Challenger ont l'autorité de sélectionner un jeton en prenant en compte, ou non, la recommandation du recommandeur.

Tableau 37. Description de l'allocation de l'autorité proposant des recommandations aux joueurs

Allocation de l'autorité	
Tâche/Fonction	Contrainte
Arbitre : Désactiver les jetons disponibles	La sélection d'un jeton n'est plus possible à partir de l'exécution de cette fonction
Fonction de l'arbitre : « Activer les jetons disponibles »	Autoriser la sélection d'un jeton

Fonction de l'arbitre : Initialiser les jetons disponibles	Presser le widget pour sélectionner la valeur du jeton sélectionné
Tâche motrice du Leader : « Sélectionner le jeton »	Effectuer la sélection
Tâche motrice du Challenger : « Sélectionner le jeton »	Effectuer la sélection
Fonction du recommandeur : « Générer les recommandations »	Définit des valeurs pour les alternatives de choix possibles pour les joueurs
Tâche de décision du Leader : « Choisir un jeton »	Valeur du jeton à sélectionner
Tâche de décision du Challenger : « Choisir un jeton »	Valeur du jeton à sélectionner
Fonction de l'arbitre : « Désactiver le bouton fin de tour »	Empêcher la pression du bouton fin de tour
Fonction de l'arbitre : « Activer le bouton fin de tour »	Autoriser la pression du bouton fin de tour
Fonction de l'arbitre : « Initialiser le bouton fin de tour »	Bouton à presser pour mettre fin au tour
Tâche motrice du Leader : « Mettre fin au tour »	Mettre fin au tour
Tâche motrice du Challenger : « Mettre fin au tour »	Mettre fin au tour
Fonction de l'arbitre : « Initialiser le joueur courant »	Définir valeur premier joueur
Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Challenger
Fonction de l'arbitre : « Changer le joueur courant »	Définir valeur du prochain joueur sur Leader
Tâche cognitive de décision du Leader : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Tâche cognitive de décision du Challenger : « Décider que la fin de partie est atteinte »	Définir la valeur du résultat de la partie
Initialiser le bouton rejouer	Presser le bouton pour rejouer
Activer le bouton rejouer	Autoriser pression bouton rejouer
Tâche motrice du Leader : « Rejouer »	Lancer une nouvelle partie
Tâche motrice du Challenger : « Rejouer »	Lancer une nouvelle partie
Tâche cognitive de décision du Leader : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche cognitive de décision du Challenger : « Décider de rejouer »	Décide de Rejouer
Tâche cognitive de décision du Leader : « Décider de ne pas rejouer »	Décider d'arrêter
Tâche de groupe du Leader et du Challenger : « Les joueurs se mettent d'accord pour rejouer »	Défini qui presse le bouton pour rejouer
Tâche cognitive d'analyse du Leader : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Tâche cognitive d'analyse du Challenger : « Trouver des alternatives »	Défini un sous-ensemble de valeurs pour le jeton à sélectionner
Fonction de l'Arbitre : « Initialiser le bouton lancer partie »	Presser le bouton lancer partie pour lancer la partie

Fonction de l'Arbitre : « Activer le bouton lancer partie »	Activer bouton lancer partie
Tâche motrice du Joueur : « Lancer la partie »	Lancer la partie
Tâche motrice parler du joueur : « Annoncer la partie du jeu des 15 »	Définir quel joueur lance la partie
Fonction de l'arbitre : « Allouer les rôles aléatoirement aux joueurs »	Définir la valeur du rôle des joueurs

2.2.6 Description de l'allocation RCRA dans les modèles de tâches ségrégués et de fonctions

La Figure 101 présente le modèle de fonctions du rôle recommandeur pour recommander des jetons aux joueurs. L'allocation RCRA des autres rôles n'évolue pas dans cette version du jeu des 15 par rapport à la version précédente du jeu des 15 présentée au Chapitre 4.1..

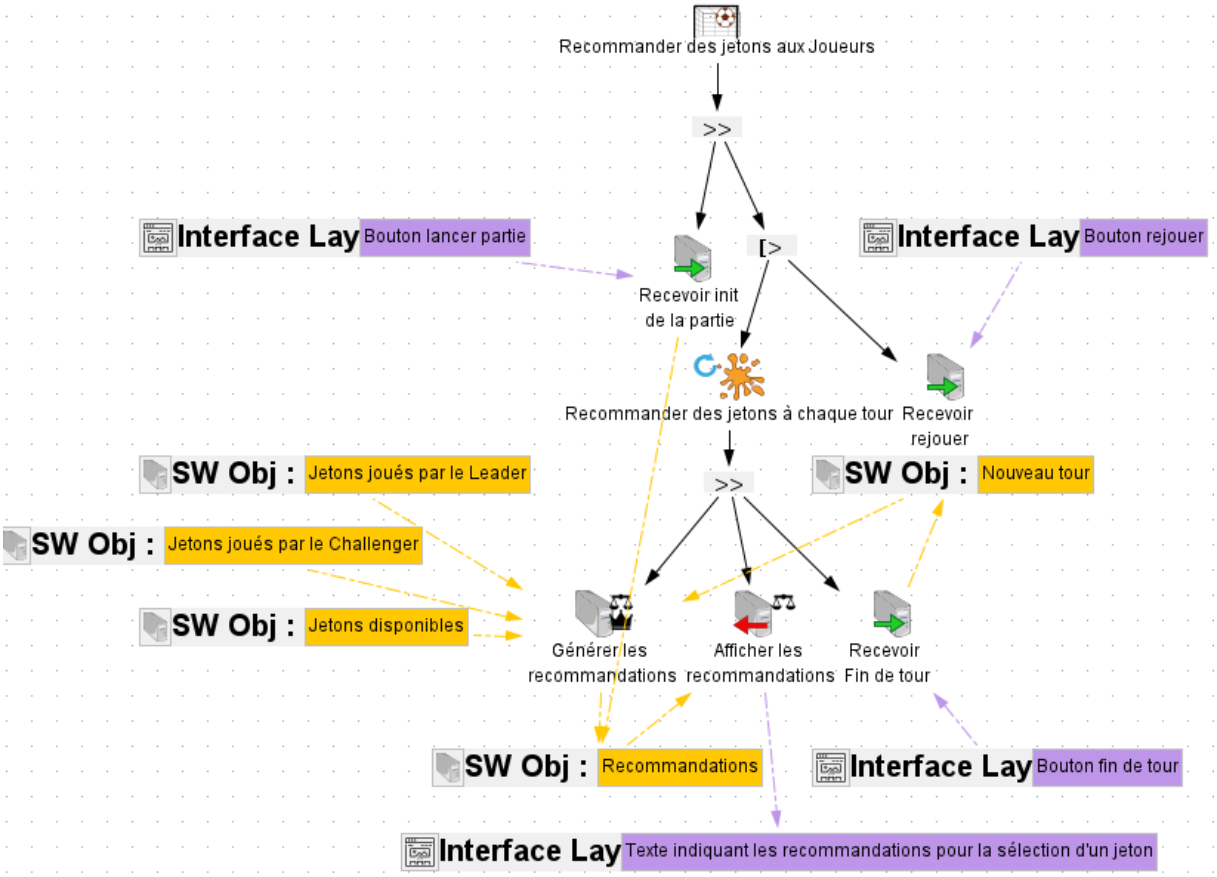


Figure 101. Modèle de fonctions du recommandeur pour recommander des jetons aux joueurs.

2.3 Comparaison des versions du jeu des 15

Dans cette section, nous présentons les différences en termes d'allocation RCRAFT entre le jeu des 15 qui a été présenté au Chapitre 4.1 et la version du jeu des 15 proposant des recommandations aux joueurs.

2.3.1 Comparaison de l'allocation des tâches et des fonctions

Dans cette sous-section, nous présentons la comparaison de l'allocation des tâches et des fonctions entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du des 15 proposant des recommandations aux joueurs.

2.3.1.1 Identification des différences

Un rôle supplémentaire est nécessaire pour version du jeu des 15 proposant des recommandations aux joueurs. Ce rôle, appelé recommandeur, recommande des jetons aux joueurs. Ce rôle alloué à l'acteur système exécute les fonctions suivantes :

- Recevoir l'initialisation de la partie (fonction d'entrée coopérative)
- Recommander des jetons à chaque tour (fonction abstraite itérative)
    - Générer les recommandations (fonction)
    - Afficher les recommandations (fonction de sortie coopérative)
    - Recevoir la fin de tour (fonction d'entrée coopérative)
- Recevoir rejouer (fonction d'entrée coopérative)

Des tâches supplémentaires sont également allouées au Leader et le Challenger pour prendre en compte les recommandations du recommandeur. Ces tâches supplémentaires sont :

- Voir les recommandations (tâche perceptive coopérative)
- Trouver un jeton à jouer visible dans la Figure 98 où la tâche cognitive d'analyse « Trouver des alternatives » a en entrée les recommandations sous forme d'information (pour trouver un jeton pour jouer son tour pour jouer au jeu des 15)
- Voir les recommandations (tâche perceptive coopérative) à la fin du tour du Challenger.

2.3.1.2 Impact

Un nouveau rôle est attribué à l'acteur système : recommandeur. Les rôles Leader et Challenger ont de nouvelles tâches allouées pour voir et prendre en considération les recommandations du recommandeur.

2.3.2 Comparaison de l'allocation des ressources

Dans cette sous-section, nous présentons la comparaison de l'allocation des ressources entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du des 15 proposant des recommandations aux joueurs.

2.3.2.1 Identification des différences

Les nouveaux DODs de la version du jeu des 15 proposant des recommandations aux joueurs sont :

- Objet logiciel : Recommandations
- Information : Recommandations
- Texte indiquant les recommandations pour la sélection d'un jeton : Couche Interface

Dans cette version du jeu des 15, l'acteur système est attribué à un nouveau rôle affichant un nouvel élément graphique le « l'arbitre affiche un nouvel élément le « Texte indiquant les

recommandations pour la sélection d'un jeton » par rapport à la version du jeu des 15 précédente. Le recommandeur communique des valeurs à privilégier ou à éviter pour la sélection d'un jeton pour le tour.

2.3.2.2 Impact

Le Leader et le Challenger peuvent prendre connaissance des conseils donné par l'arbitre pour sélectionner un jeton pendant leur tour.

2.3.3 Comparaison de l'allocation des transitions de contrôle et de l'initiative

Dans cette sous-section, nous présentons la comparaison de l'allocation des transitions de contrôle et de l'initiative entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du jeu des 15 proposant des recommandations aux joueurs.

2.3.3.1 Identification des différences

Il n'y a pas de nouvelles transitions de contrôle pour la version du jeu des 15 proposant des recommandations aux joueurs. Les tâches et fonctions coopératives supplémentaires dans cette version du jeu ne font que communiquer des informations à d'autres rôles ou permettent au recommandeur de se coordonner avec les autres rôles. Ces nouvelles tâches et fonctions coopératives sont :

- Recevoir l'initialisation de la partie (fonction d'entrée coopérative du recommandeur) lors de la coopération permettant de lancer la partie
- Recevoir la fin de tour (fonction d'entrée coopérative du recommandeur) lors de la coopération permettant de mettre fin au tour du Challenger ou du Leader lorsque le Challenger ou le Leader clique sur le bouton fin de tour
- Recevoir rejouer (fonction d'entrée coopérative du recommandeur) lors de la coopération permettant de rejouer une partie lorsque le Challenger ou le Leader clique sur le bouton rejouer.
- Afficher les recommandations (fonction de sortie coopérative du recommandeur) et voir les recommandations (tâche perceptive coopérative du Leader et du Challenger) lors d'une coopération permettant de communiquer les recommandations de jeu pour le tour.

Le recommandeur n'effectue pas de *handover* ou de *takeover*. Le Leader, le Challenger et l'arbitre n'initient pas de nouvelles transitions de contrôle.

2.3.3.2 Impact

Il n'y a pas d'évolution de l'allocation des transitions de contrôle dans cette version du jeu des 15 par rapport à la version précédente du jeu des 15 présentée au Chapitre 4.1.. Le recommandeur est à l'initiative de coopérations pour communiquer aux joueurs les recommandations.

2.3.4 Comparaison de l'allocation de la responsabilité

Dans cette sous-section, nous présentons la comparaison de l'allocation de la responsabilité entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du des 15 proposant des recommandations aux joueurs.

2.3.4.1 Identification des différences

Dans la version du jeu des 15 proposant des recommandations, le recommandeur peut recommander de jouer ou d'éviter certains jetons au Leader ou au Challenger. Les fonctions « générer les recommandations » et « afficher les recommandations » du recommandeur sont à responsabilité. Ces fonctions manipulent les jetons recommandés au Leader et au Challenger. Ces fonctions peuvent faire dévier les résultats attendus du Leader et du Challenger de « Posséder 3 jetons dont la somme est égale à 15 » en influençant le choix du Leader et du Challenger.

2.3.4.2 Impact

Le recommandeur peut être responsable si le Leader ou le Challenger ne gagne pas. Dans la version précédente du jeu présentée au Chapitre 4.1., seuls le Leader et le Challenger réalisaient des tâches et fonctions susceptibles de faire dévier leurs résultats attendus.

2.3.5 Comparaison de l'allocation de l'autorité

Dans cette sous-section, nous présentons la comparaison de l'allocation de l'autorité entre la version du jeu des 15 présentée au Chapitre 4.1 et la version du des 15 proposant des recommandations aux joueurs.

2.3.5.1 Identification des différences

Dans cette version du jeu des 15, le recommandeur à l'autorité de définir des valeurs à privilégier et à éviter pour le jeton à sélectionner pour le tour (fonction : générer les recommandations). Les Joueurs, l'arbitre le Leader et le Challenger n'ont pas de tâches ou de fonctions leur permettant de modifier les recommandations.

Ensuite, le Leader et le Challenger ont l'autorité de sélectionner un jeton en prenant en compte, ou non, la recommandation du recommandeur.

2.3.5.2 Impact

Le rôle recommandeur a l'autorité de conseiller le Leader et le Challenger dans cette version du jeu des 15. Les rôles Joueurs, arbitre Leader et Challenger ont autant d'autorité que dans la version précédente du jeu. Le rôle recommandeur étant alloué à l'acteur système, le système a plus d'autorité dans cette version du jeu des 15 que dans celle présentée au Chapitre 4.1.

2.4 Synthèse des comparaisons

Les analyses et les comparaisons de l'allocation RCRAFT des différentes versions du jeu des 15 montrent que l'ajout d'une seule fonctionnalité à un système peut faire évoluer l'allocation RCRAFT pour chacun des rôles. Le graphique présenté en Figure 102 présente une synthèse quantitative des comparaisons en termes d'allocation RCRAFT des différentes versions du jeu des 15. En concordance avec les critiques exprimées à l'encontre des niveaux d'automatisation, une analyse quantitative de l'allocations des tâches, des fonctions et des éléments RCRAFT donne peu d'indication pour comparer ou caractériser un système contenant de l'automatisation par rapport à un autre. Une analyse qualitative (réalisée en section 1.3 et 2.3) permet de caractériser les différences en termes d'allocation RCRAFT entre les rôles des acteurs selon l'automatisation du système.

La synthèse quantitative permet cependant d'observer que pour la version du jeu des 15 avec un temps limité pour jouer son tour :

- Plus de fonctions et de tâches (19 de plus) sont allouées aux rôles,
- Plus de DODs sont manipulés par les rôles (59 DODs au total pour les joueurs et 22 pour l'arbitre),
- L'arbitre peut initier 4 transitions de contrôle de plus que pour les autres versions du jeu des 15,
- L'arbitre a plus de responsabilité (2 fonctions) et plus d'autorité (3 fonctions) que pour les autres versions du jeu des 15.

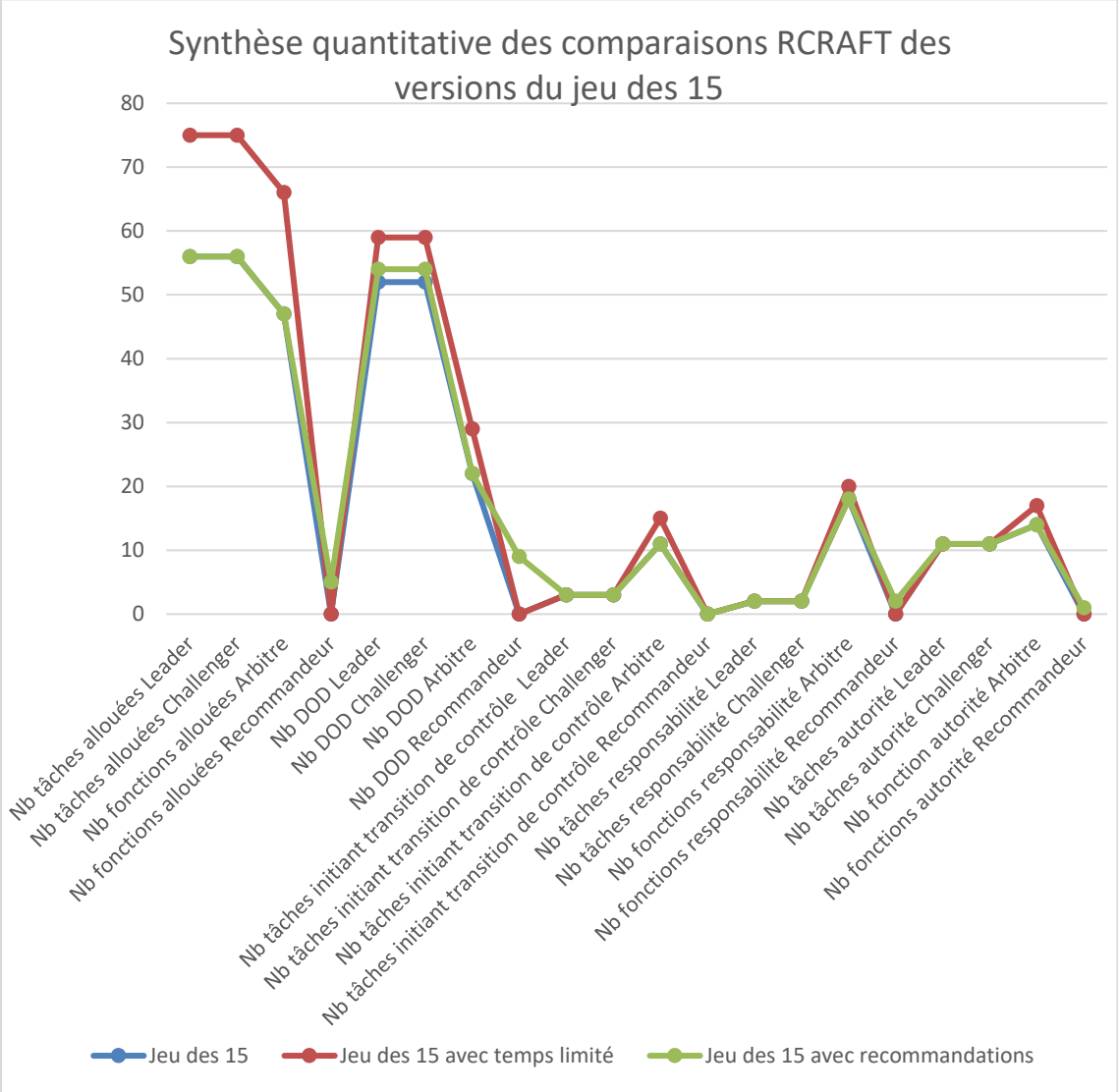


Figure 102. Synthèse des comparaisons RCRAFT des différentes versions du jeu des 15.

3 Conclusion

Dans ce chapitre, nous avons présenté différentes versions de notre exemple illustratif : le jeu des 15. Nous avons présenté une version du jeu des 15 avec un temps limité pour les joueurs pour jouer leur tour et une version recommandant des jetons à privilégier ou à éviter pour le

tour de jeu des joueurs. Puis, nous avons réalisé l'analyse RCRAFT sur ces versions du jeu des 15 en appliquant le processus RCRAFT à ces versions du jeu des 15. Ensuite, nous avons comparé l'allocation RCRAFT du jeu des 15 avec un temps limité pour les joueurs pour jouer leur tour et du jeu des 15 recommandant des jetons avec la version du jeu des 15 présentée au Chapitre 4.1. Nous avons pu montrer les différences en termes d'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité, de l'autorité, des fonctions et des tâches des différentes versions du jeu des 15. Également, nous avons pu décrire de manière explicite et non ambigüe ces différentes allocations au niveau des tâches et des fonctions dans les modèles de tâches ségrégués et de fonctions des rôles pour ces exemples simples. Dans le prochain chapitre, nous présentons l'application du processus RCRAFT à une étude de cas industrielle.

Chapitre 9. Etude de cas : Analyse RCRAFT de systèmes de gestions des alarmes des cockpits AIRBUS

Ce chapitre présente la mise en œuvre de notre approche sur une étude de cas produite dans le cadre du projet de recherche IKKY (Intégration du cocKpit et de ses sYstèmes) impliquant des experts de l'entreprise AIRBUS. Nous montrons la faisabilité de notre approche par son application à une étude de cas industrielle. Cette étude de cas porte sur les systèmes de gestion des alarmes. Dans le cadre de ce projet de recherche, nous avons proposé le prototype d'un nouveau système de gestion des alarmes. Le prototype proposé a pour but de faciliter la prise en compte du choix du pilote pour la résolution des alarmes. En effet, le système actuel de gestion des alarmes étudié donne la possibilité au pilote de supprimer certaines alarmes mais l'utilisation de cette fonctionnalité doit rester exceptionnelle et le système peut refuser la suppression en cas d'alarmes non-supprimables. Cependant, une alarme peut toujours être effacée, même s'il s'agit d'une alarme non supprimable, en l'effaçant après avoir demandé sa suppression. Avec les données issues du travail réalisé lors du projet IKKY nous avons :

- Appliqué le processus RCRAFT au système actuel et au prototype proposé et, ainsi, décrit explicitement les caractéristiques de ces automatisations selon le cadre conceptuel RCRAFT
- Comparé le système actuel et le prototype proposé selon le cadre conceptuel RCRAFT et, ainsi, caractérisé les différences au niveau des tâches et des fonctions des rôles que provoque une modification du paradigme de gestion des alarmes.

Nous présentons les artefacts produits à l'issue des différentes phases et sous-phases du processus RCRAFT appliquées à l'analyse de l'automatisation présente dans le système de gestion des alarmes existant et appliquées à l'analyse de l'automatisation présente dans le prototype produit pendant le projet.

La première section de ce chapitre présente le projet de recherche IKKY et le lot de travail MAGIC (modélisation de l'état opérationnel de l'avion et de ses défaillances).

La seconde section de ce chapitre présente le système actuel étudié (*flight warning system*), puis l'analyse RCRAFT du *flight warning system* pour la gestion des alarmes avec le pilote *monitoring*.

La troisième section de ce chapitre présente le prototype du système de gestion des alarmes proposé et l'analyse RCRAFT du prototype de système alternatif de gestion des alarmes pour la gestion des alarmes avec le pilote *monitoring*.

La quatrième section de ce chapitre présente la comparaison de ces deux automatisations par rapport aux éléments de RCRAFT.

La cinquième section conclut ce chapitre.

1 Présentation du projet de recherche IKKY

Le projet IKKY (Intégration du cocKpit et de ses sYstèmes) est un projet de recherche financé par le CORAC (Conseil pour la Recherche Aéronautique Civile). La mise en place de ce conseil en 2008 résulte de l'idée de joindre les efforts de recherche et d'innovation dans le domaine aéronautique. Ce projet s'inscrit dans la volonté de repenser les cockpits d'aviation civil du futur pour les moderniser tout en gardant un niveau de sureté de fonctionnement maximal.

Le lot de travail 6.3 MAGIC (modélisation de l'état opérationnel de l'avion et de ses défaillances) du projet IKKY a pour but de définir des notations, méthodes et outils pour la conception d'application interactives devant être déployées dans des cockpits pour la gestion contextuelle de l'état de l'avion. Ce projet est mené en partenariat avec Airbus (service R&T Opérations Cockpit) et l'équipe ICS (http://www.irit.fr/recherches/ICS/index.html) de l'Institut de Recherche en Informatique de Toulouse IRIT.

La complexité croissante des systèmes de l'avion augmente le nombre de pannes potentielles sur ces systèmes et, avec elle, le nombre d'alarmes et d'actions de récupération à réaliser par les pilotes. Le Flight Warning System (FWS) a pour but d'aider et de guider les pilotes dans la gestion de la complexité des systèmes lorsqu'ils sont en panne. Nous avons étudié un système alternatif au système de gestion des alarmes actuel. Ce système alternatif a comme objectif principal de donner l'autorité du choix des actions de récupération aux pilotes, en particulier pour les situations de pannes multiples dont des éléments contextuels non détectés par l'automatisation peuvent orienter le choix des actions de récupération à réaliser. Nous avons prototypé ce système alternatif. Afin de comparer le système existant et le prototype du système alternatif de gestion des alarmes, nous avons appliqué le processus d'analyse de l'allocation RCRAFT. L'application de ce processus nous permet de décrire l'allocation des tâches et des fonctions et l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre le système actuel et le pilote, puis entre le prototype proposé et le pilote. Ainsi, il est possible de comparer les automatisations de ces deux versions de système par rapport aux concepts RCRAFT.

Dans ce chapitre, nous présentons les modèles de tâches ségrégués et de fonctions décrivant l'allocation de tâches et de fonctions entre le pilote monitoring et les systèmes étudiés. Nous présentons également le tableau RCRA décrivant l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre le pilote *monitoring* et les systèmes étudiés. Par soucis de lisibilité, les modèles de tâches ségrégués et de fonctions complets sont présentés en annexes. Pour la même raison, les colonnes complètes du tableau RCRA décrivant l'allocation des ressources sont présentées en annexes et un extrait représentatif de nos objectifs d'analyse est présenté dans ce chapitre.

2 Système de gestion des alarmes actuel

Le *Flight Warning System* (FWS) est l'actuel système en charge de la gestion des alarmes. Le FWS fait partie du système ECAM (*Electronic Centralized Aircraft Monitoring*) visible au centre de la Figure 103 encadré en rouge.

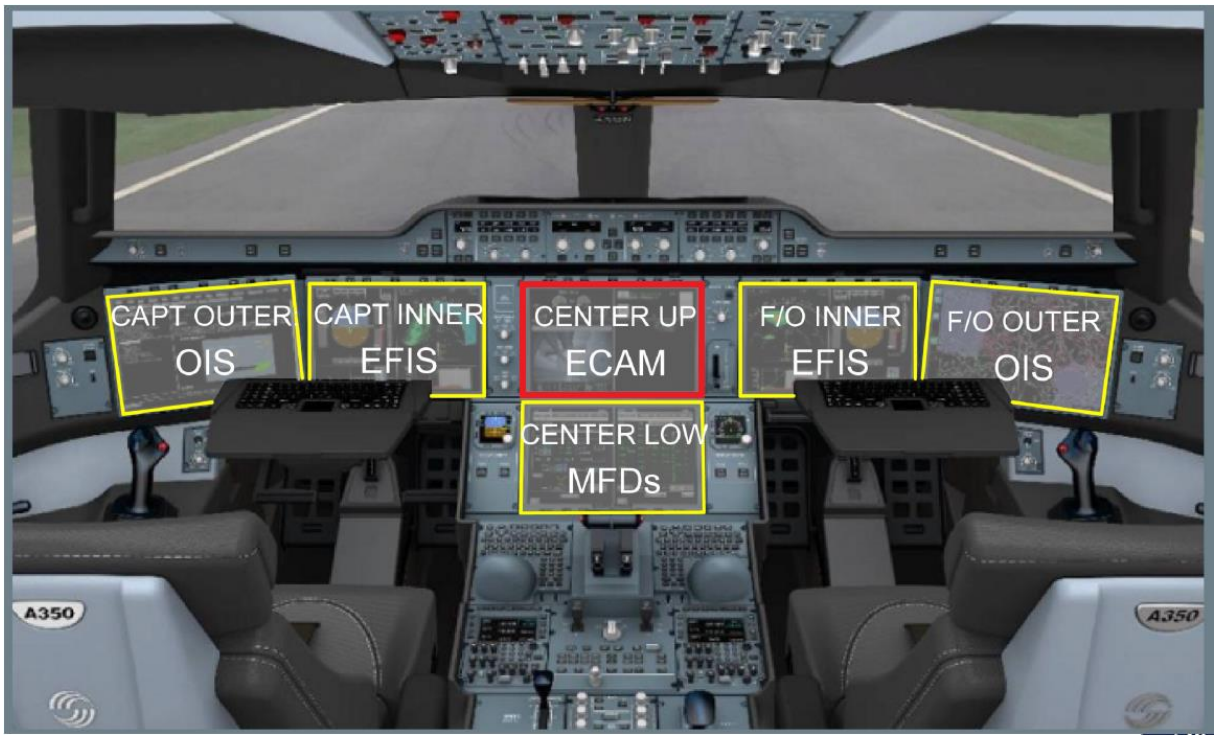


Figure 103. Cockpit de l'A350. L'ECAM est encadré en rouge.

2.1 Présentation du « Flight Warning System »

Le FWS sélectionne l'alarme la plus prioritaire et d'autres informations (alarmes actives, mémos, limitations, …) à afficher sur les différents écrans selon le contexte courant. Comme le montre la vue schématique de ce système en Figure 105. a., le FWS calcule la ou les alarmes actives, puis filtre ces alarmes (et leurs procédures associées de même nom) selon différents inhibiteurs. Ces inhibiteurs peuvent être la phase de vol (croisière, atterrissage, approche, etc.) ou l'état des autres systèmes de l'avion, par exemple. Ceci permet de sélectionner et d'afficher la procédure de l'alarme la plus prioritaire et les autres informations pertinentes pour les pilotes selon le contexte courant.

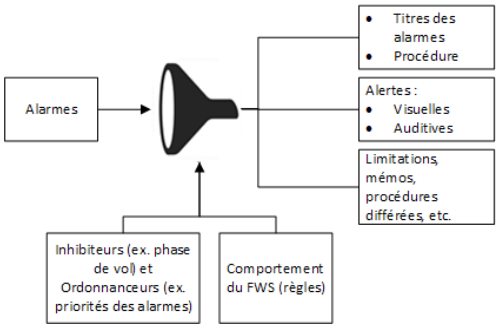


Figure 105. a. Filtrage effectué par le *Flight Warning System*.

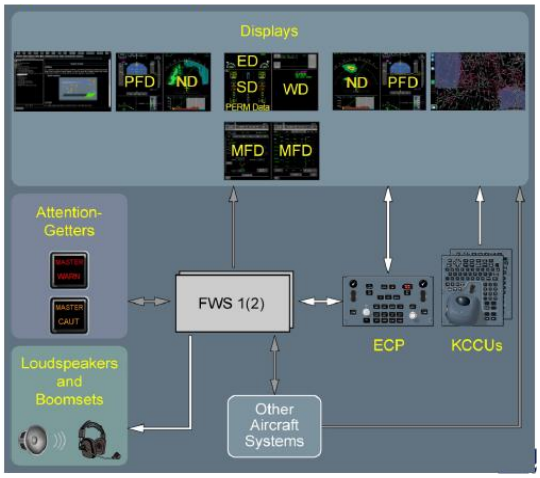


Figure 105. b. Architecture de l'ECAM - Lien du FWS avec d'autres systèmes du cockpit

Le FWS communique avec les autres systèmes de l'ECAM comme l'illustre la Figure 105. b. :

- Le FWS affiche sur le WD (*Warning Display*) les procédures, les limitations et les mémos. Les alarmes sont affichées dans l'ordre de leur priorité. Seule la procédure associée à l'alarme la plus prioritaire est affichée. La procédure suivante est affichée lorsque la première est terminée et a été effacée. Les procédures sont réalisables séquentiellement.
- Le PFD (*Primary Flight Display*) duplique les mémos et les limitations.
- Le FWS affiche sur le SD (*System Display*) les informations sur le/les systèmes impactés.
- Le FWS contrôle également les *Attention-getters* (au milieu à gauche sur la Figure 105. b.) : il allume le bouton *Master Caution* en cas d'avertissement (alarme ambre) et il fait clignoter le bouton *Master Warning* en cas d'alerte (alarme rouge) visibles sur la Figure 106.
- Certaines alarmes déclenchent des alertes sonores comme un son répétitif ou une voix synthétisée. Pour cela, le FWS contrôle les haut-parleurs et les casques audio.

Figure 106. Les boutons MASTER Warning (à gauche) et MASTER Caution (à droite) allumés.

2.2 Procédures de résolution des alarmes

Dans cette sous-section, nous détaillons les priorités des alarmes et des procédures associées et leur gestion. Nous finissons avec une présentation succincte du panneau plafond et des pages systèmes (SD).

2.2.1 Priorités des alarmes

Le FWS affiche sur le WD les alarmes selon leur niveau d'alerte. La Figure 107 présente la procédure de l'alarme ENG ALL ENGs FLAME OUT de niveau 3. Il existe trois niveaux d'alertes pour les alarmes.

- Niveau 3 : Urgence. Le titre de l'alarme est rouge. Lorsqu'une alarme de ce niveau est levée, elle est accompagnée d'une alerte sonore et le bouton *Master Warning* est allumé. Ce niveau d'alerte indique que l'avion est dans une configuration dangereuse ou bien dans une condition de vol limité et que la défaillance à l'origine de l'alarme impacte la sécurité du vol. L'équipage doit immédiatement prendre connaissance de l'alarme et y apporter une réponse dans les plus brefs délais.
- Niveau 2 : Alerte. Le titre de l'alarme est ambre. Lorsqu'une alarme de ce niveau est levée, elle est accompagnée d'une alerte sonore et le bouton *Master Caution* est allumé. L'équipage doit immédiatement prendre connaissance de l'alarme et y apporter une réponse pour éviter une dégradation du système impacté. La procédure associée à une alarme de niveau 2 peut ne pas contenir d'actions de récupération (c.à.d. la procédure peut être vide).
- Niveau 1 : Le titre de l'alarme est ambre. Lorsqu'une alarme de ce niveau est levée, il n'y a pas d'alerte sonore et le bouton *Master Warning/Caution* reste éteint. L'équipage doit prendre connaissance de l'alarme et peut y apporter une réponse. La procédure associée à une alarme de niveau 1 peut ne pas contenir d'actions de récupération (c.à.d. la procédure peut être vide).

Les alarmes les plus prioritaires sont de niveau 3 et sont affichées en premières. Puis viennent les alarmes de niveau 2, puis 1. Si plusieurs alarmes de même niveau sont actives, elles sont affichées dans l'ordre de leur priorité. Cette priorité a été définie statiquement pour chacune des alarmes lors de leur conception. Par exemple, l'alarme ENG ALL ENGs FLAME OUT de niveau 3 est de priorité 20 et est levée lorsque les deux moteurs s'éteignent ou échouent en vol. L'alarme de même niveau CAB PRESS EXCESS CAB ALT de priorité 5 est levée lorsque l'altitude de la cabine est trop importante. Si ces deux alarmes sont actives, l'alarme CAB PRESS EXCESS CAB ALT sera affichée avant l'alarme ENG ALL ENGs FLAME OUT.

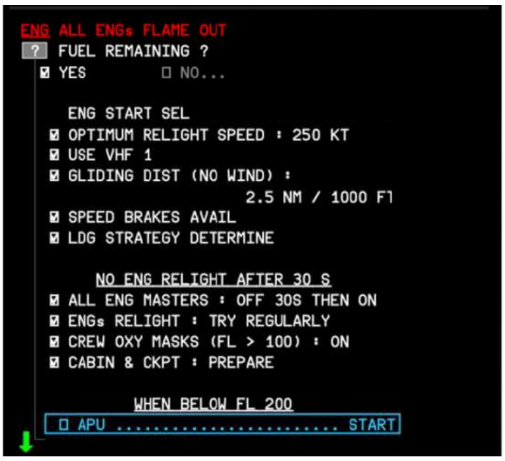


Figure 107. Procédure ALL ENGs FLAME OUT affichée sur le WD.

2.2.2 Supprimer une alarme intempestive

La procédure associée à l'alarme la plus prioritaire est automatiquement affichée sur le WD. Le pilote doit d'abord finir cette première procédure affichée pour pouvoir l'effacer et réaliser la suivante. Cependant, il est possible pour le pilote de supprimer l'alarme la plus prioritaire grâce

au bouton EMER CANCEL. Le bouton EMER CANCEL est situé sur l'ECP (*ECAM Control Panel*) visible dans la Figure 108 (bouton gardé en haut à droite de l'ECP). Le comportement du système après la pression du bouton EMER CANCEL diffère selon le niveau de l'alarme :

- Niveau 3 : La procédure associée à une alarme de niveau 3 n'est pas supprimable avec le bouton EMER CANCEL.
- Niveau 2 et 1 : Les procédures associées aux alarmes de niveau 2 et 1 sont supprimables avec le bouton EMER CANCEL.

Le pilote peut également effacer (CLEAR) la procédure sans effectuer les actions de récupération.

2.2.3 Effacer une procédure et valider les lignes de procédures

Le bouton CLEAR, visible sur la Figure 108 (en bas à gauche et à droite de l'ECP), permet d'effacer une procédure. Le bouton est allumé et une barre d'effacement apparait à l'écran (WD) lorsque l'effacement d'une procédure est possible. Le pilote efface une procédure lorsque qu'il a terminé toutes les actions de récupération de la procédure. Les actions de récupération associées aux lignes de la procédure peuvent être automatiquement détectées par le système lorsqu'elles sont réalisées. Dans ce cas, le système change la couleur de la ligne de procédure de cyan à blanc et le rectangle cyan de désignation se déplace sur la ligne de procédure suivante (voir Figure 107). Lorsque l'action de récupération associée à la ligne de procédure n'est pas automatiquement détectée par le système lorsqu'elle est réalisée, une case à cocher est présente au début de la ligne de procédure (voir Figure 107). Lorsque le pilote a terminé l'action de récupération, il coche la ligne de procédure sélectionnée associée grâce au bouton de validation de l'ECP (bouton rond avec une coche en haut à droite et à gauche de l'ECP sur la Figure 108).

2.2.4 Naviguer parmi la procédure et les alarmes

Le pilote peut faire défiler la procédure affichée et les autres alarmes moins prioritaires pour les lire grâce aux molettes de l'ECP (à droite et à gauche de l'ECP sur la Figure 108). Lorsque le défilement est possible une flèche verte pointant vers le bas indique que des lignes sont masquées. La procédure présentée sur la Figure 107 est trop longue pour que toutes les lignes de procédures soient visibles à l'écran. La flèche verte indique que des lignes de procédure supplémentaires ne sont pas affichées sur le WD. Le pilote peut utiliser la molette de l'ECP pour consulter les lignes de procédures.

2.2.5 Rappeler la dernière ou toutes les procédures et alarmes

Le pilote peut rappeler des alarmes et les procédures associées qu'il a effacées si les conditions pour que ces alarmes soient actives sont toujours présentes. Le bouton RCL LAST visible sur la Figure 108 (à gauche sur l'ECP) permet au pilote de rappeler la dernière alarme (et procédure associée) effacée. L'alarme est rappelée seulement si les conditions de déclenchement de l'alarme sont toujours présentes. Par exemple, si toutes les actions de récupérations de la procédure ENG ALL ENGs FLAME OUT ont été réalisées et que la procédure a été effacée mais que les deux moteurs sont toujours éteints, le pilote peut presser le bouton RCL LAST pour rappeler l'alarme et la procédure associée. Le bouton RCL ALL visible sur la Figure 108 (à droite en dessous du bouton gardé EMER CANCEL sur l'ECP) permet de rappeler toutes les

alarmes effacées dont les conditions de déclenchement sont toujours présentes. Si le bouton est pressé plus de trois secondes, les alarmes supprimées avec EMER CANCEL, inhibées par la phase de vol ou effacées sont rappelées. La procédure de l'alarme la plus prioritaire est affichée sur le WD.

2.2.6 Les pages systèmes

Lorsque des alarmes sont activées, la page du système impacté par l'alarme la plus prioritaire est automatiquement affichée sur le SD. Le pilote peut choisir de consulter les pages des autres systèmes grâce aux boutons gérant l'affichage de la page SD (boutons encadrés en pointillés jaunes sur la Figure 108). Chacun des boutons correspond à un système ou un groupe de systèmes. Par exemple, si le pilote veut afficher l'état du système de pressurisation de l'avion, il presse le bouton PRESS de l'ECP (bouton en bas à droite des boutons encadrés en pointillé jaune de l'ECP sur la Figure 108) pour que la page SD associée au système de pressurisation s'affiche sur le SD.

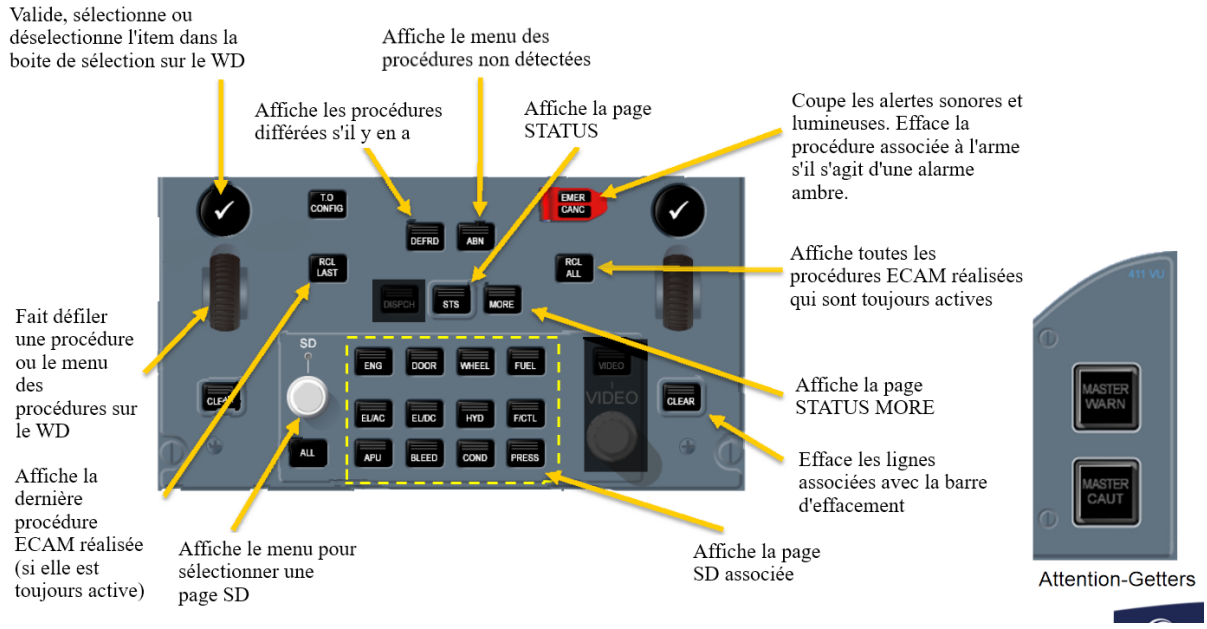


Figure 108. Description de l'ECP ("ECAM Control Panel")

2.3 Panneau plafond

Le panneau plafond, présenté à gauche de la Figure 109, permet aux pilotes de contrôler les différents systèmes de l'avion. Le panneau plafond est situé au-dessus des pilotes comme visible à droite de la Figure 109. Le panneau plafond présente des informations redondantes avec l'ECAM sur l'état des systèmes. Pour valider une alarme, les pilotes doivent vérifier si les états présentés à l'ECAM et sur le panneau plafond sont cohérents. Certaines actions de récupération décrites dans les lignes de procédure demandent aux pilotes d'utiliser les boutons du panneau plafond pour réaliser ces actions de récupération. Par exemple, la ligne de procédure sélectionnée dans la Figure 107 demande au pilote d'allumer l'APU (l'unité de puissance auxiliaire). Pour cela, le pilote doit presser le bouton MASTER SW PB-SW sur le panel de l'APU présent sur le panneau plafond.

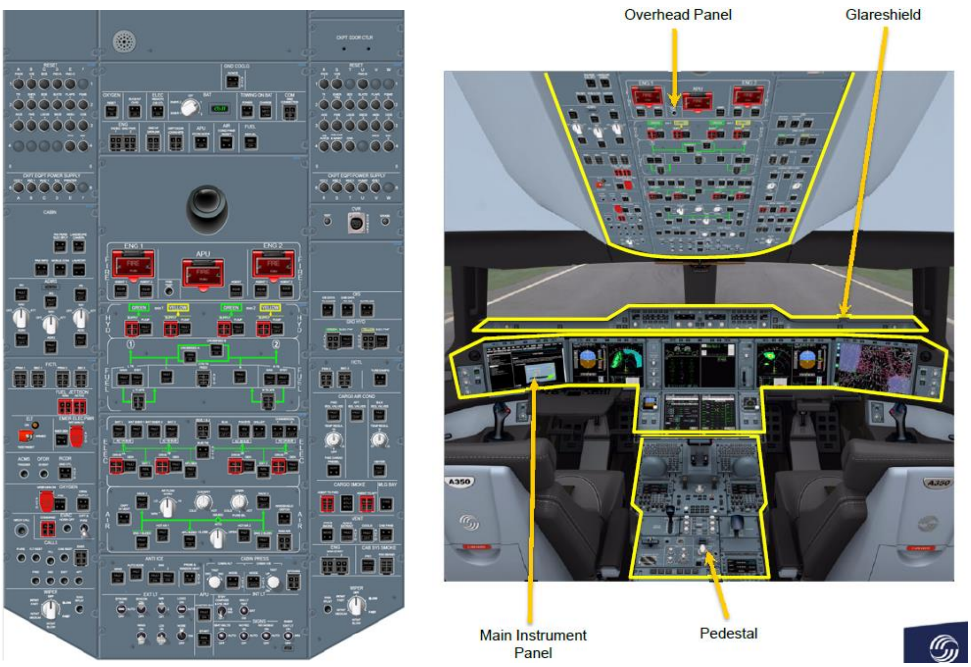


Figure 109. Le panneau plafond à gauche et son emplacement dans le cockpit de l'A350.

2.4 Rôles de pilote *flying* et de pilote *monitoring*

Les pilotes se répartissent deux rôles distincts : le rôle de pilote *flying* et le rôle de pilote *monitoring*. Chaque rôle s'occupe d'actions spécifiques à certains buts pour assurer la mission : VOL, NAVIGUER (au sens de gérer la mission), COMMUNIQUER, et GERER LES SYSTEMES. Le pilote *flying* a les commandes de l'avion. Il s'occupe de faire voler l'avion, de naviguer et de la communication. Le pilote *monitoring* supervise l'état des systèmes. Il s'occupe de gérer et superviser les systèmes et de gérer la communication. Le pilote *monitoring* sous la supervision du pilote *flying* réalise les actions de récupération d'une procédure lorsqu'une ou des alarmes sont actives.

2.5 Analyse RCRAFT de la gestion des défaillances du pilote « monitoring » avec le « flight warning system »

Dans cette section, nous réalisons l'analyse RCRAFT sur la gestion des alarmes dont la *concept map* est présentée en Figure 110. Nous étudierons l'allocation des tâches et fonctions, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité entre un acteur humain et un acteur système ayant les rôles de pilote *monitoring* (PM) et de *flight warning system* (FWS). Les rôles de chacun des acteurs est défini pour le vol.

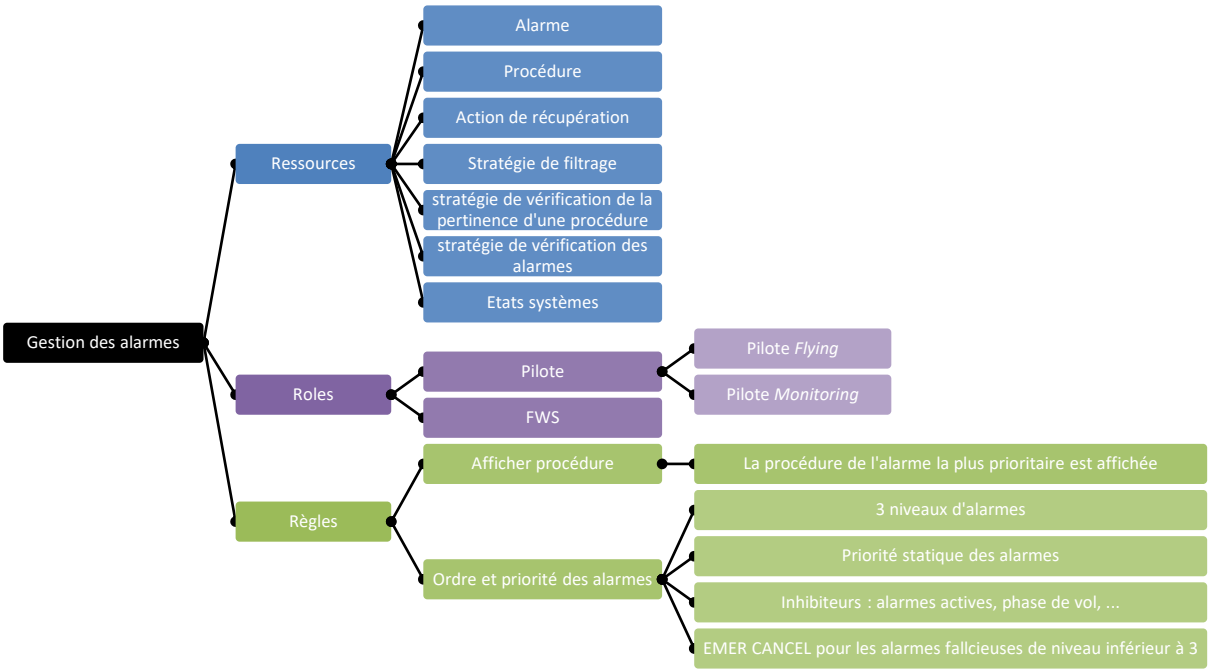


Figure 110. Concept map de la gestion des alarmes.

2.5.1 Modélisation des tâches du pilote monitoring et des fonctions du FWS

Cette section présente les artefacts produits lors de la phase 1 du processus RCRAFT rappelé dans la Figure 111 :

- Les modèles de tâches ségrégués du pilote *monitoring*,
- Les modèles de fonctions du FWS

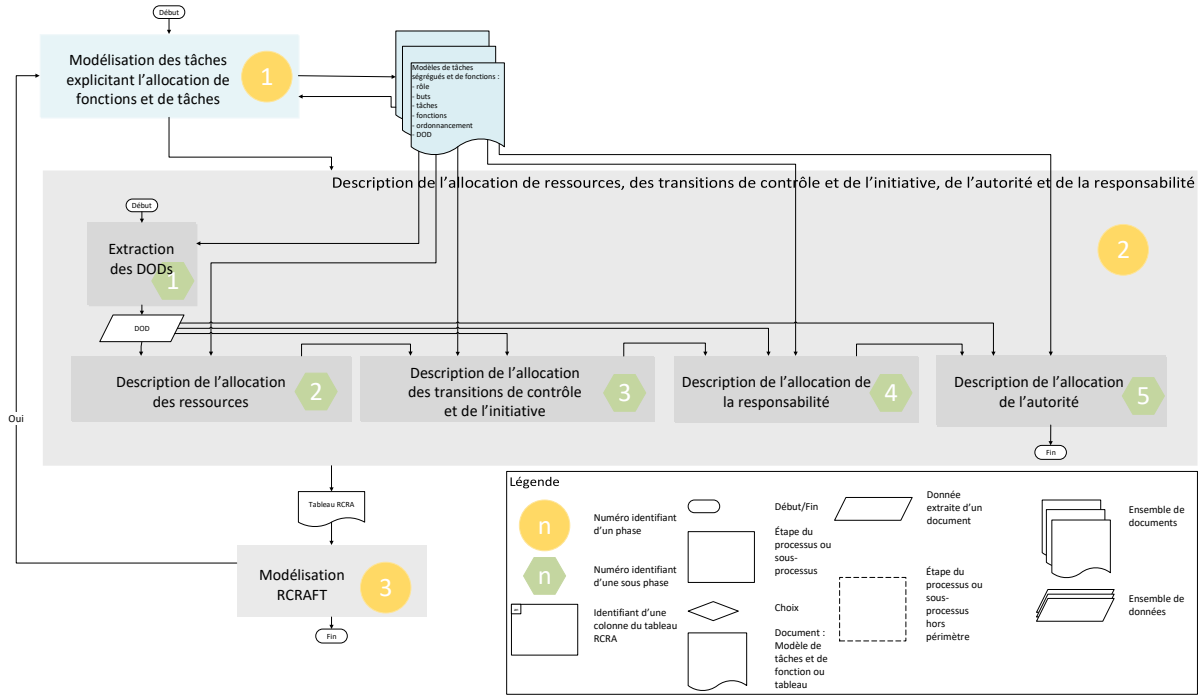


Figure 111. Vue haut-niveau du processus RCRAFT avec les artefacts modèles de tâches et de fonctions et la phase 1 mis en valeur.

2.5.1.1 Phase 1 du processus RCRAFT : Modélisation des fonctions du FWS

La Figure 112 présente le modèle de fonctions haut-niveau du FWS pour gérer les alarmes produit lors de la phase 1 du processus RCRAFT (voir Chapitre 6). La documentation ne décrit pas le FWS comme étant une entité ayant un rôle de gestion des alarmes. La modélisation présentée ci-après est le résultat de l'analyse de la documentation en y joignant les données apportées par les experts Airbus dans le but d'analyser l'automatisation. Les modèles de fonctions décrivant le détail des fonctions du FWS sont présentés en Annexes 1.2. La gestion des alarmes débute par la réception du signal d'une défaillance que le FWS va traduire en alarme.

- Fonction abstraite « Recevoir une/des nouvelles alarmes » : Un premier filtrage est effectué à ce niveau, le FWS filtre l'affichage des alarmes en fonction de la phase de vol courante et combine les alarmes quand cela est possible.
- Fonction abstraite « Afficher les alarmes » : Ensuite, le FWS affiche les alarmes aux pilotes. Pour afficher les alarmes, le FWS filtre les procédures différées et ordonnance les alarmes par niveau et priorité. Le FWS sélectionne la première procédure, puis l'affiche. Le FWS affiche sous la procédure les autres alarmes rouges et ambres actives par ordre de priorité. Le FWS affiche la page système associée à l'alarme la plus prioritaire et lance les *attention getters* (son et lumière d'alerte). Les boutons EMER CANCEL, CLEAR, Master W/C et de validation sont actifs.
- Fonction abstraite « Soutenir l'acquittement des alarmes » : Le FWS supporte l'acquittement des alarmes par le pilote *monitoring*. Le FWS coupe les *attention getters*.
- Fonction abstraite optionnelle « Soutenir le défilement » : Le FWS permet le défilement de la procédure et des alarmes actives affichées sur le WD.
- Fonction abstraite « Soutenir la réalisation de la première procédure » : Le FWS supporte la réalisation de la procédure affichée par le pilote monitoring. Il marque comme réalisées les lignes de procédure décrivant des actions de récupération détectées par le système ou les lignes de procédures cochées par le pilote *monitoring.* Également, le FWS indique l'action de récupération suivante pour le pilote *monitoring*.
- Fonction abstraite optionnelle « Soutenir la sélection d'une procédure » : Le pilote monitoring peut décider d'accéder à une autre procédure que celle affichée par le FWS (interruption la fonction abstraite « Soutenir la réalisation de la première procédure »). Le FWS écoute la pression des boutons Recall Last, Recall All et EMER CANCEL et CLEAR et rappelle ou supprime les alarmes selon les actions du pilote *monitoring*.

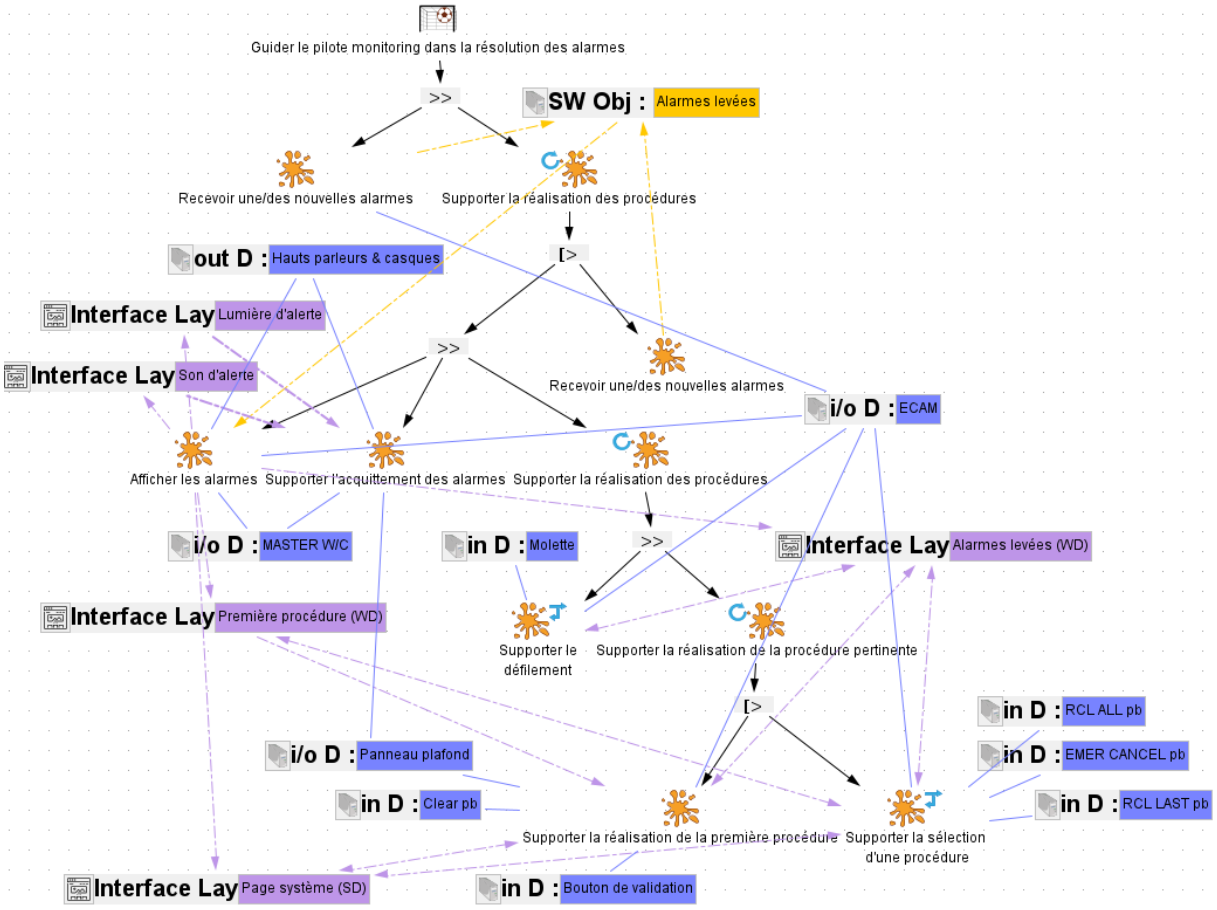


Figure 112. Modèle de fonctions haut-niveau du FWS pour gérer les alarmes

2.5.1.2 Phase 1 du processus RCRAFT : Modélisation des tâches du pilote Monitoring

La Figure 113 présente le modèle de tâches ségrégué haut-niveau du pilote *monitoring* pour résoudre les alarmes produit lors de la phase 1 du processus RCRAFT (voir Chapitre 6). Dans notre étude de cas, nous détaillons seulement les tâches du pilote *monitoring*. Les modèles de tâches décrivant le détail des tâches du pilote *monitoring* sont présentés en Annexes 1.1. Les tâches du pilote *monitoring* sont décrites ci-après :

- Tâche abstraite « Acquitter et vérifier les alarmes » : Après que le FWS ait levé les alarmes et les attention getters, les pilotes commencent par acquitter l'alarme en pressant le MASTER W/C. Ensuite, les pilotes vérifient que l'alarme n'est pas intempestive. Pour cela, ils comparent la page système affichée (SD) avec les états des systèmes indiqués sur le panneau plafond.
- Tâche abstraite « Consulter la procédure et les alarmes » : Ensuite, le pilote *monitoring* peut faire défiler la procédure affichée et les alarmes grâce à la molette de l'ECP.
- Tâche abstraite optionnelle « Accéder à la procédure pertinente » : Si la procédure pertinente pour le pilote *monitoring* est différente de la procédure affichée par le FWS, le pilote *monitoring* peut accéder à la procédure qu'il juge plus pertinente en supprimant et en rappelant des alarmes.

- Tâche abstraite « Réaliser la première procédure » : Ensuite, le pilote *monitoring* réalise la procédure. Le pilote *monitoring* annonce, réalise les actions de récupération, coche les lignes de procédure si besoin et efface la procédure lorsqu'elle est terminée.
- Tâche abstraite optionnelle « Choisir de réaliser une autre procédure » : Le pilote *monitoring* peut interrompre la réalisation des procédures en décidant d'en choisir une autre.
- Interruption de la tâche abstraite itérative « Réaliser les procédures » par la tâche abstraite « Acquitter les alarmes » : Le pilote *monitoring* peut également être interrompu par l'arrivée d'une nouvelle alarme qu'il devra acquitter.

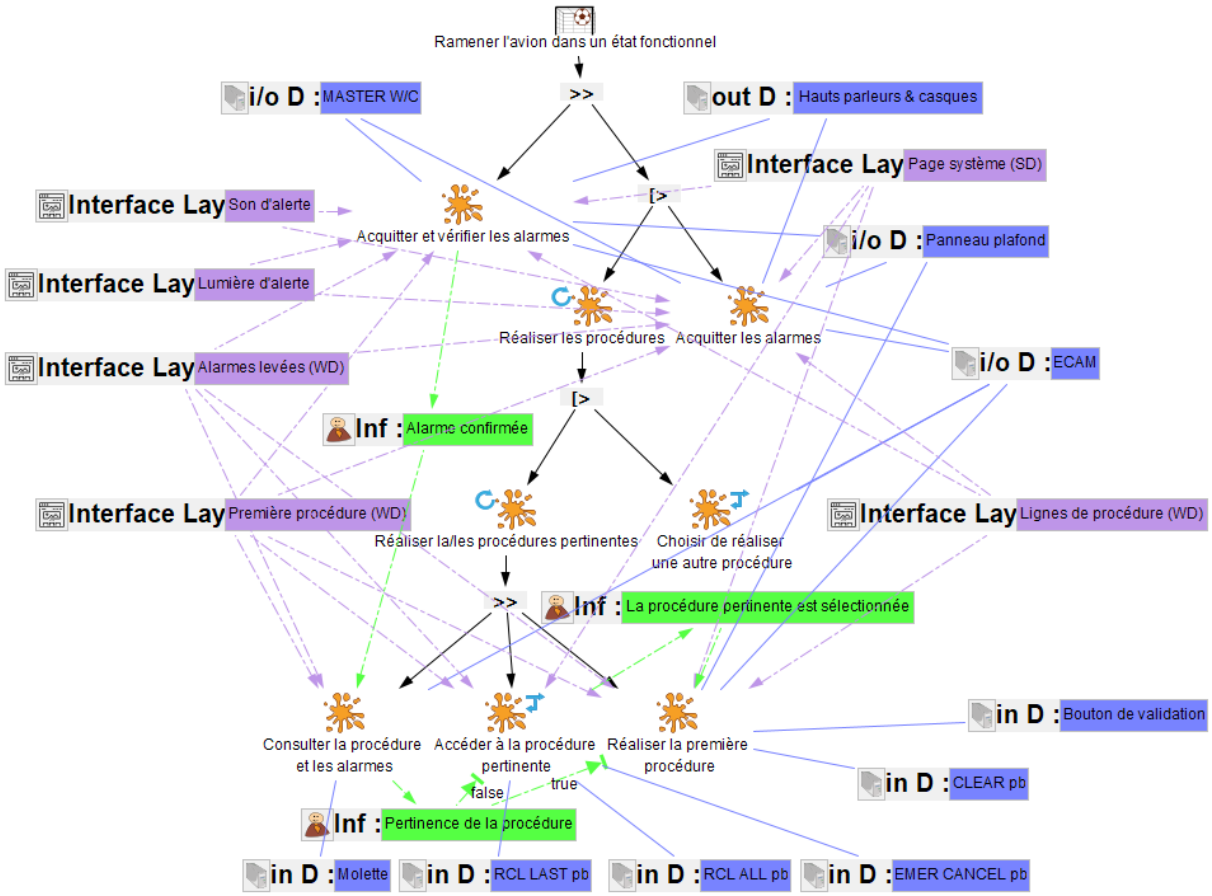


Figure 113. Modèle de tâches ségrégué haut-niveau du pilote monitoring pour résoudre les alarmes et ramener l'avion dans un état fonctionnel

2.5.2 Allocation des ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité entre le pilote monitoring et le FWS

Cette section présente l'artefact produit lors de la phase 2 du processus RCRAFT rappelé dans la Figure 114 : le tableau RCRA. Ce tableau décrit l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre le pilote *monitoring* et le FWS.

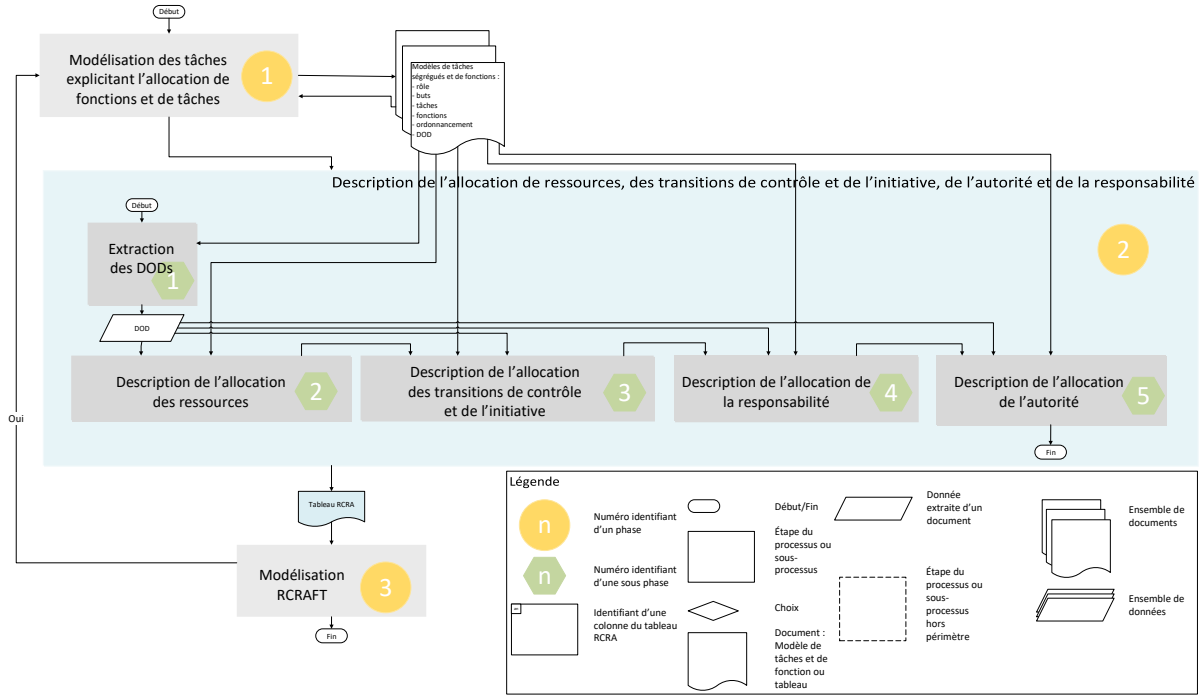


Figure 114. Vue haut-niveau du processus RCRAFT avec l'artefact : tableau RCRA et la phase 2 mis en valeur.

2.5.2.1 Phase 2 du processus RCRAFT : Allocation des ressources

Les ressources utiles à la résolution des alarmes sont : les alarmes, la stratégie de filtrage, la stratégie de vérification des alarmes, les procédures, la stratégie de vérification de la pertinence de l'alarme associée à une procédure, les actions de récupération et les états systèmes. Le Tableau 38 présente un extrait du tableau RCRA décrivant l'allocation des ressources entre le FWS et le pilote *monitoring* produit à la fin des itérations nécessaires (pour traiter tous les DODs des modèles) de la sous-phase 2 de la phase 2 du processus RCRAFT (voir Chapitre 6). Le tableau décrivant l'allocation complète des ressources entre le pilote *monitoring* et le FWS est consultable en annexe (Tableau 46). Les ressources : alarmes, procédures, actions de récupération et systèmes sont partagées entre les deux rôles. Par exemple, les alarmes sont partagées par le FWS au pilote par une lumière d'alerte sur le MASTER W/C, les sons d'alerte à travers les hauts parleurs & casques et les titres des alarmes affichés sur le WD. La stratégie de filtrage des alarmes et des procédures associées n'est pas partagée par le FWS avec le PM. La stratégie de vérification des alarmes n'est pas partagée par le pilote monitoring avec le FWS. La stratégie de vérification de la pertinence de la procédure n'est pas partagée par le PM avec le FWS.

Tableau 38. Extrait des colonnes du tableau RCRA décrivant l'allocation des ressources entre le FWS et le PM pour la résolution des alarmes.

Allocation des ressources				
Ressource [Partagée]	Description DOD	Type DOD	Possesseur	Acquéreur
Alarmes [P]	Lumière d'alerte	Couche interface	FWS	PM
	Son d'alerte	Couche interface	FWS	PM

	Hauts parleurs & casques	Périphérique d'entrée/sortie	FWS	PM
	Master W/C	Périphérique d'entrée/sortie	FWS	PM
	Alarmes (WD)	Couche interface	FWS	PM
Alarmes / Procédures [P]	Recall ALL pb	Périphérique d'entrée	FWS	PM
	Recall LAST pb	Périphérique d'entrée	FWS	PM
	Clear pb	Périphérique d'entrée	FWS	PM
	EMER CANCEL pb	Périphérique d'entrée	FWS	PM
Procédures [P]	Première procédure	Couche interface	FWS	PM
Actions de récupération [P]	Ligne de procédure	Couche interface	FWS	PM
	Bouton de validation	Périphérique d'entrée	FWS	PM
Stratégie de filtrage	Stratégie utilisée pour le filtrage	SW Object	FWS	
Stratégie de vérification pertinence procédure	Stratégie utilisée pour vérifier la pertinence de la procédure	Information	PM	
Stratégie de vérification des alarmes	Les capteurs utilisés pour l'overhead panel sont différents des capteurs qui déclenchent l'alarme	Connaissance	PM	
	Les capteurs utilisés pour le SD sont différents des capteurs qui déclenchent l'alarme	Connaissance	PM	
	Connaissance de pilotage	Connaissance	PM	
Etats Systèmes [P]	Page système (SD)	Couche interface	FWS	PM
	Boutons systèmes	Périphériques d'entrée	FWS	PM

2.5.2.2 Phase 2 du processus RCRAFT : Allocation des transitions de contrôle et de l'initiative

Le Tableau 39 présente les colonnes du tableau RCRA décrivant l'allocation des transitions de contrôle et de l'initiative entre le FWS et le pilote *monitoring* produit à la fin des itérations nécessaires de la sous-phase 3 de la phase 2 du processus RCRAFT (voir Chapitre 6). Le Tableau 39 indique que lorsque le Master W/C est allumé par le FWS (par la fonction « Lancer les *attention getters* »), le FWS initie une transition de contrôle. Cette transition de contrôle déclenche le début d'une nouvelle tâche du pilote *monitoring*, il doit acquitter les alarmes. Pour

permettre l'acquittement des alarmes, le FWS rend le contrôle au pilote *monitoring* sur le bouton Master W/C (FWS: *handover*, PM : *takeover*). Lorsque le FWS affiche la procédure : « Afficher la procédure de la première alarme », il n'y a pas de transition de contrôle. Le FWS garde le contrôle sur la procédure. Lorsque le pilote monitoring utilise le bouton EMER CANCEL : « Utiliser EMER CANCEL », il reprend le contrôle (*takeover*) par le biais de ce bouton pour demander la suppression de la procédure et l'alarme la plus prioritaire (*handover*). Le FWS efface la procédure si cette procédure est supprimable : « Afficher résultat EMER CANCEL » et « Voir la prochaine procédure » ou « Voir la procédure non supprimée ». Le FWS rend le contrôle au pilote *monitoring* sur le bouton EMER CANCEL. Les lignes correspondantes du Tableau 39 sont sur fond bleu.

Tableau 39. Colonnes du tableau décrivant les transitions de contrôle et l'allocation de l'initiative entre le FWS et le PM pour la résolution des alarmes.

Transitions de contrôle et allocation de l'initiative				
Transi-sition	Initia-tive	*Handover* ou émetteur	*Takeover* ou receveur	Tâches et Fonctions
X Master W/C	FWS	FWS : Lancer les "attention getters" **HANDOVER**	PM : Percevoir les attention getters **TAKEOVER**	FWS : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
X Master W/C	FWS	FWS : Lancer les "attention getters" **TAKEOVER HANDOVER**	PM : Percevoir les attention getters **TAKEOVER**	Si le PM était en train de réaliser une procédure, il est interrompu par l'arrivée d'une alarme plus prioritaire
	FWS	FWS : Afficher les alarmes rouges	PM : Voir les alarmes actives	FWS : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
	FWS	FWS : Afficher les alarmes ambres	PM : Voir les alarmes actives	FWS : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
	FWS	FWS : Afficher la procédure de la première alarme	PM : Voir la première procédure	FWS : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
	FWS	FWS : Afficher la page système	PM : Voir la page système	FWS : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
X Master W/C	PM	PM : Presser le Master W/C **HANDOVER**	FWS : Recevoir pression Master W/C	
	FWS	FWS : Stopper le son	PM : Entendre l'alerte sonore s'arrêter	
	FWS	FWS : Eteindre le Master W/C	PM : Voir le Master W/C éteint	

X titres des alarmes	PM	PM : Faire défiler **HANDOVER**	FWS : Recevoir défilement	
	FWS	FWS : Faire défiler les alarmes **HANDOVER**	PM : Voir le résultat	
	FWS	FWS : allumer bouton MORE page **HANDOVER**	PM : Voir MORE sur la page SD et bouton MORE allumé	
	PM	PM : Presser le bouton MORE **HANDOVER**	FWS : Recevoir demande page MORE	
	FWS	FWS : Afficher page MORE sur SD **HANDOVER**	PM : Voir la page MORE	
x boutons systè-mes	FWS	FWS : Allumer bouton système **HANDOVER**	PM : Voir les boutons systèmes sur l'ECP allumés	
	PM	PM : Presser le bouton correspondant au système **HANDOVER**	FWS : Recevoir pression syst button	
	FWS	FWS : Afficher page SD **HANDOVER**	PM : Voir les informations sur le système	
x EMER CANC EL pb + Pre-mière procé-dure	PM	PM : Presser le bouton EMER CANCEL **TAKEOVER**	FWS : Recevoir EMER CANCEL	"Accéder à la procédure pertinente" du PM, "Soutenir la sélection d'une procédure" du FWS
	FWS	FWS : Afficher résultat EMER CANCEL	PM : Voir la procédure	
	FWS	Afficher la procédure non supprimée	PM : Voir la procédure	
X RECA LL ALL	FWS	FWS : allumer RECALL ALL pb	PM : Voir le bouton RECALL ALL allumé	
	PM	PM : Presser le bouton RECALL ALL	FWS : Recevoir RECALL ALL	
		FWS : Afficher les procédures rappelées	PM : Voir les titres	
X RECA LL LAST x	FWS	FWS : Allumer RECALL LAST pb	PM : Voir le bouton RECALL LAST allumé	
	PM	PM : Presser le bouton RECALL LAST	FWS : Recevoir RECALL LAST	
	FWS	FWS : Afficher la procédure rappelée	PM : Voir la procédure rappelée	
X ECP & Ligne de procéd ure	PM	PM : Cliquer sur la checkbox **HANDOVER**	FWS : Recevoir que le PM valide la ligne de procédure	

	FWS	FWS : Cocher la ligne de procédure	PM : Voir la case à cocher [de la ligne de procédure] cochée	
	FWS	FWS : Cocher la ligne de procédure	PM : Voir la ligne automatiquement cochée	
x	FWS	FWS : Allumer bouton CLEAR et afficher la barre d'effacement **HANDOVER**	PM : Voir la barre d'effacement et le bouton CLEAR allumé	
x	PM	PM : Presser le bouton CLEAR **HANDOVER**	FWS : Recevoir CLEAR	
	FWS	FWS : Afficher la prochaine procédure	PM : Voir la procédure	
	FWS	FWS : Collaboration check ligne de procédure **TAKEOVER/ HANDOVER**	PM : Collaboration ligne de procédure **HANDOVER/ TAKEOVER**	

2.5.2.3 Phase 2 du processus RCRAFT : Allocation de la responsabilité

Le Tableau 40 présente les colonnes du tableau RCRA décrivant l'allocation de la responsabilité entre le pilote *monitoring* et le FWS produit à la fin des itérations nécessaires de la sous-phase 4 de la phase 2 (voir Chapitre 6). Le FWS soutient le but du pilote en guidant le pilote *monitoring* dans la résolution des alarmes. Le but du pilote *monitoring* est de ramener l'avion dans un état fonctionnel. Le FWS et le pilote *monitoring* partagent le résultat attendu de résoudre toutes les alarmes lors de la gestion des alarmes. Les tâches et fonctions modifiant des DOD pouvant faire dévier ce résultat attendu sont à responsabilité. Par exemple, les objets logiciels concernant les procédures peuvent faire dévier la résolution de toutes les alarmes. Le FWS a la responsabilité de filtrer les procédures différées, de récupérer la procédure associée à la première alarme et d'afficher la procédure de la première alarme. Le pilote *monitoring* a la responsabilité de déterminer si l'alarme est intempestive (« Décider que la première procédure est pertinente » ou « Décider que la première procédure n'est pas pertinente pour la situation ») et de réaliser les actions de récupération. Les lignes correspondantes sont sur fond bleu dans le Tableau 40.

Tableau 40. Colonnes du tableau RCRA décrivant l'allocation de la responsabilité entre le FWS et le PM pour la résolution des alarmes.

Allocation de la responsabilité	
Tâche/ Fonction	Résultat attendu possiblement impacté
FWS : Traiter une (ou plusieurs) alarme reçue d'un système avion	Résoudre toutes les alarmes
FWS : Inhiber des alarmes	
FWS : Combiner des alarmes	
FWS : Lever une/des alarmes	
FWS : Filtrer les procédures différées	
FWS : Ordonnancer les alarmes par priorité	

FWS : Récupérer procédure associée à la première alarme	
FWS : récupérer les infos systèmes	
FWS : Afficher les alarmes rouges	
FWS : Afficher les alarmes ambres	
FWS : Lancer les "attention getters"	
FWS : Afficher la procédure de la première alarme	
FWS : Afficher la page système	
PM : Se rappeler des réactions immédiates	
PM : Décider que l'observation est cohérente avec l'alarme	
PM : Décider que la première procédure est pertinente	
PM : Décider que la première procédure n'est pas pertinente pour la situation	
PM : Déterminer la procédure pertinente	
PM : Presser le bouton EMER CANCEL	
FWS : Emer cancel alarme la plus prioritaire	
FWS : Afficher résultat EMER CANCEL	
FWS : Afficher résultat EMER CANCEL	
FWS : Afficher la procédure non supprimée	
PM : Presser le bouton RECALL LAST	
FWS : RECAL LAST (rappeler dernière) procédure	
FWS : Afficher la procédure rappelée	
PM : Presser le bouton RECALL ALL	
FWS : RECALL ALL Procedures	
FWS : Afficher les procédures rappelées	
PM : Réaliser l'action de récupération	
PM : Cliquer sur la checkbox	
PM : Presser le bouton CLEAR	
FWS : Détecter l'action de récupération	
FWS : Cocher la ligne de procédure	
FWS : Détecter si la procédure est terminée	
FWS : Effacer la procédure	
FWS : Afficher la prochaine procédure	
PM : Décider qu'il reste des procédures	
PM : Décider qu'il n'y a plus de procédure	

2.5.2.4 Phase 2 du processus RCRAFT : Allocation de l'autorité

Le Tableau 41 présente les colonnes du tableau RCRA décrivant l'allocation de l'autorité entre le FWS et le pilote *monitoring* produit à la fin des itérations nécessaires de la sous-phase 5 de la phase 2 (voir Chapitre 6). Par exemple, le Tableau 41 indique que le FWS a l'autorité de définir l'ordre de priorité des alarmes et l'ordre de réalisation des procédures des alarmes. Également, le FWS a l'autorité de définir que la procédure associée à l'alarme (rouge) doit être réalisée dans les plus brefs délais et que la procédure associée à l'alarme (ambre) doit être réalisée. Le pilote *monitoring* a l'autorité de définir si l'alarme levée par le FWS est intempestive ou pertinente (« Décider que la première procédure est pertinente » ou « Décider que la première procédure n'est pas pertinente pour la situation »). Le pilote *monitoring* a l'autorité de supprimer une alarme intempestive avec EMER CANCEL. Le FWS a l'autorité de

refuser la suppression de la procédure associée à l'alarme si cette alarme n'est pas supprimable. Les lignes correspondantes du Tableau 41 sont sur fond bleu.

Tableau 41. Colonnes du tableau décrivant l'allocation de l'autorité entre le FWS et le PM pour la résolution des alarmes.

Allocation de l'autorité	
Tâche/Fonction	**Contrainte**
FWS : Combiner des alarmes	Défini la valeur de l'alarme
FWS : Inhiber les alarmes	Masque des alarmes au PM
FWS : Filtrer les procédures différées	
FWS : Ordonnancer les alarmes par priorité	Défini l'ordre de priorité des alarmes (rouge, ambre)
	Défini l'ordre de réalisation des procédures des alarmes
FWS : Récupérer procédure associée à la première alarme	Défini les actions à réaliser pour résoudre l'alarme
FWS : Récupérer les infos systèmes	Défini les informations systèmes pertinentes à présenter au PM
FWS : Ecouter EMER CANCEL	Le PM peut appuyer sur le bouton pour EMER CANCEL
FWS : Ecouter Master W/C	Le PM peut appuyer sur le bouton pour couper les attention getters
FWS : Ecouter bouton de validation	Le PM peut appuyer sur le bouton pour valider
PM : Presser le Master W/C	Demande l'arrêt des attentions getters
PM : Faire défiler	Demande de faire défiler les alarmes
FWS : Détecter page MORE	Active le bouton MORE pour afficher des informations systèmes supplémentaires
FWS : Ecouter boutons systèmes	Active les bouton systèmes. Le PM doit presser ces boutons pour afficher les informations du système correspondant.
PM : Presser le bouton MORE	Demande l'affichage des informations supplémentaires
PM : Presser le bouton correspondant au système	Demande l'affichage des informations du système
PM : Déterminer les conséquences	Défini les conséquences possibles de la défaillance sur l'état de l'avion et sur la mission
PM : Décider que l'observation est cohérente avec l'alarme	Défini que l'alarme est cohérente avec les informations des pages SD et de l'overhead panel
PM : Décider que la première procédure est pertinente	Défini que l'alarme (et la procédure associée) n'est pas intempestive et appropriée à la situation
PM : Décider que la première procédure n'est pas pertinente pour la situation	Défini que l'alarme (et la procédure associée) est intempestive ou non appropriée à la situation en considérant des éléments contextuels inconnus du FWS
PM : Déterminer la procédure pertinente	Défini la procédure à réaliser
PM : Décider de la méthode	Défini quelles actions réaliser pour afficher la procédure voulue

PM : Presser le bouton EMER CANCEL	Demande la suppression de l'alarme la plus prioritaire
FWS : Afficher la procédure non supprimée	Refus de la suppression (la procédure/l'alarme n'est pas supprimable)
FWS : Ecouter RECALL LAST	Le PM peut appuyer sur le bouton pour RECALL LAST
PM : Presser le bouton RECALL LAST	Demande le rappel de la dernière procédure effacée
FWS : Ecouter RECALL ALL	Le PM peut appuyer sur le bouton pour RECALL ALL
PM : Presser le bouton RECALL ALL	Demande le rappel de toutes les procédures effacées
PM : Réaliser l'action de récupération	Réalise l'action de récupération
PM : Cliquer sur la checkbox	Défini l'action comme étant terminée
FWS : Détecter l'action de récupération	Défini l'action comme étant terminée
FWS : Détecter si la procédure est terminée	Défini la procédure comme étant terminée
FWS : Ecouter bouton CLEAR	le PM doit appuyer sur CLEAR pour effacer la procédure
PM : Presser le bouton CLEAR	Demande l'effacement de la procédure

2.5.2.5 Phase 3 du processus RCRAFT : Modélisation RCRAFT

L'analyse RCRAFT de la gestion des alarmes par le pilote *monitoring* et le FWS a été réalisée en phase 2. Les modèles de tâches ségrégués et de fonctions du pilote *monitoring* et du FWS intégrant la modélisation des éléments RCRAFT sont consultables en Annexe 1.1 et 1.2.

3 Prototype de système alternatif de gestion des alarmes

Dans le cadre du lot de travail MAGIC, nous avons proposé une alternative au système actuel de gestion des alarmes. Le prototype de système alternatif de gestion des alarmes a pour but de prendre en compte les éléments contextuels inconnus du système. Pour cela, nous proposons de laisser au pilote le choix de l'ordre de réalisation des actions de récupération pour la résolution des alarmes. En effet, le pilote a la possibilité d'accéder à une procédure particulière en supprimant les alarmes plus prioritaires avec EMER CANCEL mais son utilisation doit rester exceptionnelle et le système peut refuser la suppression si l'alarme n'est pas supprimable. Le prototype de système alternatif de gestion des alarmes recommande des actions de récupération au pilote de manière similaire aux systèmes de recommandation couramment utilisés pour le commerce en ligne par exemple (Hauser et al., 2009).

3.1 Interface pour donner le choix au pilote

La Figure 115 présente un écran du prototype de système alternatif de gestion des alarmes. Les états des systèmes sont présentés sur la gauche, les alarmes actives au centre et les recommandations d'actions de récupération sur la droite.

3.1.1 Recommandations d'actions de récupération plutôt que des procédures

Nous avons proposé de recommander des actions de récupération plutôt que des procédures. Le pilote *monitoring* pourra choisir quelles actions de récupération réaliser en premier selon les alarmes, les états des systèmes affichés et le contexte actuel que le système avion ne peut pas détecter (comme le malaise d'un passager par exemple). Afin de guider le pilote dans la résolution des alarmes, un filtrage est effectué sur les actions de récupération. Le filtrage des actions de récupération utilise des contraintes similaires au filtrage effectué par le FWS sur les alarmes et les procédures associées comme la phase de vol, l'altitude, la vitesse, l'état des systèmes, etc. Également, nous proposons de spécifier les actions de récupérations par :

- Le temps restant pour les réaliser,
- Leur criticité,
- Les alarmes associées,
- Le temps nécessaire pour les réaliser,
- Les systèmes impactés,
- Les buts utilisateurs associés

afin d'ordonnancer les actions de récupération, de les regrouper selon les buts utilisateurs que les actions de récupération permettent d'atteindre et les classer selon deux niveaux de criticité : immédiates ou non immédiates.

3.1.1.1 Recommandations d'actions de récupération immédiates

Les recommandations d'actions de récupération immédiates sont des actions de récupération à réaliser immédiatement. Ces recommandations sont présentées sur une timeline sur la Figure 115 (à droite dans la section haute des recommandations). La timeline indique le temps restant estimé pour réaliser l'action de récupération. Les durées indiquées sur Figure 115 sont seulement des exemples à titre illustratif. Sur l'exemple présenté sur la Figure 115, les alarmes CAB PRESS EXCESS CAB ALT et ENG ALL ENGs FLAME OUT sont actives. Le système recommande aux pilotes d'utiliser les masques à oxygène (CREW OXY MASK … USE) avant de faire descendre l'altitude de l'avion (DIVERSION … INITIATE) et de tenter d'allumer des sources d'énergie alternatives aux moteurs (APU … START ; RAT MAN … ON).

3.1.1.2 Recommandations d'actions de récupération non immédiates

Les recommandations d'actions de récupération non immédiates sont des actions de récupération à réaliser pour lesquelles le temps de réalisation n'a pas d'influence ou bien encore assez lointain pour ne pas être considérées comme immédiates. Par exemple, sur la Figure 115, l'action de récupération pour activer une nouvelle source d'air (RAM AIR … ON) et moins urgente que les actions de récupération indiquant aux pilotes d'utiliser leurs masques à oxygène, d'initier une descente et d'activer de nouvelles sources d'énergie.

De la même manière que les lignes de procédures, le pilote *monitoring* peut utiliser la molette de l'ECP pour sélectionner une recommandation et le bouton de validation pour indiquer qu'une action de récupération qui n'est pas détectée par le système a été réalisée. Lorsque les recommandations ont été réalisées, le pilote monitoring les efface avec le bouton CLEAR. Le pilote monitoring peut également rappeler les actions de recommandation si l'alarme, que ces

actions de récupération permettent de résoudre, est toujours active (RECALL ALL, RECAL LAST). Il peut également supprimer les alarmes intempestives (EMER CANCEL).

Le pilote monitoring réalise les actions de récupération dans l'ordre qui lui semble le plus pertinent selon la situation actuelle.

3.1.2 Etats systèmes

Dans le cadre du lot de travail MAGIC, nous avons proposé une nouvelle description générique des états des systèmes de l'avion (OQCR : états Opérationnels, Qualitatifs, Contexte, Restriction) ainsi qu'une nouvelle décomposition hiérarchique de l'architecture des systèmes de l'avion (DSCU : « Device », Service système, service Composé, « User » service) (Elodie Bouzekri, Canny, Martinie, et al., 2019a) et (Elodie Bouzekri, Canny, Martinie De Almeida, et al., 2019). Le prototype de nouveau système de gestion des alarmes intègre la représentation des états OQCR et de l'architecture DSCU des systèmes. Ces états ont été définis après un travail d'abstraction des informations présentées sur les pages systèmes (SD) et décrites dans la documentation.

Le prototype de système alternatif de gestion des alarmes affiche les états système sous forme d'états discret composés de deux valeurs représentant l'état du système ou du service (OQCR) : l'état opérationnel (O : allumé, en train de s'éteindre, ...), l'état qualitatif (Q : fonctionnel, dégradé ou hors service), et deux autres valeurs binaires décrivant si le système ou le service évolue dans son contexte d'utilisation (C) et sa restriction d'usage (R) c.-à-d. si son utilisation est autorisée ou non. Lorsque le pilote *monitoring* sélectionne une action de récupération, les états systèmes et alarmes à l'origine de la recommandation sont indiqués. Le pilote *monitoring* a aussi la possibilité de modifier l'état d'un système pour que le système lui indique quelles actions de récupération sont nécessaires pour que le système soit dans l'état choisi.

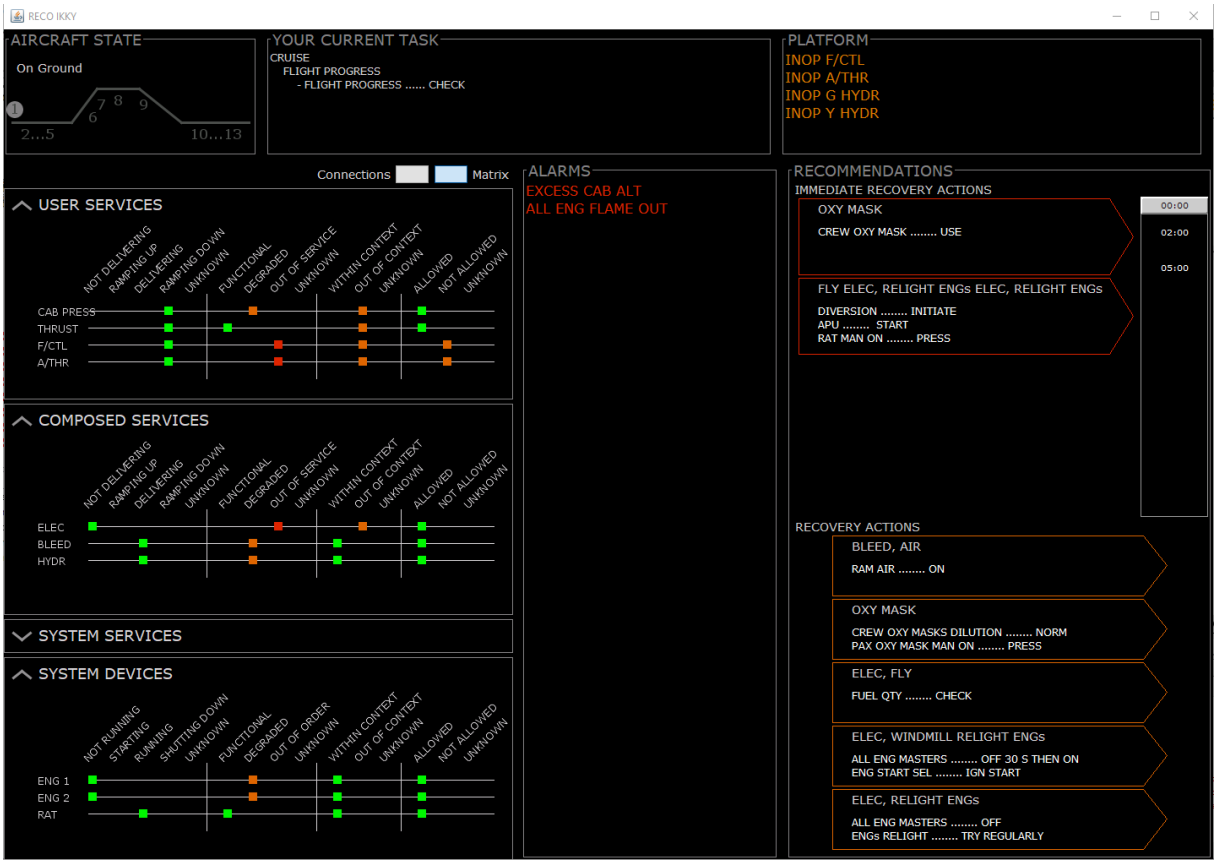


Figure 115. Prototype de système alternatif de gestion des alarmes.

3.2 Analyse RCRAFT de la gestion des défaillances du pilote « monitoring » avec le système alternatif de gestion des alarmes

Dans cette section, nous réalisons l'analyse RCRAFT sur la gestion des alarmes dont la concept map est présentée en Figure 116. Nous étudierons l'allocation des tâches et fonctions, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité entre un acteur humain ayant le rôle de pilote *monitoring* (PM) et un acteur système alternatif de gestion des alarmes ayant le rôle de de recommandeur. Les rôles de chacun des acteurs est défini pour le vol.

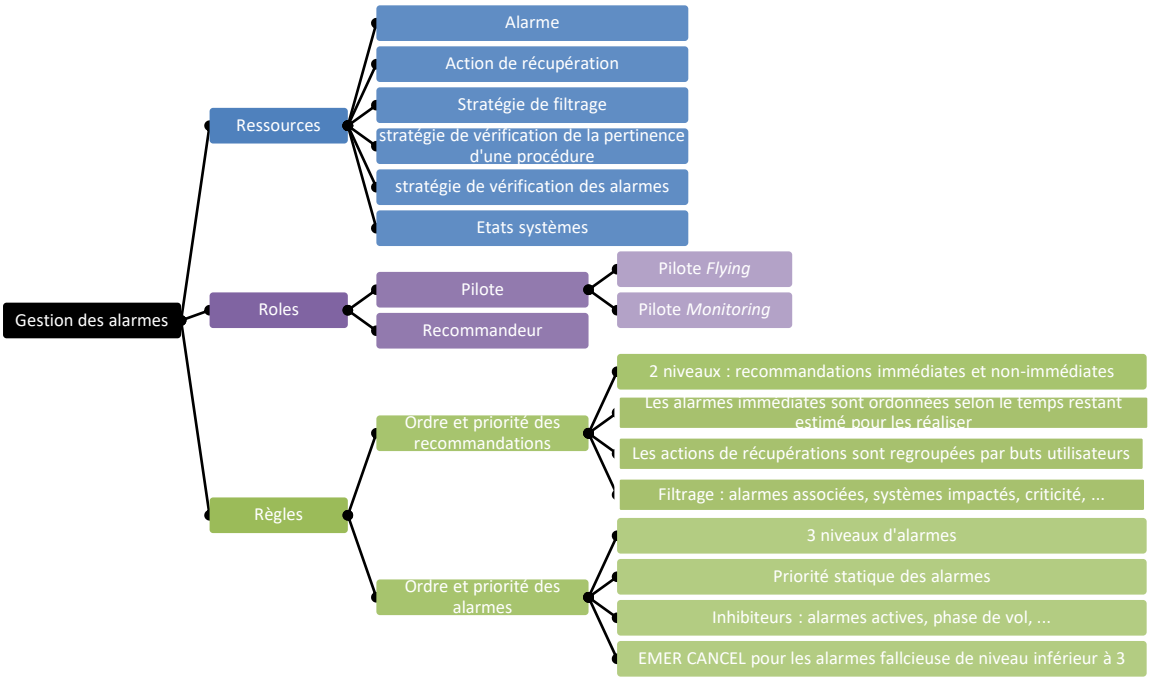


Figure 116. Concept map de la gestion des alarmes avec le système alternatif de gestion des alarmes.

3.2.1 Modélisation des tâches du pilote monitoring et des fonctions du recommandeur

Cette section présente les artefacts produits lors de la phase 1 du processus RCRAFT rappelé dans la Figure 117 :

- Les modèles de tâches ségrégués du pilote *monitoring*,
- Les modèles de fonctions du recommandeur,

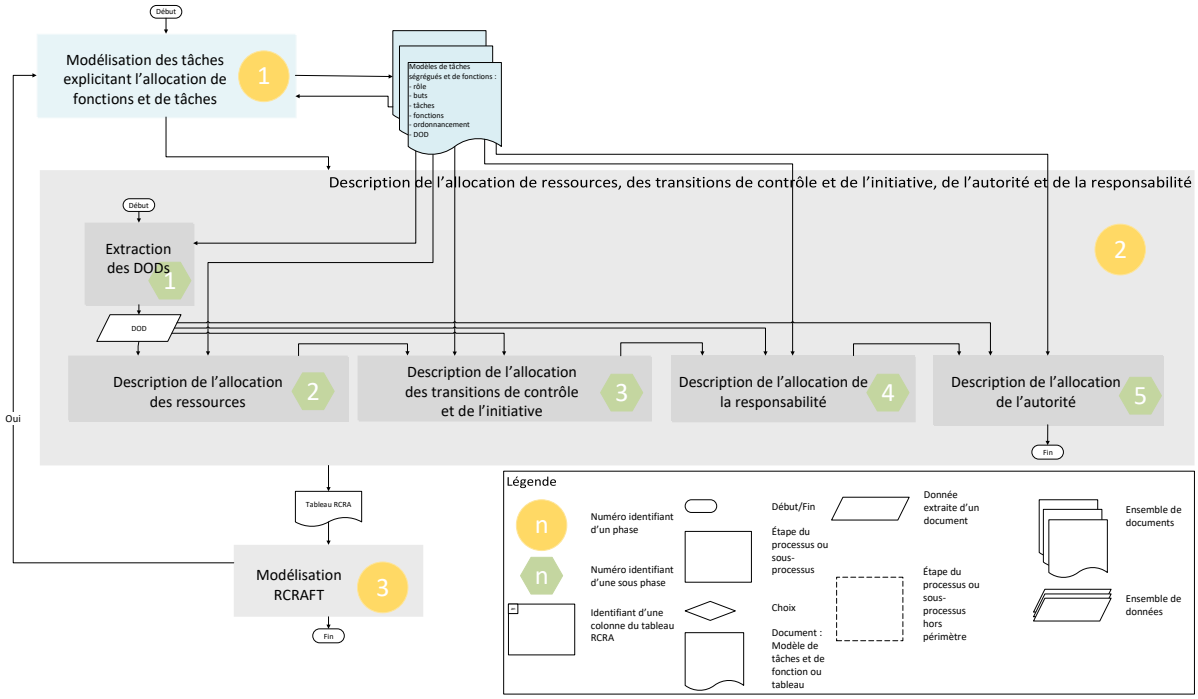


Figure 117. Vue haut-niveau du processus RCRAFT avec les artefacts modèles de tâches ségrégués et de fonctions et la phase 1 mis en valeur.

3.2.1.1 Phase 1 du processus RCRAFT : Modélisation des fonctions du recommandeur

La Figure 118 présente le modèle de fonctions haut-niveau du recommandeur pour gérer les alarmes produit lors de la phase 1 du processus RCRAFT (voir Chapitre 6). Le détail des fonctions du recommandeur est présenté en Annexes 3.2. La gestion des alarmes débute par la réception d'un état système dégradé que le recommandeur va traduire en alarme.

- Fonction abstraite « Recevoir une/des nouvelles alarmes » : De la même manière que le FWS, le recommandeur inhibe et combine les alarmes. Ensuite, le recommandeur génère les recommandations d'actions de récupération puis les filtres en respectant des contraintes.
- Fonction abstraite « Afficher les actions de récupération recommandées » : Le recommandeur trie les actions de récupération en deux catégories. Ces deux catégories sont : les actions de récupération immédiates pour lesquelles il estime le temps restant pour les réaliser et les actions de récupération non immédiates. Il groupe également les actions de récupération par les buts utilisateurs que ces actions permettent d'atteindre. Ensuite, il affiche les recommandations, les états OQCR, les alarmes actives et lève les *attentions getters*.
- Fonction abstraite « Soutenir l'acquittement des alarmes » : Le recommandeur stoppe le son d'alerte et éteint la lumière d'alerte lorsque le pilote *monitoring* presse le Master W/C. Le recommandeur fait défiler les recommandations lorsque le pilote *monitoring* utilise la molette. Le recommandeur met en valeur les données affichées à l'origine de la recommandation sélectionnée. Lorsque le pilote monitoring sélectionne l'état voulu d'un système, le recommandeur met en valeur les recommandations qui permettront au pilote monitoring de modifier l'état du système comme il le souhaite.

- Fonction abstraite optionnelle « Soutenir la sélection d'une recommandation et rappeler les actions de récupération effacées » : De la même manière que le FWS, le recommandeur peut rappeler les actions de récupération effacées.
- Fonction abstraite « Soutenir la réalisation des actions de récupération » : De la même manière que le FWS, le système peut détecter lorsque certaines actions de récupération sont terminées ou bien soutenir la validation des actions de récupération non détectée. Comme le FWS, le recommandeur détermine quand une recommandation est terminée et effaçable.

Le recommandeur peut gérer l'arrivée d'une nouvelle alarme (fonction abstraite : « recevoir une/des nouvelles alarmes »), afficher les nouvelles recommandations (fonction abstraite : « afficher les actions de récupération recommandées ») et soutenir l'acquittement de ces nouvelles alarmes (fonction abstraite : « soutenir l'acquittement des alarmes ») sans interrompre (opérateur d'entrelacement) ses fonctions permettant de soutenir la réalisation des recommandations.

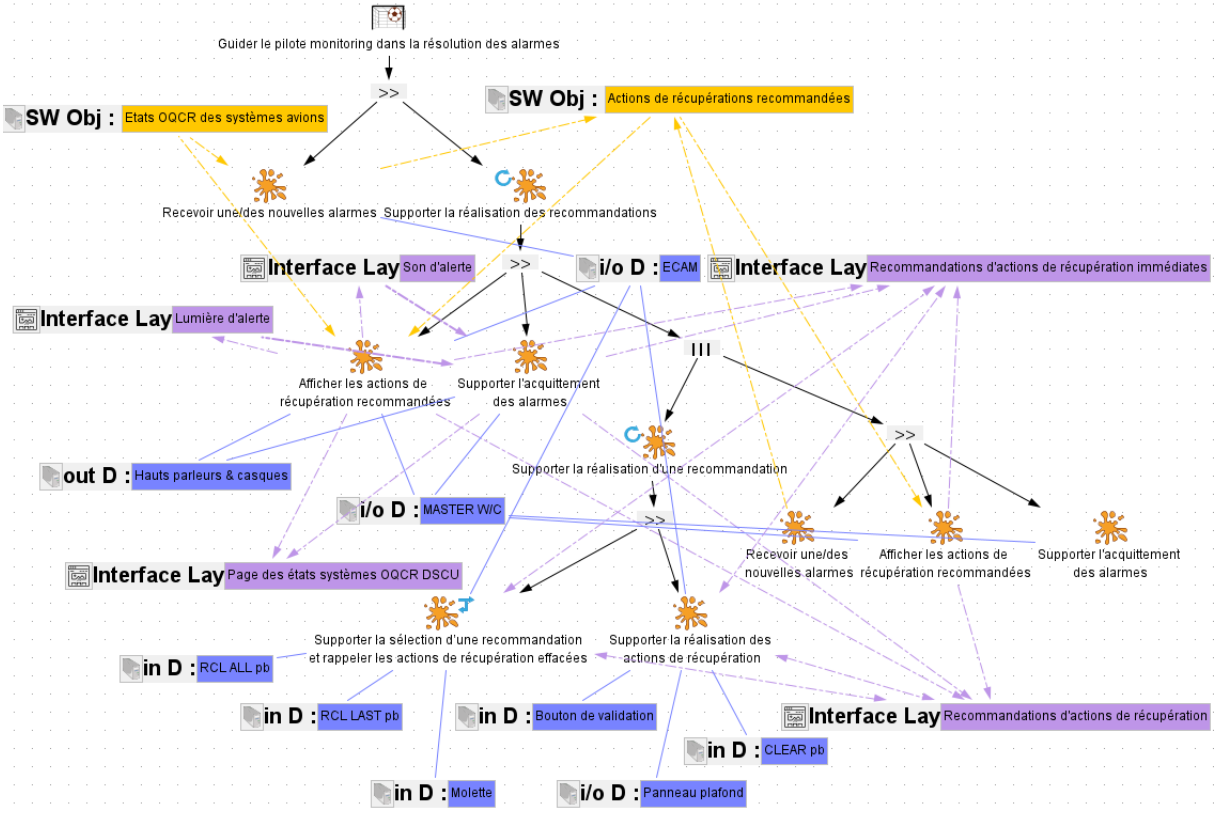


Figure 118. Modèle de fonctions du recommandeur pour gérer les alarmes.

3.2.1.2 Phase 1 du processus RCRAFT : Modélisation des tâches du pilote monitoring

La Figure 119 présente le modèle de tâches ségrégué haut-niveau du pilote *monitoring* pour résoudre l'alarme produite lors de la phase 1 du processus RCRAFT (voir Chapitre 6). Le détail des tâches abstraites de la Figure 119 sont présentés en Annexe 3.1. Les tâches réalisées par le pilote *monitoring* sont :

- Tâche abstraite « Acquitter les alarmes » : Après que le recommandeur ait levée les alarmes, les attention getters et afficher les recommandations, les pilotes acquittent l'alarme en pressant le MASTER W/C et vérifient que l'alarme n'est pas intempestive. Pour cela, ils comparent les états OQCR des systèmes affichés avec les états des systèmes indiqués sur le panneau plafond. Il peut sélectionner un état pour un système pour consulter les actions de récupération recommandées pour changer l'état du système en l'état sélectionné. Il peut sélectionner une recommandation pour voir une mise en valeur des états systèmes et alarmes à l'origine de la recommandation.
- Tâche abstraite « Choisir une recommandation » : Le pilote *monitoring* peut faire défiler les recommandations affichées grâce à la molette de l'ECP. Comme avec les FWS, le pilote *monitoring* peut rappeler la dernière ou les recommandations effacées.
- Tâche abstraite « réaliser la recommandation » : Ensuite, le pilote *monitoring* réalise la procédure : le pilote *monitoring* annonce, réalise les actions de récupération, coche les lignes de procédure si besoin et efface la procédure lorsqu'elle est terminée (tâche abstraite « Réaliser la première procédure » de la Figure 113).
- Tâche cognitive de décision « Choisir de réaliser une autre recommandation » : Le pilote *monitoring* peut interrompre la réalisation d'une recommandation en décidant d'en choisir une.

Le tâche de réalisation des recommandations peut être suspendue par l'arrivée d'une nouvelle alarme que le pilote *monitoring* devra acquitter : suspension de la tâche abstraite itérative « Réaliser les recommandations » par la tâche abstraite « Acquitter les alarmes » sur la Figure 119.

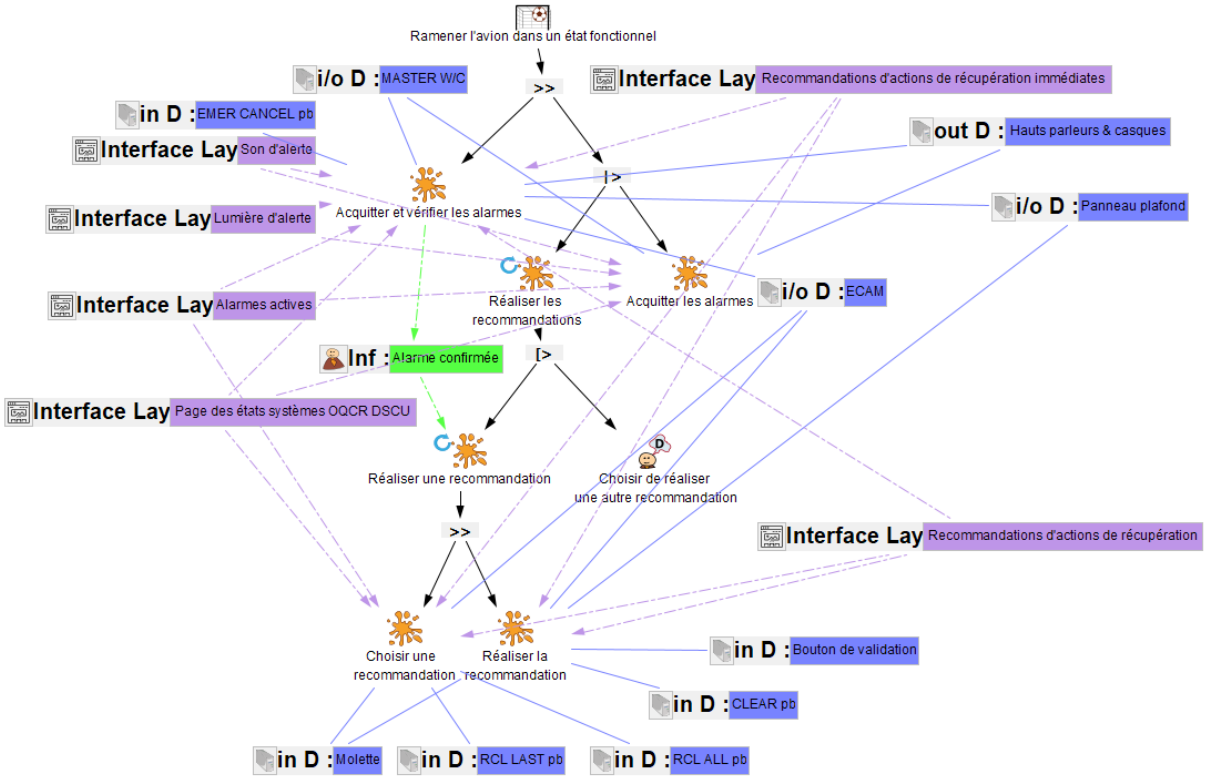


Figure 119. Modèle de tâches du PM pour résoudre les alarmes avec le prototype de recommandeur.

3.2.2 Allocation des ressources, des transitions de contrôle et de l'initiative, de l'autorité et de la responsabilité entre le pilote monitoring et le recommandeur

Cette section présente l'artefact produit lors de la phase 2 du processus RCRAFT rappelé dans la Figure 120 : le tableau RCRA. Ce tableau décrit l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre le pilote *monitoring* et le recommandeur.

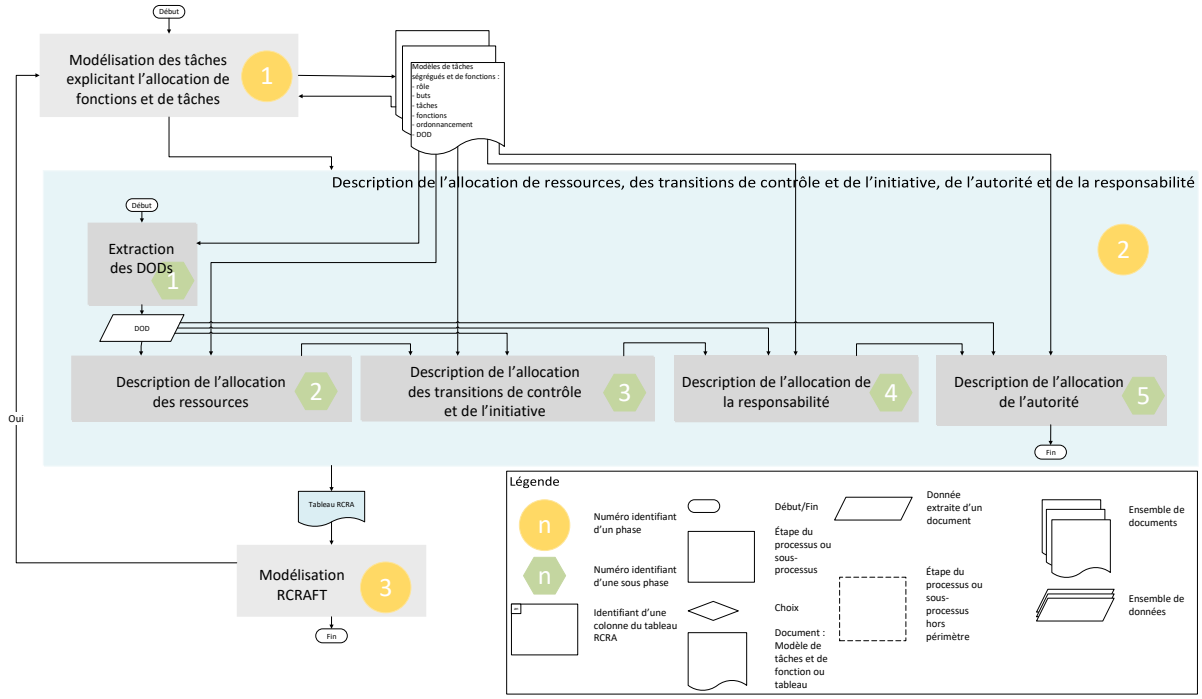


Figure 120. Vue haut-niveau du processus RCRAFT avec l'artefact : tableau RCRA et la phase 2 mis en valeur.

3.2.2.1 Phase 2 du processus RCRAFT : Allocation des ressources

Les ressources utiles à la gestion des alarmes sont : les alarmes, la stratégie de filtrage, la stratégie de vérification des alarmes, la stratégie de vérification pertinence procédure, les actions de récupération et les systèmes. Le Tableau 42 présente un extrait du tableau RCRA décrivant l'allocation des ressources entre le recommandeur et le pilote *monitoring* produit à la fin des itérations nécessaires (pour traiter tous les DODs des modèles) de la sous-phase 2 de la phase 2 du processus RCRAFT (voir Chapitre 6). Le tableau décrivant l'allocation complète des ressources entre le pilote *monitoring* et le recommandeur est consultable en annexe (Tableau 47). Les ressources : alarmes, la stratégie de filtrage, actions de récupération et systèmes sont partagées entre les deux rôles. Par exemple, de la même manière qu'avec le FWS, les alarmes sont partagées par le recommandeur au pilote par une lumière d'alerte sur le MASTER W/C, les sons d'alerte à travers les hauts parleurs & casques et les titres des alarmes affichés sur le WD. La stratégie de vérification des alarmes n'est pas partagée par le pilote *monitoring* avec le recommandeur. La stratégie de vérification de la pertinence de la procédure n'est pas partagée par le pilote *monitoring* avec le recommandeur. La stratégie de filtrage est partagée par le recommandeur avec le pilote *monitoring* mais seulement en partie. Comme le montre le Tableau 42, la ressource « stratégie de filtrage » est partagé par la Couche interface « Alarmes » et la Couche interface « page des états systèmes OQCR DSCU ». Le recommandeur montre au pilote quels états système sont à l'origine de quelle alarme. Seules les contraintes liées à l'état des systèmes et aux alarmes actives sont partagées au pilote *monitoring*.

Tableau 42. Extrait des colonnes du tableau RCRA décrivant l'allocation des ressources entre le recommandeur (Recommandeur) et le pilote *monitoring* (PM) pour la résolution des alarmes.

Allocation des ressources

Ressource [Partagée]	Description DOD	Type DOD	Possesseur	Acquéreur
Alarmes [P]	Lumière d'alerte	Couche interface	Recommandeur	PM
	Son d'alerte	Couche interface	Recommandeur	PM
	Hauts parleurs & casques	Périphérique d'entrée/sortie	Recommandeur	PM
	Master W/C	Périphérique d'entrée/sortie	Recommandeur	PM
Alarmes [P] / Stratégie de filtrage [P]	Alarmes	Couche interface	Recommandeur	PM
Alarmes [P] / Actions de récupération [P]	Recall ALL pb	Périphérique d'entrée	Recommandeur	PM
	Recall LAST pb	Périphérique d'entrée	Recommandeur	PM
	Clear pb	Périphérique d'entrée	Recommandeur	PM
	EMER CANCEL pb	Périphérique d'entrée	Recommandeur	PM
Actions de récupération [P]	Timeline des recommandations d'actions de récupération immédiates	Couche interface	Recommandeur	PM
	Recommandations d'actions de récupération	Couche interface	Recommandeur	PM
	Bouton de validation	Périphérique d'entrée	Recommandeur	PM
Stratégie de filtrage [P]	Stratégie utilisée pour le filtrage	SW Object	Recommandeur	
Stratégie de vérification pertinence procédure	Stratégie utilisée pour vérifier la pertinence de la procédure	Information	PM	
Systèmes [P] / Stratégie de filtrage [P]	Page des états OQCR DSCU	Couche interface	Recommandeur	PM
	Page STATUS	Couche interface	Recommandeur	PM

3.2.2.2 Phase 2 du processus RCRAFT : Allocation des transitions de contrôle et de l'initiative

Le Tableau 43 présente les colonnes du tableau RCRA décrivant l'allocation des transitions de contrôle et de l'initiative entre le recommandeur et le pilote *monitoring* (PM) produit à la fin des itérations nécessaires de la sous-phase 3 de la phase 2 du processus RCRAFT (voir Chapitre 6). Le Tableau 43 indique que lorsque le Master W/C est allumé par le recommandeur (par la fonction « Lancer les *attention getters* »), le recommandeur initie une transition de contrôle. Le pilote *monitoring* doit acquitter les alarmes après cette transition de contrôle. Pour permettre l'acquittement des alarmes, le recommandeur rend le contrôle au pilote monitoring sur le bouton W/C (recommandeur : *handover*). Lorsque le recommandeur affiche les recommandations : « Afficher timeline recommandations immédiates » et « Afficher les recommandations », il initie une transition de contrôle. Le recommandeur reprend le contrôle

(*takeover*) en réalisant une fonction un affichage non sollicité par le pilote *monitoring*, puis rend le contrôle (*handover*) sur les recommandations affichées au pilote *monitoring*. Le pilote *monitoring* peut manipuler les recommandations. Les lignes correspondantes du Tableau 43 sont sur fond bleu.

Tableau 43. Colonnes du tableau RCRA décrivant l'allocation des transitions de contrôle et de l'initiative entre le recommandeur (Recommandeur) et le pilote monitoring (PM)

Transitions de contrôle et allocation de l'initiative				
Transi -sition	Initia -tive	*Handover* ou émetteur	*Takeover* ou receveur	Tâches ou Fonctions
X Master W/C	RECO MMAN DEUR	RECOMMANDEUR : Lancer les "attention getters" **HANDOVER**	PM : Percevoir les attention getters	RECOMMANDEUR : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
X Recom mandati ons immédia tes	RECO MMAN DEUR	RECOMMANDEUR : Afficher timeline recommandations immédiates **TAKEOVER - HANDOVER**	PM : Voir les recommandations	RECOMMANDEUR : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
X recomm andation s	RECO MMAN DEUR	RECOMMANDEUR : Afficher les recommandations **TAKEOVER - HANDOVER**	PM : Voir les recommandations immédiates	RECOMMANDEUR : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
x	RECO MMAN DEUR	RECOMMANDEUR : Afficher les alarmes actives **HANDOVER**	PM : Voir les alarmes actives	RECOMMANDEUR : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
x	RECO MMAN DEUR	RECOMMANDEUR : Afficher états OQCR **HANDOVER**	PM : Voir les états systèmes	RECOMMANDEUR : Lancer les "attention getters" Déclenche PM : Acquitter les alarmes
X Master W/C	PM	PM : Presser le Master W/C **HANDOVER**	RECOMMANDEUR : Recevoir pression Master W/C	PM : « Acquitter les alarmes » Recommandeur : « Soutenir l'acquittement des alarmes »
	RECO MMAN DEUR	RECOMMANDEUR : Stopper le son	PM : Entendre l'alerte sonore s'arrêter	
	RECO MMAN DEUR	RECOMMANDEUR : Eteindre le Master W/C	PM : Voir le Master W/C éteint	
X recomm andation s	PM	PM : Faire défiler **HANDOVER**	RECOMMANDEUR : Recevoir défilement	
	RECO MMAN DEUR	RECOMMANDEUR : Faire défiler les recommandations **HANDOVER**	PM : Voir le résultat	

x Page des états systèmes OQCR	PM	PM : Survoler états OQCR **HANDOVER**	RECOMMANDEUR : Recevoir sélection d'un état système	
	RECOMMANDEUR	RECOMMANDEUR : Mettre en valeur actions nécessaires **HANDOVER**	PM : Voir les actions de récupération nécessaires	
X recommandations	PM	PM : Sélectionner recommandation **HANDOVER**	RECOMMANDEUR : Recevoir sélection d'une recommandation	
	PM	PM : Sélectionner recommandation **HANDOVER**	Recevoir sélection d'une recommandation immédiate	
	RECOMMANDEUR	RECOMMANDEUR : Afficher explications **HANDOVER**	PM : Voir les explications	
x EMER CANCEL pb + Alarme	PM	PM : Presser le bouton EMER CANCEL **HANDOVER**	RECOMMANDEUR : Recevoir EMER CANCEL	
	RECOMMANDEUR	RECOMMANDEUR : Afficher résultat EMER CANCEL	PM : Voir la procédure	
	RECOMMANDEUR	Afficher la procédure non supprimée	PM : Voir la procédure	
X RECALL ALL	RECOMMANDEUR	RECOMMANDEUR : allumer RECALL ALL pb	PM : Voir le bouton RECALL ALL allumé	PM : "Choisir une recommendation" RECOMMANDEUR : « Soutenir la sélection »
	PM	PM : Presser le bouton RECALL ALL	RECOMMANDEUR : Recevoir RECALL ALL	
	RECOMMANDEUR	RECOMMANDEUR : Afficher les recommandations rappelées	PM : Voir les recommandations	
X RECALL LAST	RECOMMANDEUR	RECOMMANDEUR : Allumer RECALL LAST pb	PM : Voir le bouton RECALL LAST allumé	
	PM	PM : Presser le bouton RECALL LAST	RECOMMANDEUR : Recevoir RECALL LAST	
	RECOMMANDEUR	RECOMMANDEUR : Afficher la recommandation rappelée	PM : Voir la recommandation rappelée	
X ECP & action de récupération	PM	PM : Cliquer sur la checkbox **HANDOVER**	RECOMMANDEUR : Recevoir que le PM valide l'action de récupération	PM : « Réaliser la recommandation » RECOMMANDEUR : « Soutenir la réalisation de la recommandation »
	RECOMMANDEUR	RECOMMANDEUR : Cocher l'action de récupération	PM : Voir la case à cocher [de la ligne de procédure] cochée	

	RECOMMANDEUR	RECOMMANDEUR : Cocher l'action de récupération	PM : Voir l'action automatiquement cochée	
x	RECOMMANDEUR	RECOMMANDEUR : Allumer bouton CLEAR et afficher la barre d'effacement **HANDOVER**	PM : Voir la barre d'effacement et le bouton CLEAR allumé	
x	PM	PM : Presser le bouton CLEAR **HANDOVER**	RECOMMANDEUR : Recevoir CLEAR	
	RECOMMANDEUR	RECOMMANDEUR : Effacer la recommandation	PM : Voir le résultat du CLEAR	
	RECOMMANDEUR	RECOMMANDEUR : Effacer la recommandation immédiate	PM : Voir le résultat du CLEAR	
	RECOMMANDEUR	RECOMMANDEUR : Effacer alarme	PM : Voir le résultat du CLEAR	

3.2.2.3 Phase 2 du processus RCRAFT : Allocation de la responsabilité

Le Tableau 44 présente les colonnes du tableau RCRA décrivant l'allocation de la responsabilité entre le pilote *monitoring* et le recommandeur produit à la fin des itérations nécessaires de la sous-phase 4 de la phase 2 du processus RCRAFT (voir Chapitre 6). Tout comme le FWS, le recommandeur soutient le but du pilote *monitoring* en le guidant dans la résolution des alarmes. Le but du pilote *monitoring* est lui aussi inchangé (ramener l'avion dans un état fonctionnel). Le recommandeur et le pilote *monitoring* partagent le résultat attendu de résoudre toutes les alarmes lors de la résolution des alarmes. Les tâches et fonctions modifiant des DODs pouvant faire dévier ce résultat attendu sont à responsabilité. Par exemple, les objets logiciels concernant les recommandations peuvent faire dévier la résolution de toutes les alarmes. Le recommandeur a la responsabilité de filtrer les actions de récupération, trier les recommandations en immédiates ou non immédiates, définir le temps restant estimé à afficher pour les recommandations immédiates, ordonnancer les recommandations non-immédiates par priorité, grouper les actions de récupération par buts utilisateur. Le pilote *monitoring* a la responsabilité de décider de la recommandation à réaliser et de réaliser les actions de récupération qui la composent. Les lignes correspondantes sont sur fond bleu dans le Tableau 44.

Tableau 44. Colonnes du tableau décrivant l'allocation de la responsabilité entre le recommandeur (Recommandeur) et le pilote monitoring (PM)

Allocation de la responsabilité	
Tâche/ Fonction	Résultat attendu possiblement impacté
RECOMMANDEUR : Traiter une (ou plusieurs) alarme reçue d'un système avion	Résoudre toutes les alarmes
RECOMMANDEUR : Inhiber des alarmes	

RECOMMANDEUR : Combiner des alarmes	
RECOMMANDEUR : Lever une/des alarmes	
RECOMMANDEUR : Générer les recommandations	
RECOMMANDEUR : Filtrer les actions de récupération	
RECOMMANDEUR : Trier les groupes de lignes immédiates et non immédiates	
RECOMMANDEUR : Ordonnancer les recommandations non-immédiates par priorité	
RECOMMANDEUR : Estimer temps restant pour les recommandations immédiates	
RECOMMANDEUR : Ordonnancer les alarmes par priorité	
RECOMMANDEUR : Grouper les actions de récupérations par buts utilisateur	
RECOMMANDEUR : récupérer les infos systèmes	
RECOMMANDEUR : Sélectionner systèmes impactés	
RECOMMANDEUR : Afficher états OQCR	
RECOMMANDEUR : Lancer les "attention getters"	
RECOMMANDEUR : Afficher timeline recommandations immédiates	
RECOMMANDEUR : Afficher les recommandations	
RECOMMANDEUR : Afficher les alarmes actives	
PM : Se rappeler des réactions immédiates	
PM : Décider que l'observation est cohérente avec l'alarme	
PM : Décider de la recommandation à réaliser	
PM : Presser le bouton EMER CANCEL	
RECOMMANDEUR : Emer cancel alarme sélectionnée	
RECOMMANDEUR : Afficher résultat EMER CANCEL	
RECOMMANDEUR : Afficher résultat EMER CANCEL	
RECOMMANDEUR : Afficher la recommandation non supprimée	
PM : Presser le bouton RECALL LAST	
RECOMMANDEUR : RECAL LAST (rappeler dernière) recommandation	
RECOMMANDEUR : Afficher la recommandation rappelée	
PM : Presser le bouton RECALL ALL	
RECOMMANDEUR : RECALL ALL recommandations	
RECOMMANDEUR : Afficher les recommandations rappelées	
PM : Réaliser l'action de récupération d'une recommandation immédiate	
PM : Réaliser l'action de récupération	
PM : Cliquer sur la checkbox	
PM : Presser le bouton CLEAR	
PM : Décider qu'il n'y a plus de recommandations	
PM : Décider qu'il reste des recommandations	
RECOMMANDEUR : Détecter l'action de récupération	
RECOMMANDEUR : Cocher l'action de récupération	
RECOMMANDEUR : Détecter si la recommandation est terminée	
RECOMMANDEUR : Clear recommandation	
RECOMMANDEUR : Effacer la recommandation immédiate	
RECOMMANDEUR : Effacer la recommandation	

RECOMMANDEUR : Effacer alarme	

3.2.2.4 Phase 2 du processus RCRAFT : Allocation de l'autorité

Le Tableau 45 présente les colonnes du tableau RCRA décrivant l'allocation de l'autorité entre le recommandeur et le pilote *monitoring* produit à la fin des itérations nécessaires de la sous-phase 5 de la phase 2 du processus RCRAFT (voir Chapitre 6). Par exemple, le Tableau 45 indique que le pilote *monitoring* a l'autorité de définir l'ordre de réalisation des actions de récupération. Le recommandeur a l'autorité de définir l'ordre de priorité des actions de récupération. Le recommandeur défini les recommandations comme immédiates ou non. Également, le recommandeur défini une valeur du temps restant estimé pour les recommandations d'actions de récupération immédiates. Les lignes du Tableau 45 correspondantes sont sur fond bleu.

Tableau 45. Colonnes du tableau RCRA décrivant l'allocation de l'autorité entre le recommandeur (Recommandeur) et le pilote monitoring (PM)

Allocation de l'autorité	
Tâche/Fonction	**Contrainte**
RECOMMANDEUR : Combiner des alarmes	Défini la valeur de l'alarme
RECOMMANDEUR : Inhiber les alarmes	Masque des alarmes au PM
RECOMMANDEUR : Générer les recommandations	Défini les actions à réaliser pour résoudre l'alarme
RECOMMANDEUR : Filtrer les actions de récupération	Masque des actions
RECOMMANDEUR : Trier les groupes de lignes immédiates et non immédiates	Défini l'ordre de priorité des recommandations
RECOMMANDEUR : Ordonnancer les recommandations non-immédiates par priorité	Propose un ordre de réalisation des recommandations au PM
RECOMMANDEUR : Estimer temps restant pour les recommandations immédiates	Défini le temps restant au PM pour réaliser chaque recommandations immédiates
RECOMMANDEUR : Grouper les actions de récupérations par buts utilisateur	Défini le but atteint par la réalisation des actions de récupérations
RECOMMANDEUR : Sélectionner systèmes impactés	Défini les systèmes dont les états seront présentés
RECOMMANDEUR : Ecouter EMER CANCEL	Le PM peut appuyer sur le bouton pour EMER CANCEL
RECOMMANDEUR : Ecouter Master W/C	Le PM peut appuyer sur le bouton pour couper les attention getters
RECOMMANDEUR : Ecouter bouton de validation	Le PM peut appuyer sur le bouton pour valider
PM : Presser le Master W/C	Demande l'arrêt des attentions getters
PM : Faire défiler	Demande de faire défiler les alarmes

PM : Survoler états OQCR	Demande les actions de récupération nécessaire pour atteindre l'état
PM : Sélectionner recommandation	Demande les explications de la recommandation
PM : Déterminer les conséquences	Défini les conséquences possibles de la défaillance sur l'état de l'avion et sur la mission
PM : Décider que l'observation est cohérente avec l'alarme	Défini que l'alarme est cohérente avec les informations des pages SD et de l'overhead panel
PM : Décider de la recommandation à réaliser	Défini l'ordre de réalisation des actions de récupération
PM : Presser le bouton EMER CANCEL	Demande la suppression de l'alarme la plus prioritaire
RECOMMANDEUR : Afficher l'alarme non supprimée	Refus de la suppression (l'alarme n'est pas supprimable)
RECOMMANDEUR : Ecouter RECALL LAST	Le PM peut appuyer sur le bouton pour RECALL LAST
PM : Presser le bouton RECALL LAST	Demande le rappel de la dernière procédure effacée
RECOMMANDEUR : Ecouter RECALL ALL	Le PM peut appuyer sur le bouton pour RECALL ALL
PM : Presser le bouton RECALL ALL	Demande le rappel de toutes les procédures effacées
PM : Réaliser l'action de récupération	Réalise l'action de récupération
PM : Réaliser l'action de récupération d'une recommandation immédiate	Réalise l'action de récupération
PM : Cliquer sur la checkbox	Défini l'action comme étant terminée
RECOMMANDEUR : Détecter l'action de récupération	Défini l'action comme étant terminée
RECOMMANDEUR : Détecter si la procédure est terminée	Défini la procédure comme étant terminée
PM : Décider que la (les) action de récupération de la recommandation sont terminées	
RECOMMANDEUR : Ecouter bouton CLEAR	le PM doit appuyer sur CLEAR pour effacer la procédure
PM : Presser le bouton CLEAR	Demande l'effacement de la procédure

3.2.2.5 Phase 3 du processus RCRAFT : Modélisation RCRAFT

L'analyse RCRAFT de la gestion des alarmes avec le pilote *monitoring* et le recommandeur a été réalisée en phase 2. Les modèles de tâches ségrégués et de fonctions du pilote *monitoring* et du recommandeur intégrant la modélisation des éléments RCRAFT sont consultables en annexes 3.1 et 3.2.

4 Comparaison des automatisations

L'analyse RCRAFT de la gestion des alarmes par le pilote *monitoring* avec deux systèmes contenant de l'automatisation différents (FWS et le recommandeur), nous permet de comparer

ces deux systèmes en termes d'allocation de ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Nous concentrons notre comparaison sur la gestion des actions de récupération par les rôles.

4.1 Comparaison de l'allocation des tâches et des fonctions

Le FWS et le recommandeur inhibent, combinent et ordonnancent les alarmes reçues. Les fonctions du recommandeur et du FWS diffèrent sur leur gestion des actions de récupération. Le recommandeur génère, filtre, ordonnance et affiche des recommandations et des recommandations immédiates d'actions de récupération. Le FWS affiche la procédure présentant les actions de récupération pour l'alarme la plus prioritaire. Le FWS et le recommandeur permettent au pilote de stopper les *attention getters* grâce au bouton MASTER W/C. Afin de soutenir la compréhension du pilote des alarmes et de la situation actuelle : le FWS lui permet de consulter les pages systèmes des différents systèmes avions et le recommandeur lui permet de consulter les états OQCR des systèmes impactés, montre les alarmes et états systèmes à l'origine des recommandations et indique les recommandations à réaliser pour atteindre un certain état système.

Le pilote monitoring peut décider qu'une alarme est intempestive et peut avec les deux systèmes supprimer une alarme intempestive. Avec le FWS, pour accéder à une autre procédure que la première affichée, le pilote *monitoring* doit supprimer ou rappeler des alarmes. Avec le recommandeur, le pilote *monitoring* doit choisir les premières actions de récupération à réaliser. De la même manière, le pilote *monitoring* peut supprimer et rappeler des alarmes avec le recommandeur. La réalisation des actions de récupération est similaire avec les deux systèmes.

4.2 Comparaison de l'allocation des ressources

Le recommandeur affiche des recommandations d'action de récupération et non plus des procédures comme le fait le FWS. En conséquence, les ressources utilisées dans le cadre de la résolution des alarmes ne sont plus les mêmes. La ressource « procédure » n'apparait pas dans le Tableau 42 de l'allocation des ressources entre le recommandeur et le pilote *monitoring*. Nous constatons également une différence dans le partage de la « stratégie de filtrage » du FWS et du recommandeur. Le FWS ne partage pas la stratégie de filtrage avec le pilote *monitoring*. Cependant, il affiche les états des systèmes, les alarmes actives et les limitations pour que le pilote *monitoring* comprenne l'origine de l'alarme. Le recommandeur met en valeur les données affichées qui sont à l'origine de la recommandation. Le recommandeur partage, en partie, la stratégie de filtrage des recommandations.

4.3 Comparaison de l'allocation des transitions de contrôle et de l'initiative

L'affichage d'une procédure et la désactivation possible des « attention getter » par la pression du bouton Master W/C cause une transition de contrôle et une interruption des tâches du pilote *monitoring* par le FWS (le FWS a l'initiative d'un *takeover*). Le pilote monitoring n'a pas le contrôle sur la première procédure affichée (les premières actions à réaliser) : il n'y a pas de transition de contrôle pour l'élément de la couche interface « première procédure ». Le pilote

monitoring doit utiliser le bouton EMER CANCEL (*takeover*) pour demander la suppression de la procédure et l'alarme la plus prioritaire (*handover*) si cette procédure est supprimable. De la même manière, l'affichage de nouvelles recommandations et la désactivation possible des « attention getter » par la pression du bouton Master W/C cause une transition de contrôle et une suspension des tâches du pilote *monitoring* par le recommandeur. Cependant, le contrôle sur les recommandations est rendu au pilote monitoring par le recommandeur (*handover*).

4.4 Comparaison de l'allocation de la responsabilité

Le FWS a la responsabilité sur la sélection et l'affichage de la première procédure (les premières actions de récupération à réaliser). Le pilote monitoring est responsable de la suppression de la première procédure s'il utilise EMER CANCEL. Le recommandeur n'a pas la responsabilité de la sélection des premières actions de récupération à réaliser. Le pilote *monitoring* a la responsabilité de décider quelles sont les premières actions de récupération à réaliser. Avec le recommandeur, le pilote *monitoring* a plus de responsabilité pour la tâche consistant à choisir les premières actions de récupération à réaliser.

4.5 Comparaison de l'allocation de l'autorité

Seul le pilote *monitoring* a l'autorité de définir l'ordre de réalisation des actions de récupération avec le recommandeur. Le recommandeur a seulement l'autorité de définir l'ordre de priorité des actions de récupération contrairement au FWS qui possède ces deux autorités. En conséquence, le pilote *monitoring* peut choisir de reprendre ces tâches où il les avait interrompues pour acquitter les nouvelles alarmes sans avoir à redéfinir la sélection du système. Avec le recommandeur, le pilote *monitoring* a l'autorité de choisir l'ordre de réalisation des actions de récupération sans avoir à supprimer des alarmes supprimables pour accéder aux actions de récupération désirées. Le pilote *monitoring* conserve l'autorité de demander la suppression d'alarmes supprimables.

Les deux systèmes définissent quand réaliser certaines actions de récupération. Si le FWS affiche une alarme rouge, la procédure associée est à réaliser dans les plus brefs délais. Le recommandeur affiche sur une timeline les recommandations définies comme immédiates. Cependant, la contrainte définie par le système est différente selon le cas. Le recommandeur défini une valeur du temps restant estimé pour les recommandations d'actions de récupération immédiates. Le FWS indique seulement que les actions de récupération des alarmes rouges doivent être réalisées dans les plus bref délais.

5 Conclusion

Dans ce chapitre, nous avons présenté une étude de cas qui nous permet de valider la faisabilité de notre approche : l'application du cadre conceptuel RCRAFT avec le processus RCRAFT. Cette étude de cas concerne la gestion des alarmes par le pilote *monitoring* grâce au système de gestion des alarmes actuel : le FWS, et avec le système alternatif de gestion des alarmes (ayant le rôle de recommandeur) conceptualisé durant le lot de travail MAGIC.

En comparant ces systèmes en termes d'allocation de ressources, des transitions de contrôle et de l'initiative, de responsabilité et d'autorité, nous avons pu vérifier l'objectif de conception du

système alternatif de gestion des alarmes proposé : donner le choix au pilote *monitoring*. En effet, le pilote *monitoring* a la responsabilité et l'autorité de choisir les premières actions de récupération à réaliser avec le système alternatif de gestion des alarmes.

Nous avons également pu décrire de manière explicite les différences entre ces systèmes en termes d'allocation RCRAFT. Nous avons, par exemple, pu décrire que bien le système alternatif de gestion des alarmes donne une plus grande liberté de choix des actions de récupération au pilote *monitoring*, la timeline utilisée par le système pour les recommandations immédiates définit une durée en plus d'une priorité pour ces actions de récupération. Sur cette fonction, le système alternatif de gestion des alarmes a plus d'autorité que le FWS.

Dans cette étude de cas, nous avons :

- Décrit de manière complète et non ambigüe les tâches du pilote *monitoring* pour ramener l'avion dans un état fonctionnel.
- Décrit de manière complète et non ambigüe les fonctions du FWS pour guider le pilote *monitoring* dans la résolution des alarmes.
- Décrit de manière complète et non ambigüe les fonctions du système alternatif de gestion des alarmes pour guider le pilote *monitoring* dans la résolution des alarmes.
- Décrit la collaboration entre les rôles pilote *monitoring* et FWS.
- Décrit la collaboration entre les rôles pilote *monitoring* et système de recommandation (système alternatif de gestion des alarmes).
- Décrit l'allocation des ressources entre les rôles pilote *monitoring* et FWS.
- Décrit l'allocation des transitions de contrôle et de l'initiative entre les rôles pilote *monitoring* et FWS.
- Décrit l'allocation de l'autorité entre les rôles pilote *monitoring* et FWS.
- Décrit l'allocation de la responsabilité entre les rôles pilote *monitoring* et FWS.
- Décrit l'allocation des ressources entre les rôles pilote *monitoring* et système de recommandation (système alternatif de gestion des alarmes).
- Décrit l'allocation des transitions de contrôle et de l'initiative entre les rôles pilote *monitoring* et système de recommandation (système alternatif de gestion des alarmes).
- Décrit l'allocation de l'autorité entre les rôles pilote monitoring et système de recommandation (système alternatif de gestion des alarmes).
- Décrit l'allocation de la responsabilité entre les rôles pilote monitoring et système de recommandation (système alternatif de gestion des alarmes).
- Comparer l'allocation de chacun des éléments RCRAFT entre ces deux automatisations

Cependant, cette étude de cas ne nous permet pas la validation de notre approche à d'autres domaines d'application, notre étude de cas relevant du domaine de l'aéronautique. De plus, le système alternatif de gestion des alarmes n'a pas été développé et seulement conceptualisé. Cette étude de cas applique donc l'approche RCRAFT seulement à un seul système réel : le FWS pour la gestion des alarmes. Également, notre étude de cas montre que les modèles de tâches et de fonctions comportent un nombre important de données et leur réalisation demandent des compétences de modélisation et tâches. Aussi, une connaissance approfondie

du comportement des utilisateurs et des systèmes est requise, la qualité de l'analyse RCRAFT étant dépendante de la connaissance du comportement de chacun des rôles.

Chapitre 10. Conclusion et Perspectives

L'objectif de cette thèse était de démontrer qu'il est possible de décrire de façon systématique l'allocation des ressources, des transitions de contrôle et de l'initiative, la responsabilité et l'autorité par rapport à allocation des fonctions et des tâches entre les utilisateurs et les systèmes contenant de l'automatisation. Nous avons proposé une approche à base de modèles de tâches associée à un processus outillé de description de l'allocation des ressources, des transitions de contrôle et de l'initiative, la responsabilité, l'autorité, des fonctions et des tâches dans cet objectif. Nous avons présenté cette approche et le processus outillé associé à travers les contributions suivantes :

- Le cadre conceptuel RCRAFT qui propose un cadre structuré pour l'analyse et la description de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité entre les utilisateurs et les systèmes contenant de l'automatisation par rapport à l'allocation des fonctions et des tâches (voir Chapitre 4). Ce cadre conceptuel permet de décrire les tâches et les fonctions des utilisateurs et des systèmes contenant de l'automatisation en identifiant les allocations des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité. Une telle description permet de lever les ambiguïtés de description des niveaux d'automatisation de (R. Parasuraman et al., 2000) et de la (SAE J3016, 2019). Ces ambiguïtés pouvant résulter sur la conception de plusieurs systèmes avec des allocations de ressources, de transitions de contrôle et de l'initiative, de responsabilité et d'autorité différentes pour un même niveau d'automatisation. Ces démonstrations sont publiées dans les travaux (Elodie Bouzekri, Canny, Martinie, et al., 2019b) et (Elodie Bouzekri et al., 2020).
- Les extensions apportées à la notation HAMSTERS pour la description de la coopération par des protocoles de coopération dans des modèles de tâches ségrégués et des modèles de fonctions (voir Chapitre 5). Ces extensions permettent de décrire les éléments RCRAFT dans les modèles de tâches ségrégués et de fonctions.
- Le processus RCRAFT permettant l'analyse et la description RCRAFT de manière systématique (voir Chapitre 6). La processus RCRAFT se compose des trois phases suivantes :
    - Phase de modélisations des tâches explicitant l'allocation des fonctions et des tâches produisant les modèles de tâches ségrégués et de fonctions.
    - Phase de description RCRAFT produisant le tableau RCRA
    - Phase de modélisation RCRAFT proposant de modéliser les allocations décrites dans le tableau RCRA dans les modèles de tâches ségrégués et de fonctions.
- Les extensions apportées à l'outil HAMSTERS correspondantes aux extensions apportées à la notation HAMSTERS (voir Chapitre 7). Ces extensions permettent de soutenir le processus RCRAFT.
- L'application du processus RCRAFT à une étude de cas industrielle afin de montrer a faisabilité de l'approche proposée (voir Chapitre 9).

Des limitations de l'approche et du processus outillé proposés sont liées au périmètre de validation de faisabilité. La faisabilité du processus RCRAFT est partiellement validée par son application à une étude de cas industrielle appartenant au domaine de l'aéronautique. La validation de l'approche et du processus outillé associé est donc partielle et restreinte à ce domaine d'application. Une autre limitation est que les travaux de cette thèse n'incluent pas d'études avec les utilisateurs finaux (concepteurs ou experts en automatisation) afin de recueillir leurs besoins en termes de notation et d'outillage pour l'analyse et la description de l'automatisation selon leur domaine d'application. Également, des tests avec des concepteurs et ingénieurs de systèmes de commande et contrôle de l'approche et du processus associé est nécessaire afin d'évaluer que leurs besoins pour la conception de système contenant de l'automatisation sont bien couverts.

Il existe également des limitations de l'approche et du processus outillé proposés liées au support fourni pour la modélisation. En effet, les extensions ont été proposées pour l'outil HAMSTERS. Nous avons illustré et utilisé ces extensions de modélisation seulement pour cet outil et non pas pour les autres outils de modélisation de tâches présentés au Chapitre 3.2.3. Cependant, les extensions de la notation proposées sont transférables et réutilisables pour d'autres notations et outils existants comme CTT (Paternò, 2004) par exemple.

Des perspectives à court et à moyen terme peuvent être dégagées à partir des limitations des contributions de cette thèse :

- L'approche RCRAFT et le processus associé peuvent être appliqués à d'autres domaines d'application où l'on trouve des systèmes contenant de l'automatisation, comme par exemple les systèmes d'aide à la personne en situation de handicap ou les systèmes du domaine médical comme les systèmes d'aide à la décision clinique pour la supervision de patients (Vives-Boix et al., 2017).
- L'approche RCRAFT et le processus associé peuvent faire l'objet d'études avec des experts en automatisation afin d'identifier et d'adapter l'approche proposée à des besoins spécifiques à différents domaines d'application. Également, une étude avec ces experts pourrait permettre un retour sur la notation et les éléments graphiques utilisés dans l'outil.

Une autre perspective à court ou à moyen terme est de fournir un support outillé au tableau RCRA en plus de la modélisation de l'allocation des ressources, des transitions de contrôle et de l'initiative, de la responsabilité et de l'autorité dans les modèles de tâches ségrégués et de fonctions. Par exemple, l'outil pourrait permettre de générer la liste des DODs des différents modèles de tâches ségrégués et de fonction. Également, l'outil pourrait permettre de parcourir les tâches et fonctions manipulant les DODs de la liste afin de réaliser les étapes de la phase 2 du processus RCRAFT. Ainsi, la totalité du processus RCRAFT pourrait bénéficier d'un support outillé.

De plus, l'identification de patterns d'allocation RCRAFT pourrait constituer un travail futur afin d'apporter un support supplémentaire aux concepteurs de systèmes contenant de l'automatisation. Ceci dans le but de permettre une identification ou une description simplifiée d'allocation de tâches et de fonctions avec une allocation des ressources, des transitions de

contrôle, de la responsabilité et de l'autorité particulière. Par exemple, la description d'un pattern d'une allocation de tâches et de fonctions où le rôle alloué à l'utilisateur peut, à tout moment, initier une transition de contrôle sur une tâche (de conduite par exemple) et définir dynamiquement l'allocation des rôles entre les acteurs (prendre le rôle de conducteur par exemple) afin d'avoir l'autorité sur une tâche.

Également, le travail préliminaire réalisé consistant à identifier les problèmes et à définir des recommandations pour l'intégration de systèmes de recommandations dans le cockpit d'avion civil dans (E Bouzekri et al., 2017) et (E. Bouzekri et al., 2018) peut être enrichi par le cadre conceptuel et le processus RCRAFT proposé. En effet, les problèmes et les recommandations pour l'intégration de systèmes de recommandations dans le cockpit d'avion civil en garantissant les propriétés de sureté de fonctionnement et de sécurité peuvent désormais être définis en termes d'allocation de ressources, de transitions de contrôle et d'initiative, de responsabilité, d'autorité de fonctions et de tâches de manière formelle. De plus, l'impact de ces recommandations en termes d'allocation RCRAFT peut avoir un impact sur les propriétés architecturales de ces systèmes qui serait à étudier.

La poursuite du travail préliminaire réalisé afin d'étudier les impacts des caractéristiques de l'automatisation sur l'expérience utilisateur (Bouzekri, Elodie et al., 2019) grâce à une approche basée sur les modèles de tâches est une autre perspective possible aux travaux présentés dans cette thèse. De la même manière que l'étude de l'impact de l'automatisation sur la réalisation des tâches utilisateurs et des recommandations proposées pour soutenir les propriétés *trust* (J. D. Lee & See, 2004) (Hoff & Bashir, 2015) et *situation awareness* (Endsley, 1999), nous pourrions étudier l'impact de différentes allocations RCRAFT sur la propriété UX des systèmes interactifs contenant de l'automatisation. Ainsi, il pourrait être possible de proposer des recommandations d'allocation RCRAFT pour la conception de système interactif contenant de l'automatisation assurant le soutien des différentes dimensions de la propriété UX.

Publications personnelles

Bouzekri, E, Canny, A., Fayollas, C., Martinie, C., Palanque, P., Barboni, E., Deleris, Y., & Gris, C. (s. d.). A List of Pre-Requisites to Make Recommender Systems Deployable in Critical Context. 14.

Bouzekri, E., Canny, A., Fayollas, C., Martinie, C., Palanque, P., Barboni, E., Deleris, Y., & Gris, C. (2018). Engineering issues related to the development of a recommender system in a critical context : Application to interactive cockpits. International Journal of Human-Computer Studies. https://doi.org/10.1016/j.ijhcs.2018.05.001

Bouzekri, Elodie. (2018). Model-Based Approach to Design and Develop Usable and Dependable Recommender Systems. Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems, 17:1–17:7. https://doi.org/10.1145/3220134.3220147

Bouzekri, Elodie, Canny, A., Martinie, C., & Palanque, P. (s. d.-a). A Generic Software and Hardware Architecture for Hybrid Interactive Systems. 5.

Bouzekri, Elodie, Canny, A., Martinie, C., & Palanque, P. (s. d.-b). Specific Task Modeling for Cyber Physical Systems. 5.

Bouzekri, Elodie, Canny, A., Martinie, C., & Palanque, P. (2020). Characterizing Sets of Systems : Representation and Analysis of Across-Systems Properties. In J. Abdelnour Nocera, A. Parmaxi, M. Winckler, F. Loizides, C. Ardito, G. Bhutkar, & P. Dannenmann (Éds.), Beyond Interactions (p. 84-96). Springer International Publishing. https://doi.org/10.1007/978-3-030-46540-7_9

Bouzekri, Elodie, Canny, A., Martinie, C., Palanque, P., & Gris, C. (2019a). Deep System Knowledge Required : Revisiting UCD Contribution in the Design of Complex Command and Control Systems. In D. Lamas, F. Loizides, L. Nacke, H. Petrie, M. Winckler, & P. Zaphiris (Éds.), Human-Computer Interaction – INTERACT 2019 (p. 699-720). Springer International Publishing. https://doi.org/10.1007/978-3-030-29381-9_42

Bouzekri, Elodie, Canny, A., Martinie, C., Palanque, P., & Gris, C. (2019b). Using Task Descriptions with Explicit Representation of Allocation of Functions, Authority and Responsibility to Design and Assess Automation. In B. R. Barricelli, V. Roto, T. Clemmensen, P. Campos, A. Lopes, F. Gonçalves, & J. Abdelnour-Nocera (Éds.), Human Work Interaction Design. Designing Engaging Automation (p. 36-56). Springer International Publishing.

Bouzekri, Elodie, Canny, A., Martinie De Almeida, C., Palanque, P., Barboni, E., Navarre, D., Gris, C., & Deleris, Y. (2019). Revisiting system's pages in engine indication and alerting system for flight crew using the DSCU architecture and the OQCR system generic state description. 1-9. https://oatao.univ-toulouse.fr/24919/

Bouzekri, Elodie, Martinie, C., & Palanque, P. (2020). A-RCRAFT Framework for Analysing Automation : Application to SAE J3016 Levels of Driving Automation. In C. Olaverri-Monreal, F. García-Fernández, & R. J. F. Rossetti (Éds.), Human Factors in Intelligent Vehicles. River Publishers. https://hal.archives-ouvertes.fr/hal-02939069

Bouzekri, Elodie, Martinie, Célia, Wallner, Günter, Palanque, Philippe, Bernhaupt, Regina, & Systemic Change. (2019). Investigating the effect of automation on user experience : Enriching a task-modeling notation. Workshop on Everyday Automation Experience. https://research.tue.nl/nl/publications/investigating-the-effect-of-automation-on-user-experience-enriching-a-taskmodeling-notation(3ef9f9b5-d245-4356-a70a-016001b0e022).html

Canny, A., Bouzekri, E., Martinie, C., & Palanque, P. (s. d.). On the Importance of Supporting Multiple Stakeholders Points of View for the Testing of Interactive Systems. 9.

Canny, A., Bouzekri, E., Martinie, C., & Palanque, P. (2019). Rationalizing the Need of Architecture-Driven Testing of Interactive Systems. In C. Bogdan, K. Kuusinen, M. K. Lárusdóttir, P. Palanque, & M. Winckler (Éds.), Human-Centered Software Engineering (p. 164-186). Springer International Publishing. https://doi.org/10.1007/978-3-030-05909-5_10

Martinie, C., Palanque, P., Bouzekri, E., Cockburn, A., Canny, A., & Barboni, E. (2019). Analysing and Demonstrating Tool-Supported Customizable Task Notations. Proc. ACM Hum.-Comput. Interact., 3(EICS), 12:1–12:26. https://doi.org/10.1145/3331154

Références

Annett, J., & Duncan, K. D. (1967). *TASK ANALYSIS AND TRAINING DESIGN*. https://eric.ed.gov/?id=ED019566

Annett, John. (2004). Hierarchical task analysis (HTA). In *Handbook of human factors and ergonomics methods* (p. 355–363). CRC Press.

Avizienis, A., Laprie, J. C., Randell, B., & Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, *1*(1), 11-33. https://doi.org/10.1109/TDSC.2004.2

Awad, E., Levine, S., Kleiman-Weiner, M., Dsouza, S., Tenenbaum, J. B., Shariff, A., Bonnefon, J.-F., & Rahwan, I. (2018). Blaming humans in autonomous vehicle accidents : Shared responsibility across levels of automation. *arXiv:1803.07170 [cs]*. http://arxiv.org/abs/1803.07170

Bainbridge, L. (1983). IRONIES OF AUTOMATION. In G. Johannsen & J. E. Rijnsdorp (Éds.), *Analysis, Design and Evaluation of Man–Machine Systems* (p. 129-135). Pergamon. https://doi.org/10.1016/B978-0-08-029348-6.50026-9

Bass, L., Little, R., Pellegrino, R., Reed, S., Seacord, R., & Sheppard, S. (1992). The Arch model : Seeheim revisited (version 1.0). *The UIMS Developers Workshop*.

Bastide, R., & Palanque, P. (1999). A Visual and Formal Glue between Application and Interaction. *Journal of Visual Languages & Computing*, *10*(5), 481-507. https://doi.org/10.1006/jvlc.1999.0127

Battiste, V., Lachter, J., Brandt, S., Alvarez, A., Strybel, T. Z., & Vu, K.-P. L. (2018). Human-Automation Teaming : Lessons Learned and Future Directions. In S. Yamamoto & H. Mori (Éds.), *Human Interface and the Management of Information. Information in Applications and Services* (p. 479-493). Springer International Publishing.

Baumeister, L. K., John, B. E., & Byrne, M. D. (2000). A comparison of tools for building GOMS models. *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, 502–509. https://doi.org/10.1145/332040.332485

Bernhaupt, R. (2015). User Experience Evaluation Methods in the Games Development Life Cycle. In R. Bernhaupt (Éd.), *Game User Experience Evaluation* (p. 1-8). Springer International Publishing. https://doi.org/10.1007/978-3-319-15985-0_1

Bertin, J. (1973). *Sémiologie graphique : Les diagrammes-Les réseaux-Les cartes*. Gauthier-VillarsMouton & Cie.

Billings, C. E. (1991). *Human-centered Aircraft Automation : A Concept and Guidelines*. National Aeronautics and Space Administration, Ames Research Center.

Bouzekri, E, Canny, A., Fayollas, C., Martinie, C., Palanque, P., Barboni, E., Deleris, Y., & Gris, C. (2017). *A List of Pre-Requisites to Make Recommender Systems Deployable in Critical Context*. 14.

Bouzekri, E., Canny, A., Fayollas, C., Martinie, C., Palanque, P., Barboni, E., Deleris, Y., & Gris, C. (2018). Engineering issues related to the development of a recommender system in a critical context : Application to interactive cockpits. *International Journal of Human-Computer Studies*. https://doi.org/10.1016/j.ijhcs.2018.05.001

Bouzekri, Elodie, Canny, A., Martinie, C., Palanque, P., & Gris, C. (2019a). Deep System Knowledge Required : Revisiting UCD Contribution in the Design of Complex Command and Control Systems. In D. Lamas, F. Loizides, L. Nacke, H. Petrie, M. Winckler, & P. Zaphiris (Éds.), *Human-Computer Interaction – INTERACT 2019* (p. 699-720). Springer International Publishing. https://doi.org/10.1007/978-3-030-29381-9_42

Bouzekri, Elodie, Canny, A., Martinie, C., Palanque, P., & Gris, C. (2019b). Using Task Descriptions with Explicit Representation of Allocation of Functions, Authority and

Responsibility to Design and Assess Automation. In B. R. Barricelli, V. Roto, T. Clemmensen, P. Campos, A. Lopes, F. Gonçalves, & J. Abdelnour-Nocera (Éds.), *Human Work Interaction Design. Designing Engaging Automation* (p. 36-56). Springer International Publishing.

Bouzekri, Elodie, Canny, A., Martinie De Almeida, C., Palanque, P., Barboni, E., Navarre, D., Gris, C., & Deleris, Y. (2019). *Revisiting system's pages in engine indication and alerting system for flight crew using the DSCU architecture and the OQCR system generic state description*. 1-9. https://oatao.univ-toulouse.fr/24919/

Bouzekri, Elodie, Martinie, C., & Palanque, P. (2020). A-RCRAFT Framework for Analysing Automation : Application to SAE J3016 Levels of Driving Automation. In C. Olaverri-Monreal, F. García-Fernández, & R. J. F. Rossetti (Éds.), *Human Factors in Intelligent Vehicles*. River Publishers. https://hal.archives-ouvertes.fr/hal-02939069

Bouzekri, Elodie, Martinie, Célia, Wallner, Günter, Palanque, Philippe, & Bernhaupt, Regina. (2019). Investigating the effect of automation on user experience : Enriching a task-modeling notation. *Workshop on Everyday Automation Experience*. https://research.tue.nl/nl/publications/investigating-the-effect-of-automation-on-user-experience-enriching-a-taskmodeling-notation(3ef9f9b5-d245-4356-a70a-016001b0e022).html

Boy, G. A. (2013). Orchestrating Situation Awareness and Authority in Complex Socio-technical Systems. In M. Aiguier, Y. Caseau, D. Krob, & A. Rauzy (Éds.), *Complex Systems Design & Management* (p. 285-296). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-34404-6_19

Boy, G. A. (1998). Cognitive function analysis for human-centered automation of safety-critical systems. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '98*, 265-272. https://doi.org/10.1145/274644.274682

Bradshaw, J. M., Hoffman, R. R., Woods, D. D., & Johnson, M. (2013). The Seven Deadly Myths of « Autonomous Systems ». *IEEE Intelligent Systems*, *28*(3), 54-61. https://doi.org/10.1109/MIS.2013.70

Brown, M., & Leveson, N. G. (1998). Modeling controller tasks for safety analysis. *Workshop on Human Error and System Development, Seattle*.

Buchholz, G., & Forbrig, P. (2017). Extended Features of Task Models for Specifying Cooperative Activities. *Proceedings of the ACM on Human-Computer Interaction*, *1*(EICS), 1-21. https://doi.org/10.1145/3095809

Buschmann, F., Henney, K., & Schmidt, D. C. (2007). *Pattern-Oriented Software Architecture, On Patterns and Pattern Languages*. John Wiley & Sons.

Caffiau, S., Scapin, D., Girard, P., Baron, M., & Jambon, F. (2010). Increasing the expressive power of task analysis : Systematic comparison and empirical assessment of tool-supported task models. *Interacting with Computers*, *22*(6), 569-593. https://doi.org/10.1016/j.intcom.2010.06.003

Card, S. K., Moran, T. P., & Newell, A. (1980). The keystroke-level model for user performance time with interactive systems. *Communications of the ACM*, *23*(7), 396–410.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The Psychology of Human-Computer Interaction*. CRC Press.

Card, S., MORAN, T., & Newell, A. (1986). The model human processor- An engineering model of human performance. *Handbook of perception and human performance.*, *2*(45–1).

Chapanis, A. (1965). On the allocation of functions between men and machines. *Occupational Psychology*, *39*(1), 1–11.

Clark, R. E., & Estes, F. (1996). Cognitive task analysis for training. *International Journal of Educational Research, 25*(5), 403-417. https://doi.org/10.1016/S0883-0355(97)81235-9

Clough, B. T. (2002). *Metrics, Schmetrics ! How The Heck Do You Determine A UAV's Autonomy Anyway*. AIR FORCE RESEARCH LAB WRIGHT-PATTERSON AFB OH, AIR FORCE RESEARCH LAB WRIGHT-PATTERSON AFB OH. http://www.dtic.mil/docs/citations/ADA515926

Cockton, G., & Gram, C. (1996). *Design Principles for Interactive Software*. Springer Science & Business Media.

Conversy, S., Barboni, E., Navarre, D., & Palanque, P. (2008). Improving Modularity of Interactive Software with the MDPC Architecture. In J. Gulliksen, M. B. Harning, P. Palanque, G. C. van der Veer, & J. Wesson (Éds.), *Engineering Interactive Systems* (p. 321-338). Springer. https://doi.org/10.1007/978-3-540-92698-6_20

Coutaz, J. (1987). PAC, an Object Oriented Model for Dialog Design. In H.-J. Bullinger & B. Shackel (Éds.), *Human–Computer Interaction–INTERACT '87* (p. 431-436). North-Holland. https://doi.org/10.1016/B978-0-444-70304-0.50074-1

Cronel, M., Dumas, B., Palanque, P., & Canny, A. (2019). MIODMIT : A Generic Architecture for Dynamic Multimodal Interactive Systems. In C. Bogdan, K. Kuusinen, M. K. Lárusdóttir, P. Palanque, & M. Winckler (Éds.), *Human-Centered Software Engineering* (p. 109-129). Springer International Publishing. https://doi.org/10.1007/978-3-030-05909-5_7

Dearden, Andrew. (2001). IDA-S : A Conceptual Framework for Partial Automation. In A. Blandford, J. Vanderdonckt, & P. Gray (Éds.), *People and Computers XV—Interaction without Frontiers* (p. 213-228). Springer. https://doi.org/10.1007/978-1-4471-0353-0_13

Dearden, Andy, Harrison, M., & Wright, P. (2000). Allocation of function : Scenarios, context and the economics of effort. *International Journal of Human-Computer Studies*, *52*(2), 289-318. https://doi.org/10.1006/ijhc.1999.0290

Dekker, S. W. A., & Woods, D. D. (2002). MABA-MABA or Abracadabra? Progress on Human–Automation Co-ordination. *Cognition, Technology & Work*, *4*(4), 240-244. https://doi.org/10.1007/s101110200022

Diaper, D. (2001). Task analysis for knowledge descriptions (TAKD) : A requiem for a method. *Behaviour & Information Technology*, *20*(3), 199-212. https://doi.org/10.1080/01449290120610

Dix, A. J. (1991). *Formal methods for interactive systems*. Academic Press.

Draper, J. V. (1995). Teleoperators for advanced manufacturing : Applications and human factors challenges. *International Journal of Human Factors in Manufacturing*, *5*(1), 53-85. https://doi.org/10.1002/hfm.4530050105

EASA. (2014). *"CS-25." Certification Specifications and Acceptable Means of Compliance for Large Aeroplanes.*

Eckstein, R. (2007). Java se application design with mvc. *Sun Microsystems Inc*.

Ellis, C. A., Gibbs, S. J., & Rein, G. (1991). Groupware : Some issues and experiences. *Communications of the ACM*, *34*(1), 39–58. https://doi.org/10.1145/99977.99987

Endsley, M. R. (1995). Toward a Theory of Situation Awareness in Dynamic Systems. *Human Factors*, *37*(1), 32-64. https://doi.org/10.1518/001872095779049543

Endsley, M. R. (1999). Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics*, *42*(3), 462-492. https://doi.org/10.1080/001401399185595

Endsley, M. R. (2016). *Designing for situation awareness : An approach to user-centered design*. CRC press.

Endsley, M. R. (2017). From Here to Autonomy : Lessons Learned From Human–Automation Research. *Human Factors*, *59*(1), 5-27. https://doi.org/10.1177/0018720816681350

Endsley, M. R. (1987). The application of human factors to the development of expert systems for advanced cockpits. *Proceedings of the Human Factors Society Annual Meeting*, *31*, 1388–1392.

Endsley, M. R., & Kiris, E. O. (1995). The Out-of-the-Loop Performance Problem and Level of Control in Automation. *Human Factors*, *37*(2), 381-394. https://doi.org/10.1518/001872095779064555

Fereidunian, A., Lucas, C., Lesani, H., Lehtonen, M., & Nordman, M. (2007). Challenges in implementation of human-automation interaction models. *2007 Mediterranean Conference on Control Automation*, 1-6. https://doi.org/10.1109/MED.2007.4433895

Fitts, P. M. (1954). *The information capacity of the human motor system in controlling the amplitude of movement. - PsycNET*. /doiLanding?doi=10.1037%2Fh0055392

Fitts, Paul M. (1951). *Human engineering for an effective air-navigation and traffic-control system*. National Research Council, Div. of.

Flemisch, F., Adams, C. A., Conway, S., Goodrich, K. H., Palmer, M., & Schutte, P. (2005). *The H-Metaphor as a Guideline for Vehicle Automation and Interaction*.

Flemisch, Frank, Heesen, M., Hesse, T., Kelsch, J., Schieben, A., & Beller, J. (2012). Towards a dynamic balance between humans and automation : Authority, ability, responsibility and control in shared and cooperative control situations. *Cognition, Technology & Work*, *14*(1), 3-18. https://doi.org/10.1007/s10111-011-0191-6

Giese, M., Mistrzyk, T., Pfau, A., Szwillus, G., & von Detten, M. (2008). AMBOSS : A Task Modeling Approach for Safety-Critical Systems. In P. Forbrig & F. Paternò (Éds.), *Engineering Interactive Systems* (p. 98-109). Springer. https://doi.org/10.1007/978-3-540-85992-5_8

Grudin, J. (1994). Computer-supported cooperative work : History and focus. *Computer*, *27*(5), 19-26. https://doi.org/10.1109/2.291294

Guerrero Garcia, J., Vanderdonckt, J., & Gonzalez Calleros, J. M. (2008). FlowiXML : A step towards designing workflow management systems. *International Journal of Web Engineering and Technology*, *4*(2), 163-182. https://doi.org/10.1504/IJWET.2008.018096

Hancock, P. A., & Chignell, M. H. (1987). 8. Adaptive Control in Human-Machine Systems. In Peter A. Hancock (Éd.), *Advances in Psychology* (Vol. 47, p. 305-345). North-Holland. https://doi.org/10.1016/S0166-4115(08)62312-2

Hancock, P. A., & Scallen, S. F. (1996). The Future of Function Allocation. *Ergonomics in Design: The Quarterly of Human Factors Applications*, *4*(4), 24-29. https://doi.org/10.1177/106480469600400406

Harrison, M., Johnson, P., & Wright, P. (2003). Relating the Automation of Functions in Multiagent Control Systems to a System Engineering Representation. In E. Hollnagel (Éd.), *Handbook of Cognitive Task Design* (Vol. 20031153, p. 503-524). CRC Press. https://doi.org/10.1201/9781410607775.ch21

Hauser, J. R., Urban, G. L., Liberali, G., & Braun, M. (2009). Website Morphing. *Marketing Science*, *28*(2), 202-223. https://doi.org/10.1287/mksc.1080.0459

Hick, W. E. (1952). On the rate of gain of information. *Quarterly Journal of Experimental Psychology*, *4*(1), 11–26.

Hix, D., & Hartson, H. R. (1993). *Developing user interfaces : Ensuring usability through product and process*. Wiley.

Hoff, K. A., & Bashir, M. (2015). Trust in Automation : Integrating Empirical Evidence on Factors That Influence Trust. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *57*(3), 407-434. https://doi.org/10.1177/0018720814547570

Hollnagel, E. (2006). Task Analysis : Why, What, and How. In *Handbook of Human Factors and Ergonomics* (p. 371-383). John Wiley & Sons, Ltd. https://doi.org/10.1002/0470048204.ch14

International Standard Organization. (2019). *ISO 9241-210:2019(fr), Ergonomie de l'interaction homme-système—Partie 210 : Conception centrée sur l'opérateur humain pour les systèmes interactifs*. https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-2:v1:fr

ISO 8807. (1989). *Systèmes de traitement de l'information—Interconnexion de systèmes ouverts—LOTOS — Technique de description formelle basée sur l'organisation temporelle de comportement observationnel*. ISO. https://www.iso.org/cms/render/live/fr/sites/isoorg/contents/data/standard/01/62/16258.html

Jansson, A., Stensson, P., Bodin, I., Axelsson, A., & Tschirner, S. (2014). Authority and Level of Automation. *Human-Computer Interaction. Applications and Services*, 413-424. https://doi.org/10.1007/978-3-319-07227-2_39

Johnson, C. D., Miller, M. E., Rusnock, C. F., & Jacques, D. R. (2017). A framework for understanding automation in terms of levels of human control abstraction. *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 1145-1150. https://doi.org/10.1109/SMC.2017.8122766

Johnson, P., Drake, K., & Wilson, S. (1991). A Framework for Integrating UIMS and User Task Models in the Design of User Interfaces. In D. A. Duce, M. R. Gomes, F. R. A. Hopgood, & J. R. Lee (Éds.), *User Interface Management and Design* (p. 203-216). Springer. https://doi.org/10.1007/978-3-642-76283-3_20

Johnson, P., & Johnson, H. (1989). Knowledge analysis of tasks : Task analysis and specification for human-computer systems. *Engineering the human-computer interface*, 119–144.

Jourde, F., Laurillau, Y., & Nigay, L. (2010). COMM notation for specifying collaborative and multimodal interactive systems. *Proceedings of the 2nd ACM SIGCHI symposium on Engineering interactive computing systems*, 125–134. https://doi.org/10.1145/1822018.1822038

Kaber, D. B., & Endsley, M. R. (2004). The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theoretical Issues in Ergonomics Science*, *5*(2), 113-153. https://doi.org/10.1080/1463922021000054335

Kirwan, B., & Ainsworth, L. K. (1992). *A Guide To Task Analysis : The Task Analysis Working Group*. CRC Press.

Klien, G., Woods, D. D., Bradshaw, J. M., Hoffman, R. R., & Feltovich, P. J. (2004). Ten challenges for making automation a « team player » in joint human-agent activity. *IEEE Intelligent Systems*, *19*(6), 91-95. https://doi.org/10.1109/MIS.2004.74

Kortenkamp, D., Bonasso, R. P., Ryan, D., & Schreckenghost, D. (1997). *Traded Control with Autonomous Robots as Mixed Initiative Interaction*. 6.

Krasner, G. E., & Pope, S. T. (1988). *A Description of the Model-View-Controller User Interface Paradigm in the Smalltalk-80 System*. 35.

Ladry, J., Palanque, P., Barboni, E., & Navarre, D. (2010). Model-based usability evaluation and analysis of interactive techniques. *5 th International Workshop on Model Driven Development of Advanced User Interfaces (MDDAUI 2010)*, 21.

Laprie, J.-C. (1996). *Guide de la Sûreté de Fonctionnement, Laprie Jean-Claude , Editions Cépaduès*. https://www.cepadues.com/livres/guide-surete-fonctionnement-2854283821.html

Lee, D. W., & Bass, E. J. (2015). Simulating Task Sharing with Delegation for Autonomy and Authority in Air Traffic Control. *2015 IEEE International Conference on Systems, Man, and Cybernetics*, 962-968. https://doi.org/10.1109/SMC.2015.175

Lee, J. D., & See, K. A. (2004). Trust in automation : Designing for appropriate reliance. *Human factors*, *46*(1), 50–80.

Lee, J., & Moray, N. (1992). Trust, control strategies and allocation of function in human-machine systems. *Ergonomics*, *35*(10), 1243–1270.

Lim, Y.-K. (2004). Multiple aspect based task analysis (MABTA) for user requirements gathering in highly-contextualized interactive system design. *Proceedings of the 3rd Annual Conference on Task Models and Diagrams - TAMODIA '04*, 7. https://doi.org/10.1145/1045446.1045451

Limbourg, Q., & Vanderdonckt, J. (2003). Comparing task models for user interface design. *The handbook of task analysis for human-computer interaction*, *6*, 135–154.

Lorenz, B., Di Nocera, F., Röttger, S., & Parasuraman, R. (2001). The Effects of Level of Automation on the Out-of-the-Loop Unfamiliarity in a Complex Dynamic Fault-Management Task during Simulated Spaceflight Operations. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *45*(2), 44-48. https://doi.org/10.1177/154193120104500209

Lu, Z., Happee, R., Cabrall, C. D. D., Kyriakidis, M., & de Winter, J. C. F. (2016). Human factors of transitions in automated driving : A general framework and literature survey. *Transportation Research Part F: Traffic Psychology and Behaviour*, *43*, 183-198. https://doi.org/10.1016/j.trf.2016.10.007

Malone, T. W., & Crowston, K. (1990). What is coordination theory and how can it help design cooperative work systems? *Proceedings of the 1990 ACM conference on Computer-supported cooperative work*, 357–370. https://doi.org/10.1145/99332.99367

Martinie, C., Palanque, P., Ragosta, M., & Fahssi, R. (2013). Extending procedural task models by systematic explicit integration of objects, knowledge and information. *Proceedings of the 31st European Conference on Cognitive Ergonomics*, 1–10. https://doi.org/10.1145/2501907.2501954

Martinie, C., Palanque, P., & Winckler, M. (2011). Structuring and Composition Mechanisms to Address Scalability Issues in Task Models. In P. Campos, N. Graham, J. Jorge, N. Nunes, P. Palanque, & M. Winckler (Éds.), *Human-Computer Interaction – INTERACT 2011* (p. 589-609). Springer Berlin Heidelberg.

Martinie De Almeida, C. (2011). *Une approche à base de modèles synergiques pour la prise en compte simultanée de l'utilisabilité, la fiabilité et l'opérabilité des systèmes interactifs critiques* [These de doctorat, Toulouse 3]. http://www.theses.fr/2011TOU30254

McCall, Rod, McGee, F., Mirnig, A., Meschtscherjakov, A., Louveton, N., Engel, T., & Tscheligi, M. (2019). A taxonomy of autonomous vehicle handover situations. *Transportation Research Part A: Policy and Practice*, *124*, 507-522. https://doi.org/10.1016/j.tra.2018.05.005

McCall, Roderick, McGee, F., Meschtscherjakov, A., Louveton, N., & Engel, T. (2016). Towards A Taxonomy of Autonomous Vehicle Handover Situations. *Proceedings of the 8th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, 193–200. https://doi.org/10.1145/3003715.3005456

Mercado, J. E., Rupp, M. A., Chen, J. Y. C., Barnes, M. J., Barber, D., & Procci, K. (2016). Intelligent Agent Transparency in Human–Agent Teaming for Multi-UxV

Management. *Human Factors*, *58*(3), 401-415. https://doi.org/10.1177/0018720815621206

Mertes, F., & Jenney, L. (1974). *Automation Applications in an Advanced Air Traffic Management System : Volume 3. Methodology for Man-Machine Task Allocation*. United States. Dept. of Transportation. Office of the Secretary.

Milgram, P., Rastogi, A., & Grodski, J. J. (1995). Telerobotic control using augmented reality. *Proceedings 4th IEEE International Workshop on Robot and Human Communication*, 21-29. https://doi.org/10.1109/ROMAN.1995.531930

Miller, C. A., & Parasuraman, R. (2007). Designing for Flexible Interaction Between Humans and Automation : Delegation Interfaces for Supervisory Control. *Human Factors*, *49*(1), 57-75. https://doi.org/10.1518/001872007779598037

Mori, G., Paterno, F., & Santoro, C. (2002). CTTE : Support for developing and analyzing task models for interactive system design. *IEEE Transactions on Software Engineering*, *28*(8), 797-813. https://doi.org/10.1109/TSE.2002.1027801

Norman, D. (2013). *The design of everyday things : Revised and expanded edition*. Basic books.

Ntuen, C. A., & Park, E. H. (1988). Human Factor Issues in Teleoperated Systems. *Proceedings of the First International Conference on Ergonomics of Hybrid Automated Systems I*, 203–210. http://dl.acm.org/citation.cfm?id=58209.58235

Palanque, P., & Bastide, R. (1994). *A Formalism for Reliable User Interfaces*.

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, *30*(3), 286-297. https://doi.org/10.1109/3468.844354

Parasuraman, Raja. (2000). Designing automation for human use : Empirical studies and quantitative models. *Ergonomics*, *43*(7), 931-951. https://doi.org/10.1080/001401300409125

Parasuraman, Raja, Mouloua, M., & Molloy, R. (1996). Effects of Adaptive Task Allocation on Monitoring of Automated Systems. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, *38*(4), 665-679. https://doi.org/10.1518/001872096778827279

Paternò, F. (2004). ConcurTaskTrees : An engineered notation for task models. *The handbook of task analysis for human-computer interaction*, 483–503.

Paterno, F., Mancini, C., & Meniconi, S. (1997). ConcurTaskTrees : A Diagrammatic Notation for Specifying Task Models. In S. Howard, J. Hammond, & G. Lindgaard (Éds.), *Human-Computer Interaction INTERACT '97* (p. 362-369). Springer US. http://link.springer.com/chapter/10.1007/978-0-387-35175-9_58

Paternò, F., Santoro, C., Spano, L. D., & Raggett, D. (2014). *W3C MBUI - Task Models*. https://www.w3.org/TR/task-models/

Penichet, V. M. R., Marin, I., Gallud, J. A., Lozano, M. D., & Tesoriero, R. (2007). A Classification Method for CSCW Systems. *Electronic Notes in Theoretical Computer Science*, *168*, 237-247. https://doi.org/10.1016/j.entcs.2006.12.007

Penichet, Victor M. R., Lozano, M. D., & Gallud, J. A. (2009). An Ontology to Model Collaborative Organizational Structures in CSCW Systems. In M. Redondo, C. Bravo, & M. Ortega (Éds.), *Engineering the User Interface* (p. 1-14). Springer London. https://doi.org/10.1007/978-1-84800-136-7_10

Penichet, Victor M. R., Lozano, M. D., Gallud, J. A., & Tesoriero, R. (2009). User interface analysis for groupware applications in the TOUCHE process model. *Advances in*

Engineering Software, *40*(12), 1212-1222. https://doi.org/10.1016/j.advengsoft.2009.01.026

Pfaff, G. E. (1985). *User interface management systems* (Vol. 1). Springer.

Pinelle, D., Gutwin, C., & Greenberg, S. (2003). Task analysis for groupware usability evaluation : Modeling shared-workspace tasks with the mechanics of collaboration. *ACM Transactions on Computer-Human Interaction*, *10*(4), 281–311. https://doi.org/10.1145/966930.966932

Pnueli, A. (1986). Applications of temporal logic to the specification and verification of reactive systems : A survey of current trends. In *Current trends in Concurrency* (p. 510–584). Springer.

Potel, M. (1996). *MVP: Model-View-Presenter The Taligent Programming Model for C++ and Java*. 14.

Price, H. E., Maisano, R. E., & Van Cott, H. P. (1982). *Allocatin of functions in man-machine systems : A perspective and literature review*. Oak Ridge National Lab.

Proud, R. W., Hart, J. J., & Mrozinski, R. B. (2003). *Methods for Determining the Level of Autonomy to Design into a Human Spaceflight Vehicle : A Function Specific Approach*. NATIONAL AERONAUTICS AND SPACE ADMINISTRATION HOUSTON TX LYNDON B JOHNSON SPACE CENTER, NATIONAL AERONAUTICS AND SPACE ADMINISTRATION HOUSTON TX LYNDON B JOHNSON SPACE CENTER. http://www.dtic.mil/docs/citations/ADA515467

Ragosta, M. (2015). *A multi-modelS based approach for the modelling and the analysis of usable and resilient partly autonomous interactive systems* [Phd, Université de Toulouse, Université Toulouse III - Paul Sabatier]. http://thesesups.ups-tlse.fr/2813/

Rasmussen, J. (1983). Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models. *IEEE Transactions on Systems, Man, and Cybernetics, SMC-13*(3), 257-266. https://doi.org/10.1109/TSMC.1983.6313160

Redmill, F., & Rajan, J. (1996). *Human factors in safety-critical systems*. Butterworth-Heinemann.

Rich, C. (2009). Building Task-Based User Interfaces with ANSI/CEA-2018. *Computer, 42*(8), 20-27. https://doi.org/10.1109/MC.2009.247

Riley, V. (1989). A General Model of Mixed-Initiative Human-Machine Systems. *Proceedings of the Human Factors Society Annual Meeting*, *33*(2), 124-128. https://doi.org/10.1177/154193128903300227

Roschelle, J., & Teasley, S. D. (1995). The Construction of Shared Knowledge in Collaborative Problem Solving. In C. O'Malley (Éd.), *Computer Supported Collaborative Learning* (p. 69-97). Springer. https://doi.org/10.1007/978-3-642-85098-1_5

Rouse, W. B. (1988). Adaptive Aiding for Human/Computer Control: *Human Factors*. https://doi.org/10.1177/001872088803000405

RTCA, & EUROCAE. (2011). *"DO-178C/ED-12C." Software Considerations in Airborne Systems and Equipment Certification*.

SAE J3016. (2019). *Levels of Driving Automation*. https://www.sae.org/site/news/2019/01/sae-updates-j3016-automated-driving-graphic

Salber, D., Coutaz, J., Decouchant, D., & Riveill, M. (1995). *De l'observabilité et de l'honnêteté : Le cas du contrôle d'accès dans la Communication Homme-Homme Médiatisée*. 7.

Sarter, N. B., & Woods, D. D. (1997). Team Play with a Powerful and Independent Agent : Operational Experiences and Automation Surprises on the Airbus A-320. *Human Factors, 39*(4), 553-569. https://doi.org/10.1518/001872097778667997

Sarter, N. B., Woods, D. D., & Billings, C. E. (1997). Automation surprises. *Handbook of human factors and ergonomics*, *2*, 1926–1943.

Scapin, D., & Pierret-Golbreich, C. (1989). Towards a method for task description : MAD. *Work with display units*, *89*, 371–380.

Semple, W. G. (1998). *Information, Decision or Action ? - The Role of IT in Fast Jet Mission Systems*. 10.

Shepherd, A. (1985). Hierarchical Task Analysis and Training Decisions. *PLET: Programmed Learning & Educational Technology*, *22*(2), 162-176. https://doi.org/10.1080/1355800850220209

Shepherd, A. (2002). *HTA as a framework for task analysis*. /paper/HTA-as-a-framework-for-task-analysis-a-HEPHERD/efb4ae7a6318ca6ba7b5864e7485b435ea9d49a9

Sheridan, T. B. (1988). Trustworthiness of Command and Control Systems. *IFAC Proceedings Volumes*, *21*(5), 427-431. https://doi.org/10.1016/S1474-6670(17)53945-2

Sheridan, T. B. (2011). Adaptive Automation, Level of Automation, Allocation Authority, Supervisory Control, and Adaptive Control : Distinctions and Modes of Adaptation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, *41*(4), 662-667. https://doi.org/10.1109/TSMCA.2010.2093888

Sheridan, T.B. (2000). Function allocation : Algorithm, alchemy or apostasy? *International Journal of Human-Computer Studies*, *52*(2), 203-216. https://doi.org/10.1006/ijhc.1999.0285

Sheridan, Thomas B. (1992). *Telerobotics, Automation, and Human Supervisory Control*. MIT Press.

Sheridan, Thomas B., & Verplank, W. L. (1978). *Human and Computer Control of Undersea Teleoperators*. MASSACHUSETTS INST OF TECH CAMBRIDGE MAN-

MACHINE SYSTEMS LAB, MASSACHUSETTS INST OF TECH CAMBRIDGE MAN-MACHINE SYSTEMS LAB. http://www.dtic.mil/docs/citations/ADA057655

Shoval, S., Koren, Y., & Borenstein, J. (1993). Optimal task allocation in task-agent-control state space. *Proceedings of IEEE Systems Man and Cybernetics Conference - SMC*, *4*, 27-32 vol.4. https://doi.org/10.1109/ICSMC.1993.390678

Sire, S. (2000). *La collaboration directe d'un paradigme d'interaction pour le travail assisté par ordinateur* [These de doctorat, Toulouse 1]. http://www.theses.fr/2000TOU10005

Sommerville, I. (2011). Software engineering 9th Edition. *ISBN-10*, *137035152*, 18.

Sørensen, E., & Mihailesc, M. I. (2010). *Model-View-ViewModel (MVVM) Design Pattern using Windows Presentation Foundation (WPF) Technology*. 19.

Swain, A. D. (1983). *Handbook of Human Reliability Analysis with Emphasis on Nuclear Power Plant Applications : Final Report*. U. S. Nuclear Regulatory Commission.

Tarby, J.-C., & Barthet, M.-F. (1996). The DIANE+ Method. *CADUI*, *96*, 95–119.

Vagia, M., Transeth, A. A., & Fjerdingen, S. A. (2016). A literature review on the levels of automation during the years. What are the different taxonomies that have been proposed? *Applied Ergonomics*, *53*, 190-202. https://doi.org/10.1016/j.apergo.2015.09.013

van der Veer, G. C., Lenting, B. F., & Bergevoet, B. A. J. (1996). GTA : Groupware task analysis — Modeling complexity. *Acta Psychologica*, *91*(3), 297-322. https://doi.org/10.1016/0001-6918(95)00065-8

Vicente, K. J. (1999). *Cognitive Work Analysis : Toward Safe, Productive, and Healthy Computer-Based Work*. CRC Press.

Villaren, T., Coppin, G., & Leal, A. (2012). Modeling task transitions to help designing for better situation awareness. *Proceedings of the 4th ACM SIGCHI symposium on*

Engineering interactive computing systems, 195–204. https://doi.org/10.1145/2305484.2305517

Vives-Boix, V., Ruiz-Fernández, D., de Ramón-Fernández, A., Marcos-Jorquera, D., & Gilart-Iglesias, V. (2017). A Knowledge-Based Clinical Decision Support System for Monitoring Chronic Patients. *International Work-Conference on the Interplay Between Natural and Artificial Computation*, 435–443.

Walch, M., Lange, K., Baumann, M., & Weber, M. (2015). Autonomous Driving : Investigating the Feasibility of Car-driver Handover Assistance. *Proceedings of the 7th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, 11–18. https://doi.org/10.1145/2799250.2799268

Winckler, M., Bach, C., & Bernhaupt, R. (2013). Identifying user experience dimensions for mobile incident reporting in urban contexts. *IEEE Transactions on Professional Communication*, *56*(2), 97–119.

Wintersberger, P., Green, P., & Riener, A. (2017). Am I Driving or Are You or Are We Both? A Taxonomy for Handover and Handback in Automated Driving. *Proceedings of the 9th International Driving Symposium on Human Factors in Driver Assessment, Training, and Vehicle Design: Driving Assessment 2017*, 333-339. https://doi.org/10.17077/drivingassessment.1655

Woods, D. D. (1996). Decomposing automation : Apparent simplicity, real complexity. *Automation and human performance: Theory and applications*, 3–17.

Wurdel, M., Sinnig, D., & Forbrig, P. (2008). *CTML: Domain and Task Modeling for Collaborative Environments*. 15.

Young, M. S., Stanton, N. A., & Harris, D. (2007). Driving automation : Learning from aviation about design philosophies. *International Journal of Vehicle Design*, *45*(3), 323. https://doi.org/10.1504/IJVD.2007.014908

Résumé

L'automatisation permet à certains systèmes d'exécuter des fonctions sans contrôle extérieur et d'adapter les fonctions qu'ils exécutent à de nouveaux contextes ou buts. Les systèmes contenant de l'automatisation sont de plus en plus utilisés, autant pour aider les humains dans les tâches du quotidien, avec, par exemple, le lave-vaisselle, que dans leur vie professionnelle, par exemple pour un pilote d'avion, le cockpit. Dans le domaine de l'aéronautique, l'automatisation a permis de progressivement réduire le nombre de 4 pilotes à 2 pilotes. L'automatisation a d'abord été considérée comme un moyen d'augmenter les performances et réduire l'effort en migrant des tâches précédemment allouées aux humains à des systèmes. Ceci, dans l'hypothèse où les systèmes seraient meilleurs que les humains pour réaliser certaines tâches et vice-versa. Le MABA-MABA (Machine Are Better At – Man Are Better At) de Paul Fitts est une proposition de méthode d'allocation de tâches et de fonctions reprenant cette hypothèse. Dans la continuité de cette hypothèse, différentes descriptions de niveaux d'automatisation ont été proposées. Les 10 niveaux d'automatisation de Parasuraman, Sheridan et Wickens décrivent différentes allocations de tâches et de fonctions entre l'humain et le système. Plus le niveau d'automatisation augmente, plus le nombre de tâches migrées de l'humain vers le système est important. Ces approches ont fait l'objet de critiques, notamment à travers « *MABA-MABA or Abracadabra? Progress on Human–Automation Coordination* » de Dekker et Woods qui soulignent que l'introduction de l'automatisation induit de nouvelles tâches allouées à l'humain pour gérer cette automatisation. Ils rappellent aussi que ces approches masquent l'aspect coopératif du couple humain-système contenant de l'automatisation. Dans le but de caractériser cette coopération humain-système, il a été montré qu'il est important de considérer, lors de la phase de conception, l'allocation de l'autorité, de la responsabilité, du contrôle et de l'initiative de modifications de ces allocations lors de l'activité. Cependant dans les approches existantes, la conception de l'automatisation et de la coopération avec l'humain est considérée à haut niveau en amont dans le cycle de conception et de développement du système. Ces travaux ne fournissent pas de support pour raisonner sur l'allocation des ressources, des transitions de contrôle, de la responsabilité et de l'autorité tout au long du cycle de conception et de développement du système. L'objectif de cette thèse est de démontrer qu'il est possible d'analyser et de décrire à bas niveau les tâches et les fonctions ainsi que la coopération entre l'humain et le système contenant de l'automatisation. Cette analyse et cette description permet de caractériser les tâches, les fonctions et la coopération en terme d'autorité, de responsabilité, de partage de ressources et d'initiation de transition de contrôle. Cela, dans l'objectif de fournir un cadre conceptuel et un processus outillé à base de modèles pour l'analyse structurée et la compréhension de l'automatisation. Afin de montrer la faisabilité de cette approche, cette thèse présente les résultats de son application à une étude de cas industrielle dans le domaine d'application de l'aéronautique.

Mots-clés: automatisation, interaction Homme-Machine, modélisation de tâches, systèmes interactifs critiques

Abstract

Automation enables systems to execute some functions without outside control and to adapt functions they execute to new contexts and goals. Systems with automation are used more and more to help humans in everyday tasks with, for example, the dishwasher. Systems with automation are also used to help humans in their professional life. For example, in the field of aeronautics, automation has gradually reduced the number from 4 pilots to 2 pilots. Automation was first considered as a way to increase performance and reduce effort by migrating tasks previously allocated to humans to systems. This, in the hypothesis that systems would be better than humans would at performing certain tasks and vice-versa. Paul Fitts proposed MABA-MABA (Machine Are Better At – Man Are Better At), a tasks and functions allocation method based on this hypothesis. In line with this hypothesis, various descriptions of levels of automation have been proposed. The 10 levels of Automation (LoA) by Parasuraman, Sheridan et Wickens describe different tasks and functions allocations between the human and the system. The higher the level of automation, the more tasks migrate from human to system. These approaches have been the subject of criticism. « *MABA-MABA or Abracadabra? Progress on Human–Automation Coordination* » of Dekker and Woods highlights that automation leads to new tasks allocated to humans to manage this automation. Moreover, they recall that these approaches hide the cooperative aspect of the human-system couple. To characterize the human-system cooperation, the importance of considering, at design time, the allocation of authority, responsibility, control and the initiative to modify these allocations during the activity was demonstrated. However, the existing approaches describe a high-level design of automation and cooperation early in the design and development process. These approaches does not provide support for reasoning about the allocation of resources, control transitions, responsibility and authority throughout the design and development process. The purpose of this thesis is to demonstrate the possibility to analyze and describe at a low-level tasks and functions as well as the cooperation between humans and the system with automation. This analysis and this description enable to characterize tasks, functions and the cooperation in terms of authority, responsibility, resource sharing and control transition initiation. . The aim of this work is to provide a framework and a model-based and tool supported process to analyze and understand automation. In order to show the feasibility of this approach, this thesis presents the results of the application of the proposed process to an industrial case study in the field of aeronautics.

Keywords: automation, human-machine interaction, task modeling, interactive critical systems

Annexes

1 Modèles de tâches ségrégués et de fonctions pour la gestion des alarmes avec le FWS

1.1 Modèles de tâches ségrégués du PM

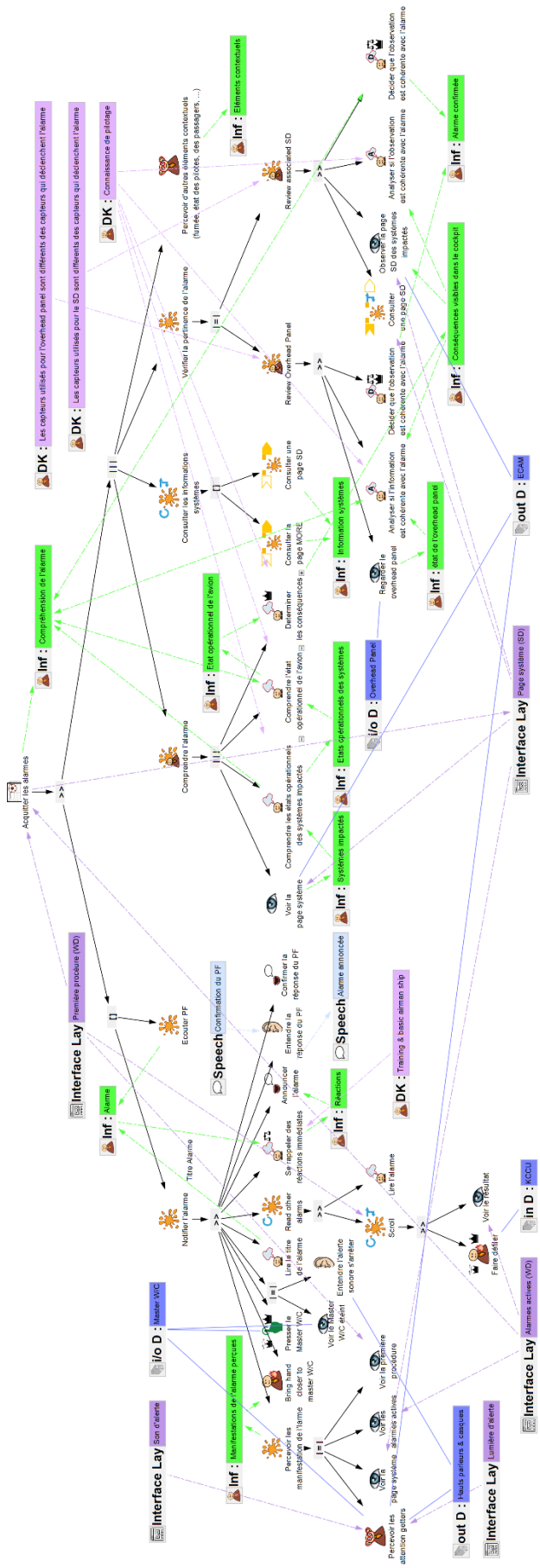


Figure 121. Modèle de tâches ségrégué du PM : Acquitter les alarmes

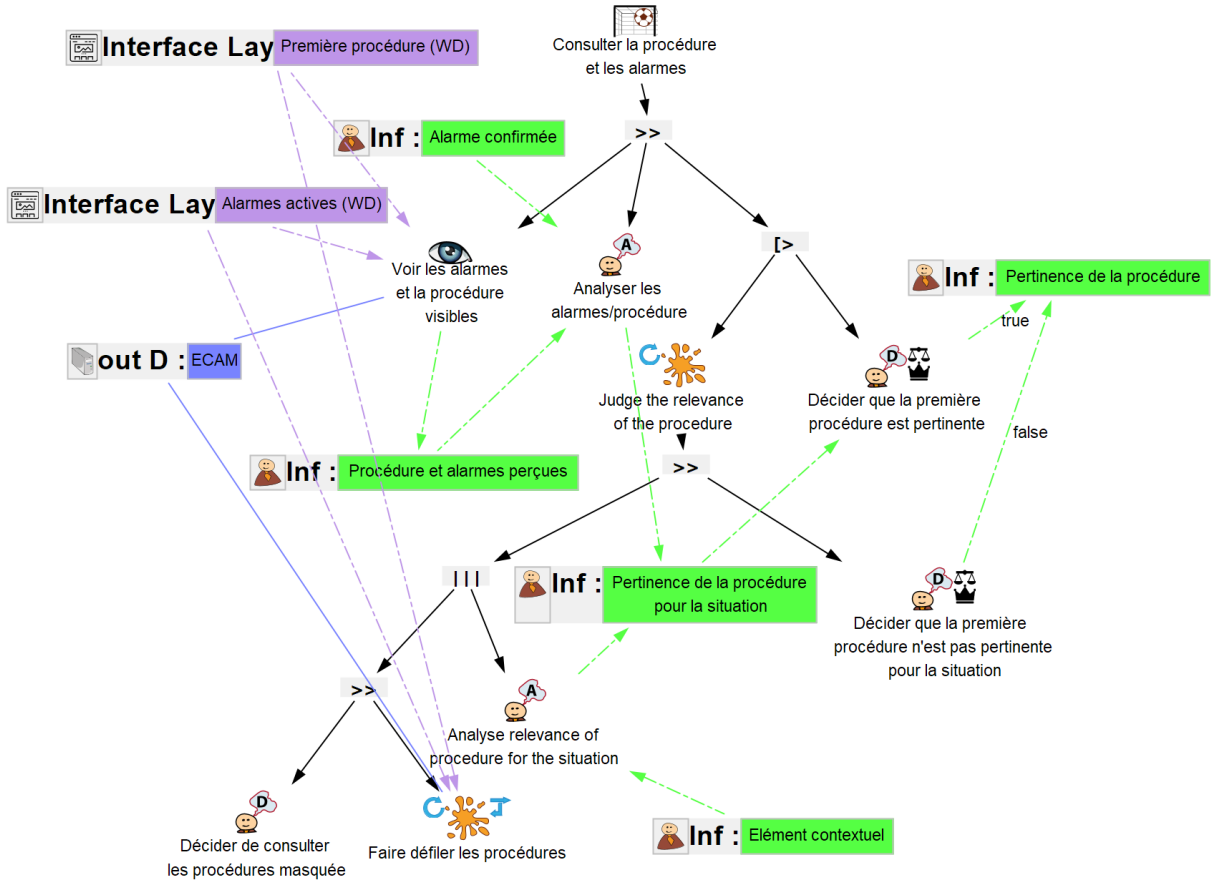


Figure 122. Modèle de tâches ségrégué du PM : Consulter la procédure et les alarmes

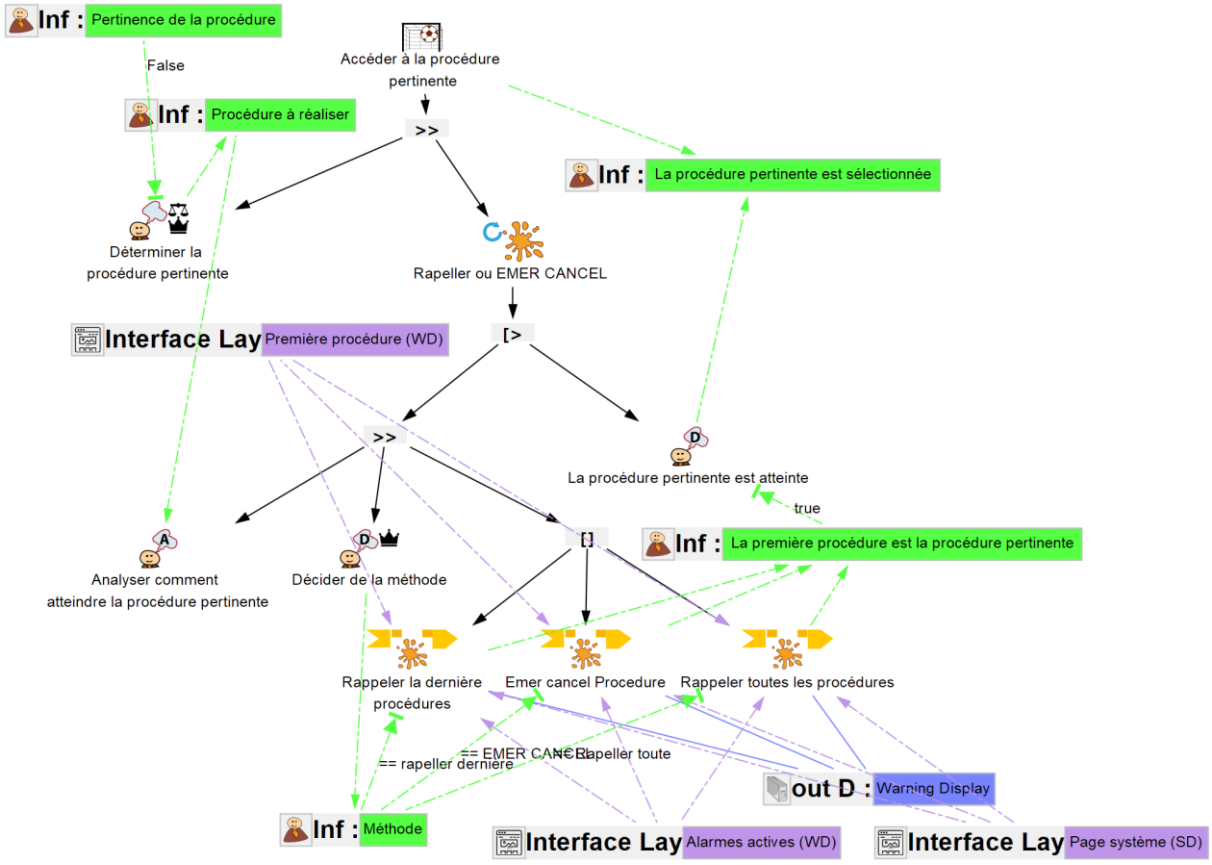


Figure 123. Modèle de tâches ségrégué du PM : Accéder à la procédure pertinente

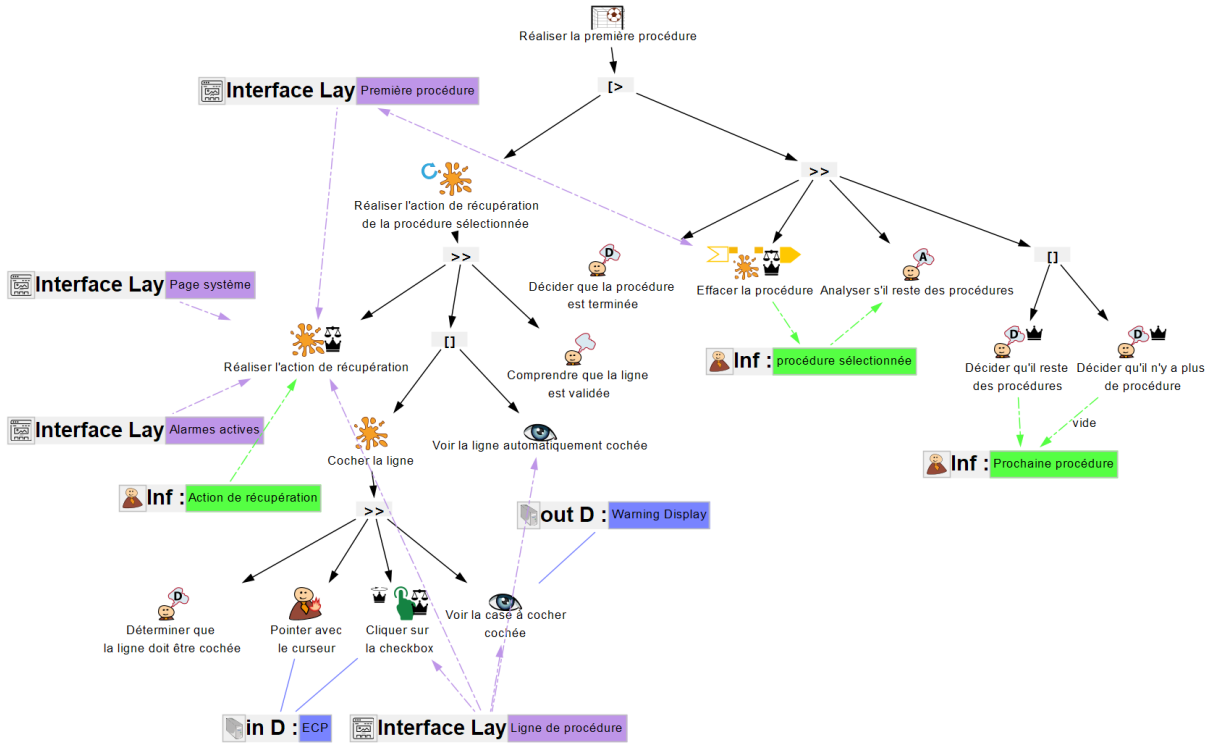


Figure 124. Modèle de tâches ségrégué du PM : Réaliser la première procédure.

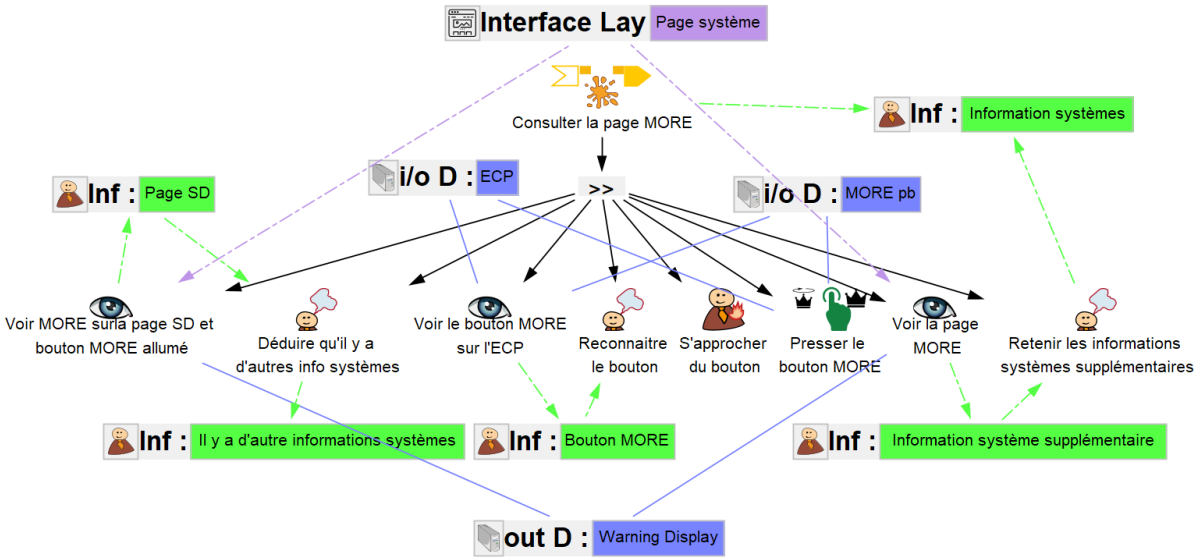


Figure 125. Modèle de tâches ségrégué du PM : Consulter la page MORE.

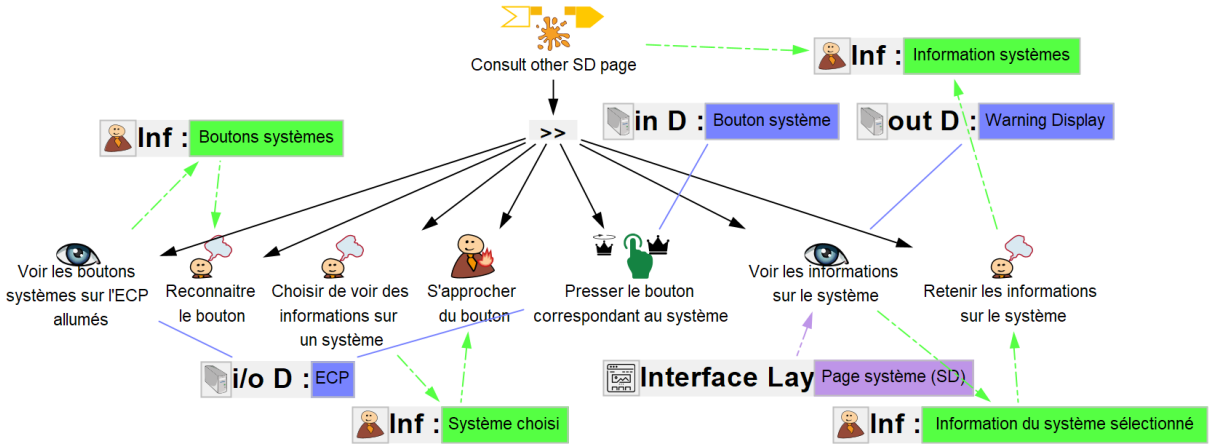


Figure 126. Modèle de tâches ségrégué du PM : Consulter une page SD

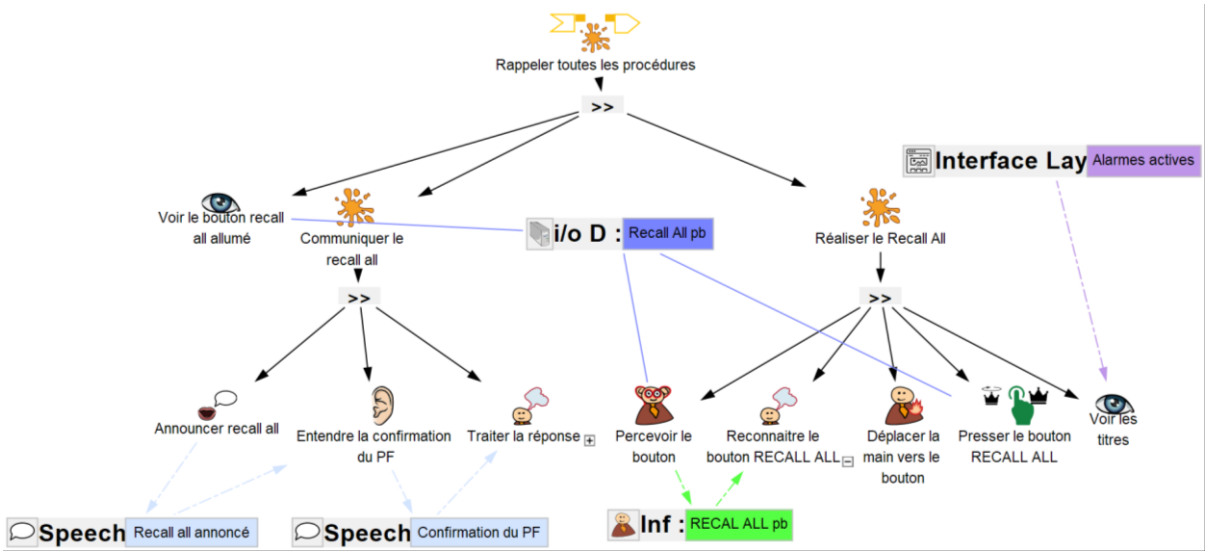


Figure 127. Modèle de tâches ségrégué du PM : Rappeler toutes (RECALL ALL) les procédures effacées

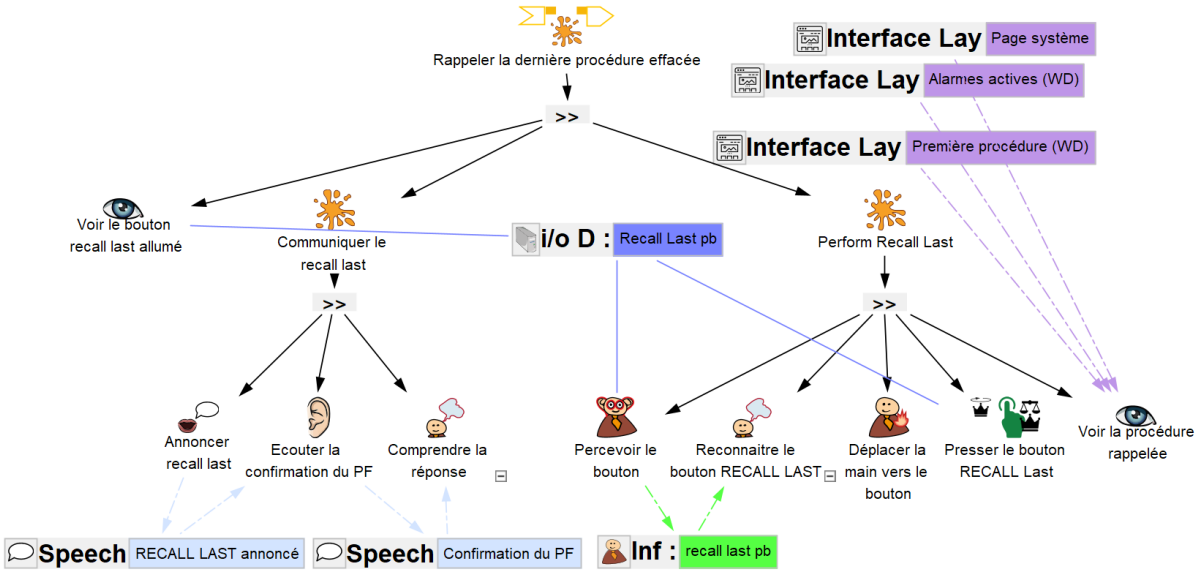


Figure 128. Modèle de tâches ségrégué du PM : Rappeler la dernière procédure effacée.

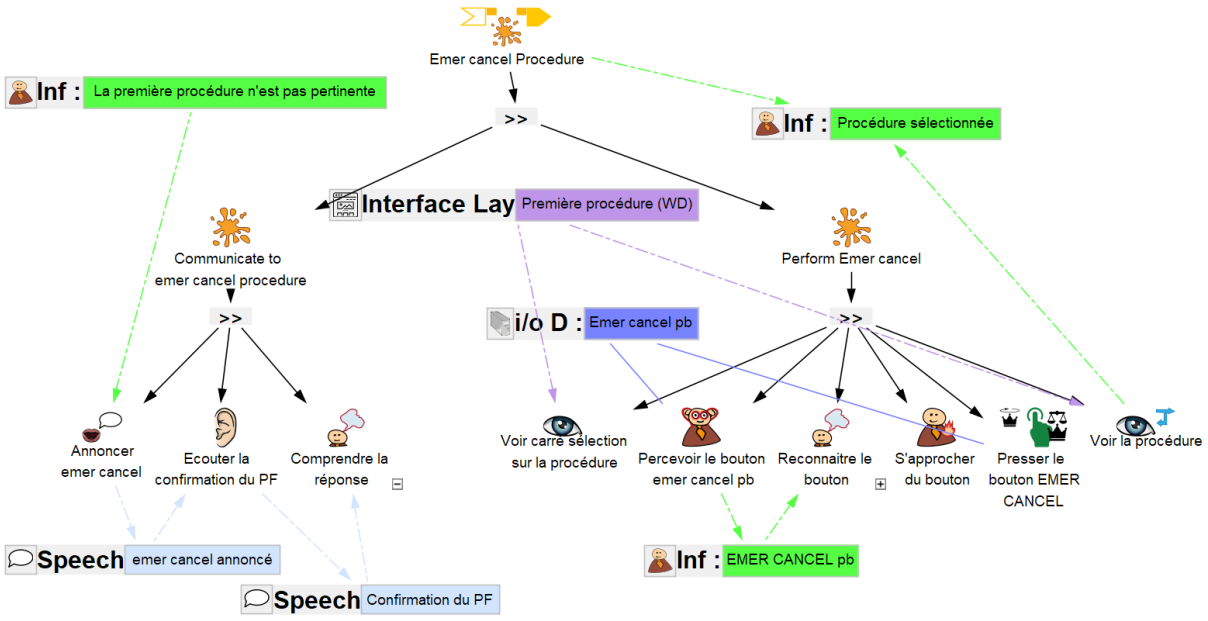


Figure 129. Modèle de tâches ségrégué du PM : EMER CANCEL (Supprimer) une procédure

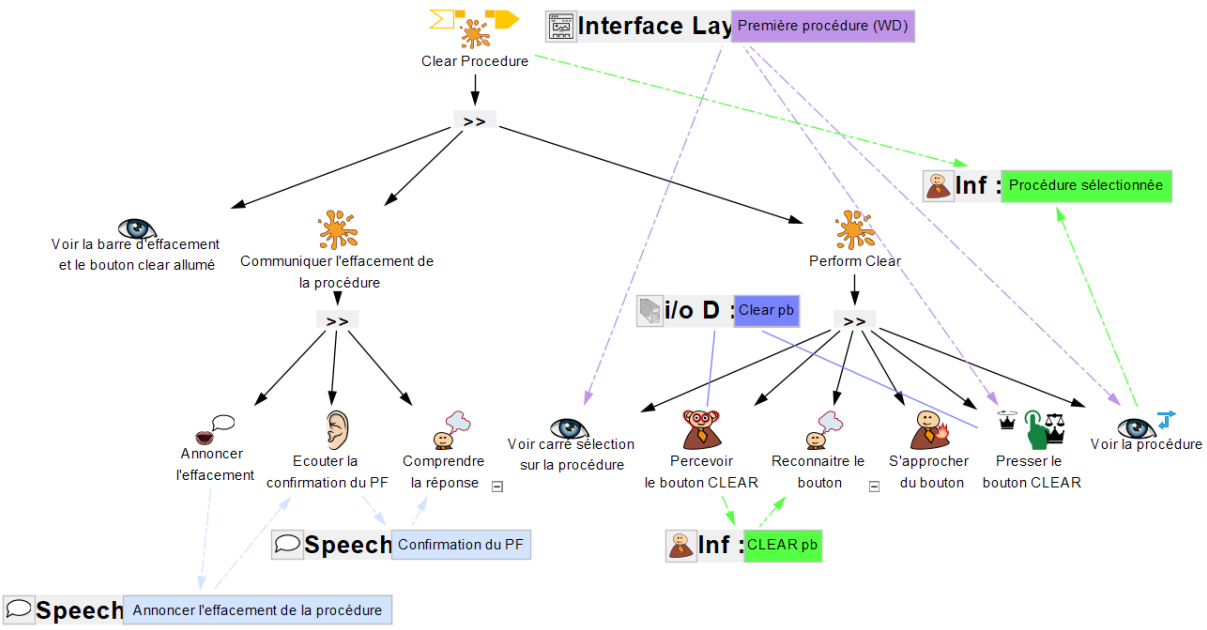


Figure 130. Modèle de tâches ségrégué du PM : CLEAR (effacer) une procédure terminée

1.2 Modèles de fonctions du FWS

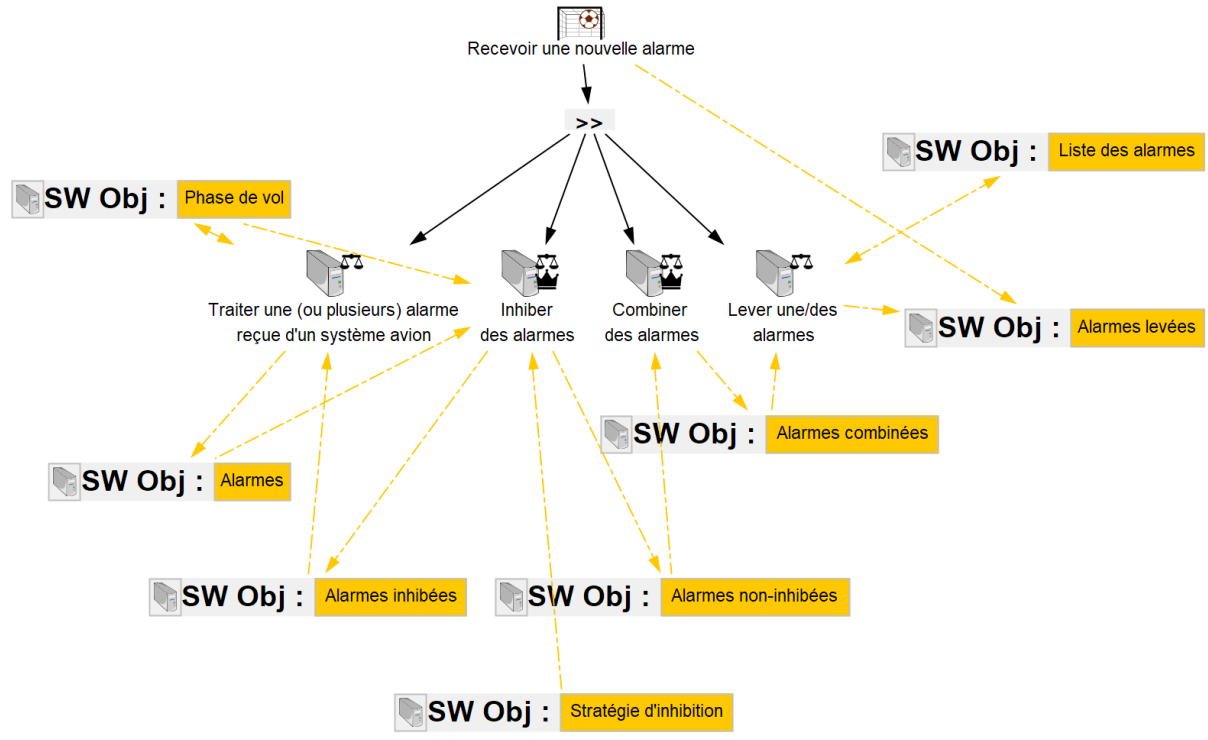


Figure 131. Modèle de fonctions du FWS : Recevoir une nouvelle alarme.

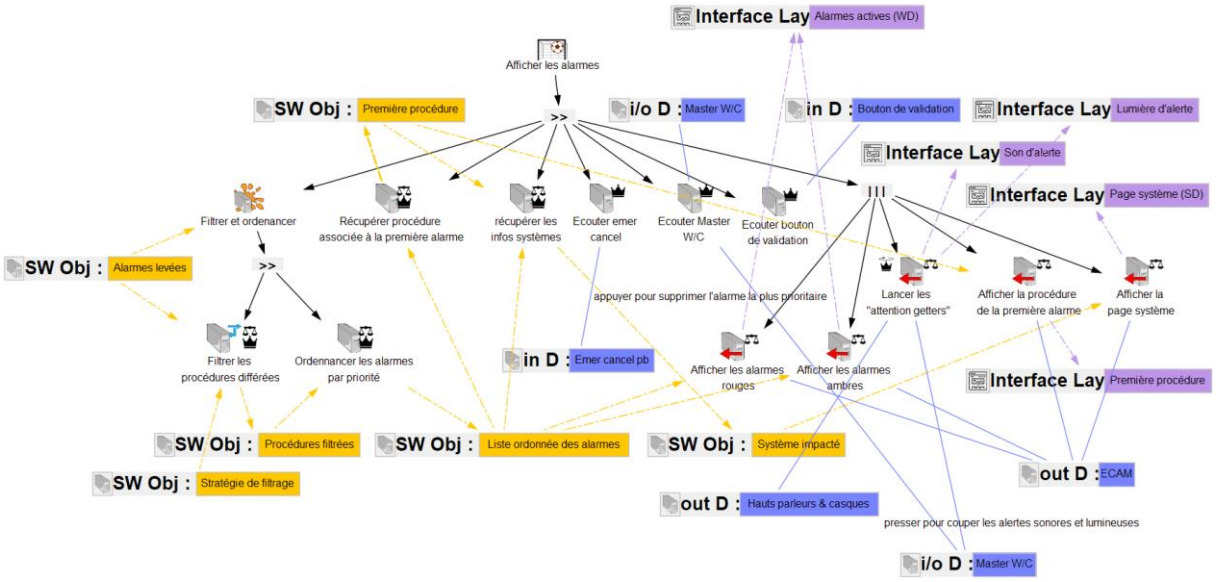


Figure 132. Modèle de fonctions du FWS : Afficher les alarmes.

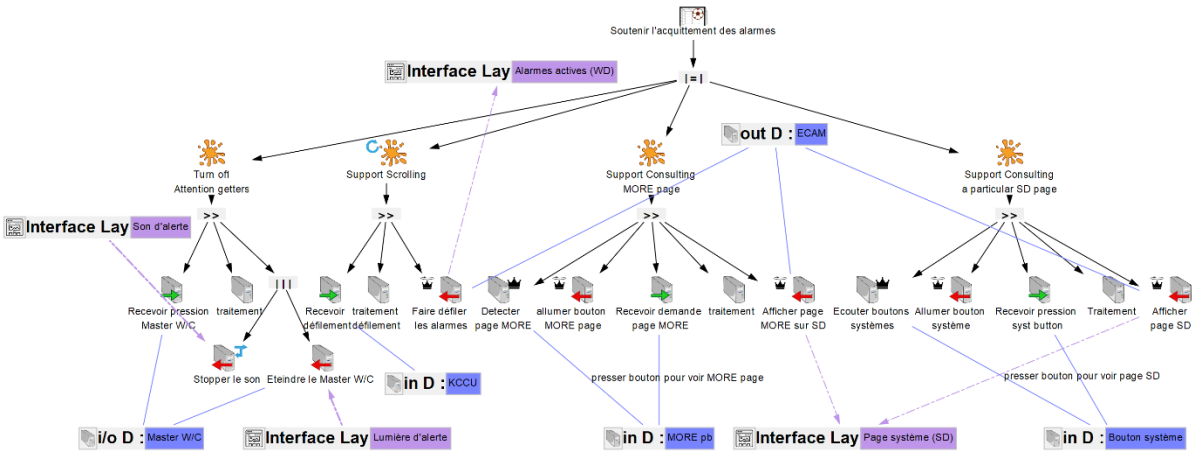


Figure 133. Modèle de fonctions du FWS : Soutenir l'acquittement des alarmes.

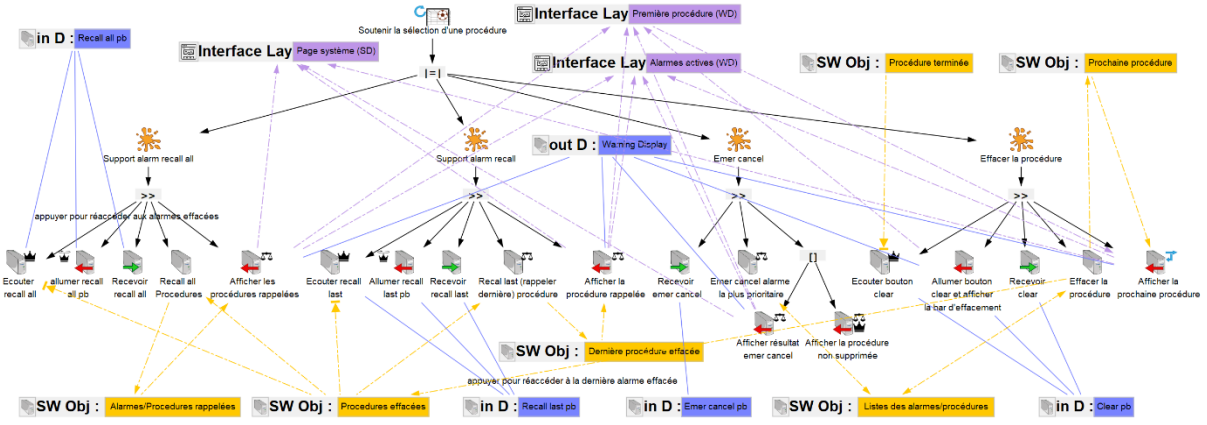


Figure 134. Modèle de fonctions du FWS : Soutenir la sélection d'une procédure (différente de la première procédure).

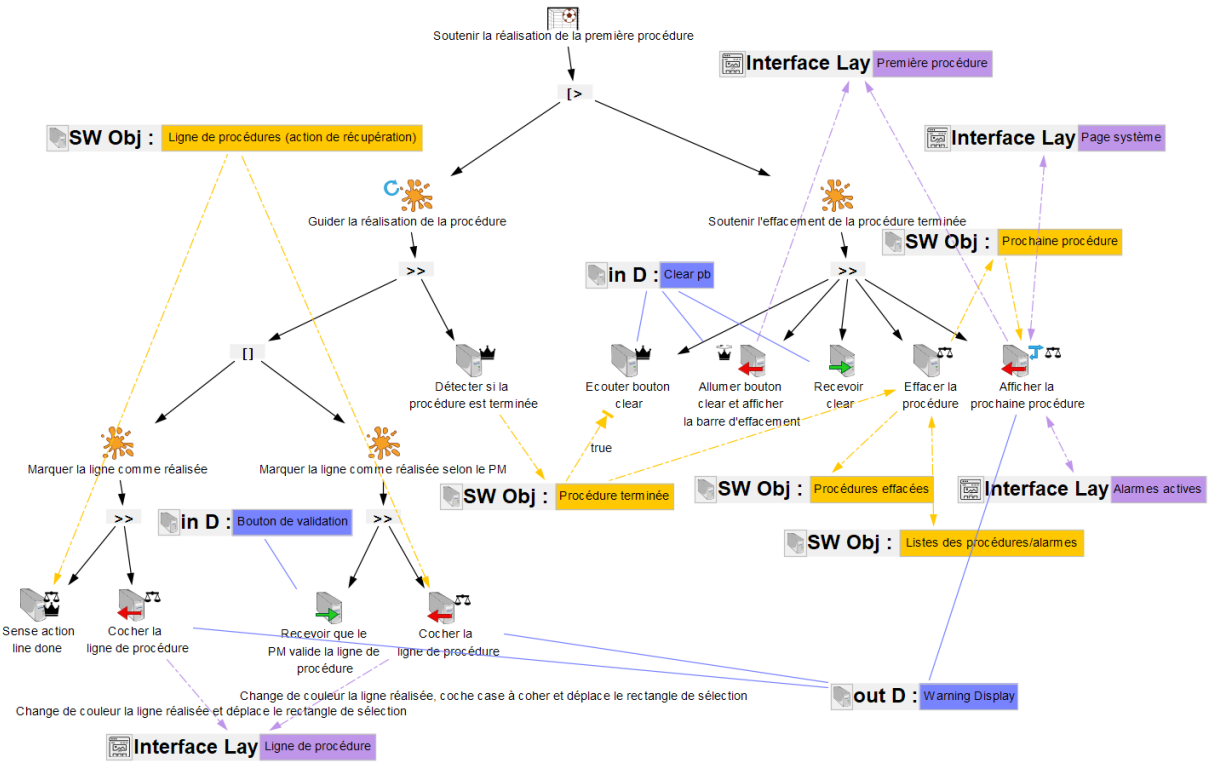


Figure 135. Modèle de fonctions du FWS : Soutenir la réalisation de la première procédure.

2 Colonnes du tableau RCRA décrivant l'allocation des ressources pour la gestion des alarmes avec le FWS

Tableau 46. Tableau décrivant l'allocation des ressources entre le FWS et le PM pour la résolution des alarmes.

Allocation des ressources				
Ressource [Partagée]	Description DOD	Type DOD	Possesseur	Acquéreur
Alarmes [P]	Lumière d'alerte	Couche interface	FWS	PM
	Son d'alerte	Couche interface	FWS	PM
	Hauts parleurs & casques	Périphérique d'entrée/sortie	FWS	PM
	Master W/C	Périphérique d'entrée/sortie	FWS	PM
	Alarmes (WD)	Couche interface	FWS	PM
	Manifestation de l'alarme perçue	Information	PM	
	Alarme	Information	PM	
	Compréhension de l'alarme	Information	PM	
	Alarme annoncée	Parole	PM	PF
	Confirmation du PF	Parole	PF	PM
	Alarme confirmée	Information	PM	
	Alarmes	Objet Logiciel	FWS	

	Alarmes inhibées	Objet Logiciel	FWS	
	Alarmes non-inhibées	Objet Logiciel	FWS	
	Alarmes combinées	Objet Logiciel	FWS	
	Liste des alarmes	Objet Logiciel	FWS	
	Alarmes actives	Objet Logiciel	FWS	
	Liste ordonnée des alarmes	Objet Logiciel	FWS	
Alarmes / Procédures [P]	Recall ALL pb	Périphérique d'entrée	FWS	PM
	Recall LAST pb	Périphérique d'entrée	FWS	PM
	Clear pb	Périphérique d'entrée	FWS	PM
	EMER CANCEL pb	Périphérique d'entrée	FWS	PM
	Procédure et alarmes perçues	Information	PM	
	Méthode	Information	PM	
	CLEAR pb	Information	PM	
	Annoncer l'effacement de la procédure	Parole	PM	PF
	Confirmation du PF	Parole	PF	PM
	EMER CANCEL annoncé	Parole	PF	PM
	Confirmation du PF	Parole	PF	PM
	EMER CANCEL pb	Information	PM	
	Recall all annoncé	Parole	PF	PM
	Confirmation du PF	Parole	PF	PM
	RECAL ALL pb	Information	PM	
	RECALL LAST annoncé	Parole	PF	PM
	Confirmation du PF	Parole	PF	PM
	RECAL LAST pb	Information	PM	
	Alarmes/Procédures rappelées	Objet Logiciel	FWS	
	Listes des alarmes/procédures	Objet Logiciel	FWS	
Procédures [P]	Première procédure	Couche interface	FWS	PM
	Pertinence de la procédure pour la situation	Information	PM	
	Pertinence de la procédure	Information	PM	
	Procédure à réaliser	Information	PM	
	La première procédure est la procédure pertinente	Information	PM	
	La procédure pertinente est sélectionnée	Information	PM	

	Procédure sélectionnée	Information	PM	
	Prochaine procédure	Information	PM	
	Procédures filtrées	Objet Logiciel	FWS	
	Première procédure	Objet Logiciel	FWS	
	Procédures effacées	Objet Logiciel	FWS	
	Dernière procédure effacée	Objet Logiciel	FWS	
	Procédure terminée	Objet Logiciel	FWS	
	Prochaine procédure	Objet Logiciel	FWS	
Actions de récupération [P]	Ligne de procédure	Couche interface	FWS	PM
	Bouton de validation	Périphérique d'entrée	FWS	PM
	Réactions	Information	PM	
	Action de récupération	Information	PM	
	Ligne de procédures (action de récupération)	Objet Logiciel	FWS	
Stratégie de filtrage	Stratégie de filtrage	Objet Logiciel	FWS	
	Stratégie d'inhibition	Objet Logiciel	FWS	
Stratégie de vérification des alarmes	Les capteurs utilisés pour l'overhead panel sont différents des capteurs qui déclenchent l'alarme	Connaissance	PM	
	Les capteurs utilisés pour le SD sont différents des capteurs qui déclenchent l'alarme	Connaissance	PM	
	Connaissance de pilotage	Connaissance	PM	
Stratégie de vérification pertinence procédure	Stratégie utilisée pour vérifier la pertinence de la procédure	Information stratégique	PM	
Systèmes [P]	Page système (SD)	Couche interface	FWS	PM
	Boutons systèmes	Périphériques d'entrée	FWS	PM
	Systèmes impactés	Information	PM	
	Etats opérationnels des systèmes	Information	PM	
	Etat opérationnel de l'avion	Information	PM	
	Information systèmes	Information	PM	
	état de l'overhead panel	Information	PM	
	Conséquences visibles dans le cockpit	Information	PM	
	Page SD	Information	PM	
	Il y a d'autre informations systèmes	Information	PM	
	Bouton MORE	Information	PM	

	Information système supplémentaire	Information	PM	
	Boutons systèmes	Information	PM	
	Système choisi	Information	PM	
	Information du système sélectionné	Information	PM	
	Phase de vol	Objet Logiciel	FWS	
	Système impacté	Objet Logiciel	FWS	
Autre éléments contextuels	Elément contextuel (sensible seulement par les pilotes (fumée, malaise, …))	Information	PM	

3 Modèles de tâches et de fonctions pour la gestion des alarmes avec le système alternatif de gestion des alarmes

3.1 Modèles de tâches ségrégués du PM

Les tâches pour rappeler, supprimer ou effacer des alarmes ou recommandations étant similaires aux tâches avec les FWS, nous nous ne les détaillons pas de nouveau ci-après. Nous nous référons aux modèles de tâches ségrégués de l'annexe 1.1 pour le rappel, la suppression ou l'effacement des alarmes ou recommandations.

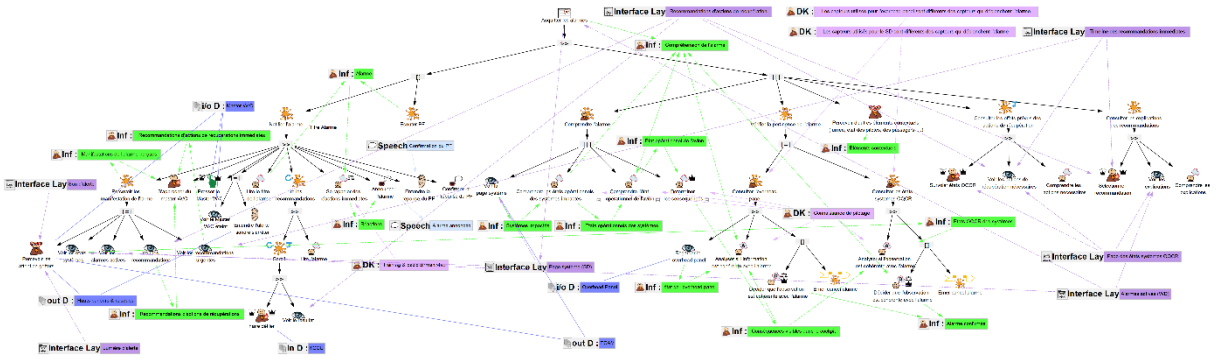


Figure 136. Modèle de tâches ségrégué du PM : Acquitter les alarmes

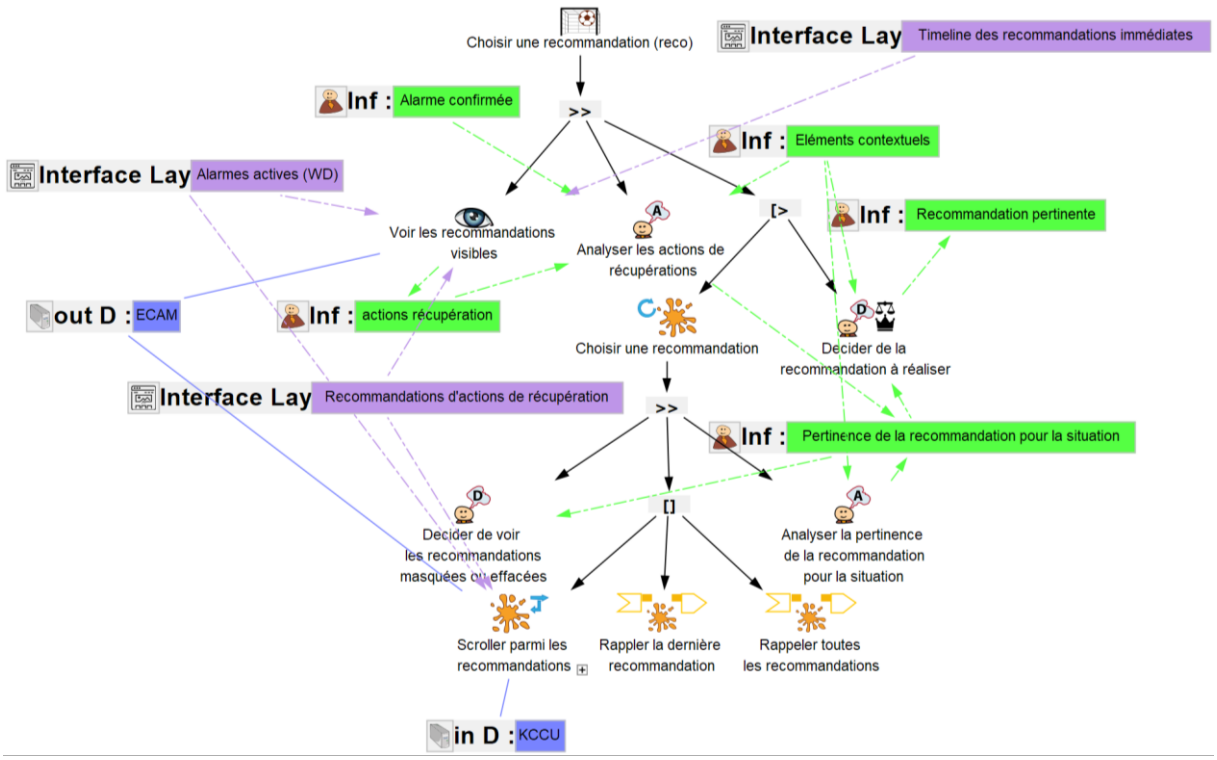


Figure 137. Modèle de tâches ségrégué du PM : Choisir une recommandation à réaliser.

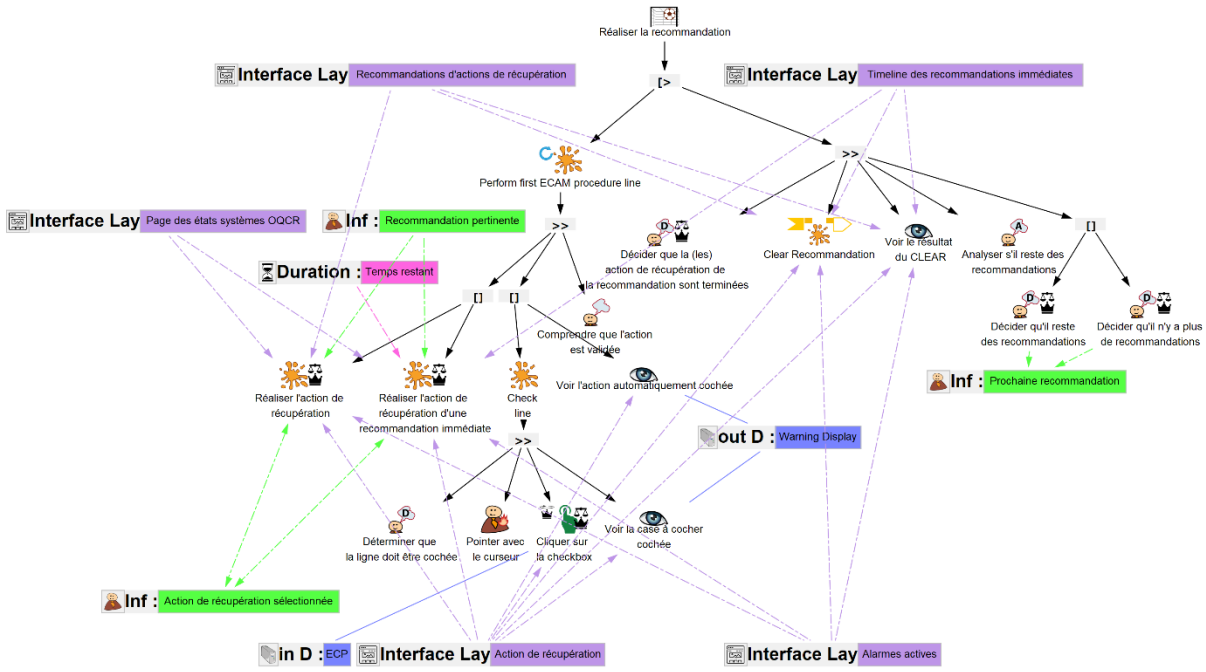


Figure 138.Modèle de tâches ségrégué du PM : Réaliser la recommandation sélectionnée.

3.2 Modèles de fonctions du Recommandeur

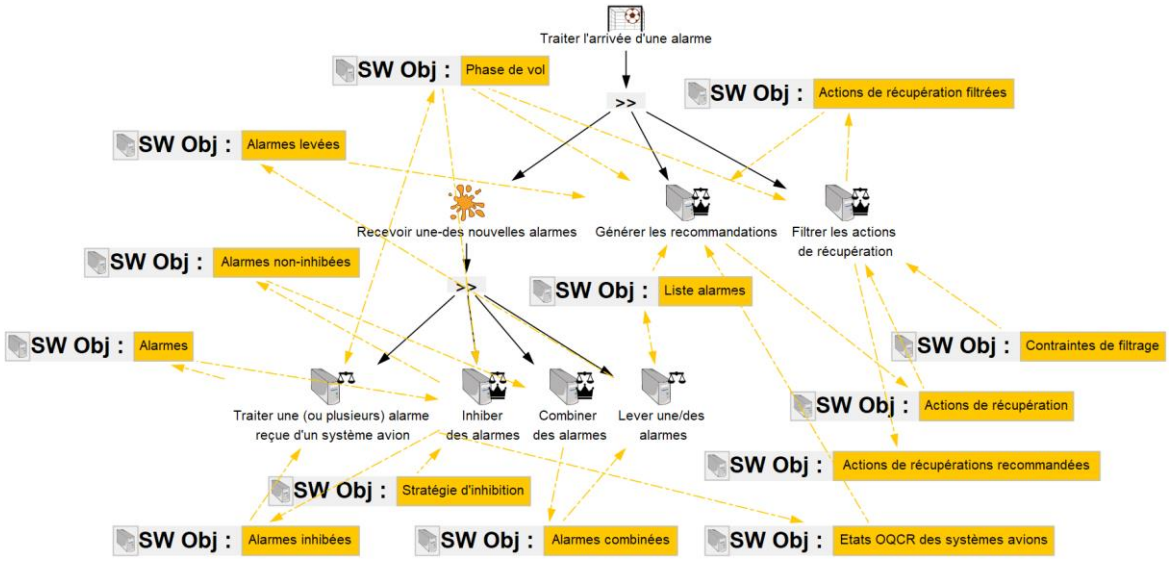


Figure 139. Modèle de fonctions du recommandeur : Traiter l'arrivée d'une alarme.

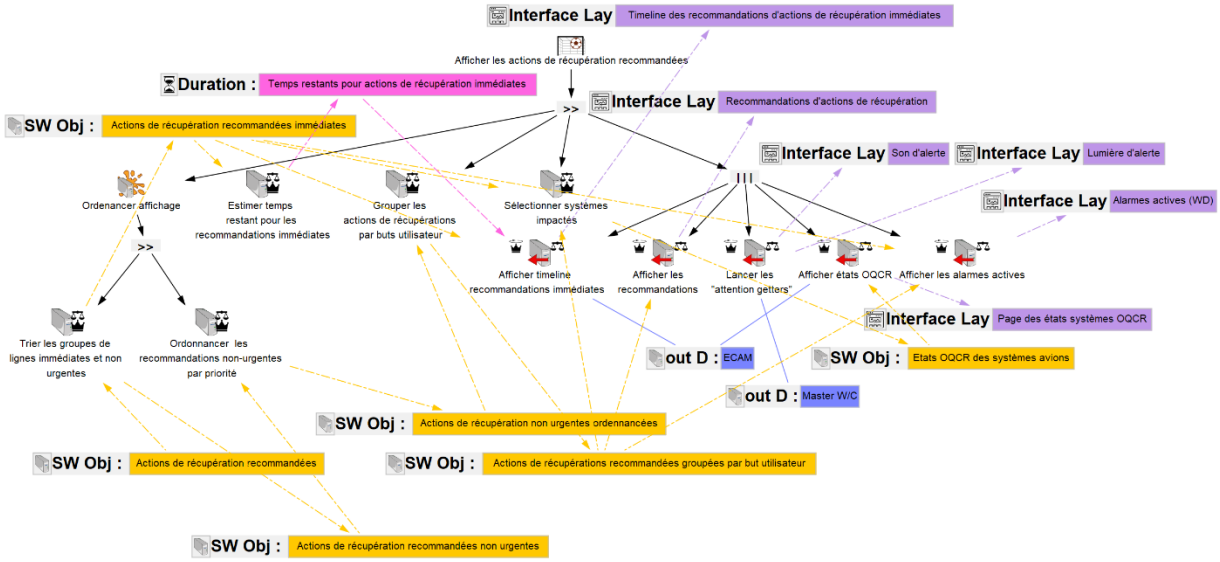


Figure 140. Modèle de fonctions du recommandeur : Afficher les recommandations.

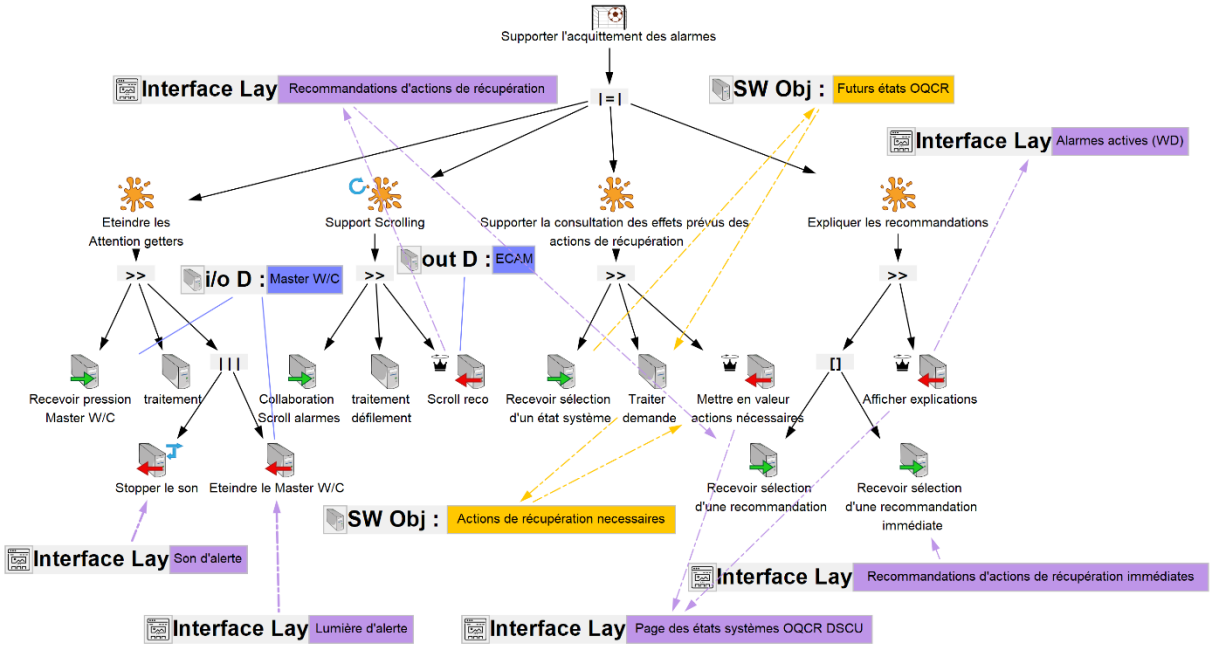


Figure 141. Modèle de fonction du recommandeur pour soutenir l'acquittement des alarmes.

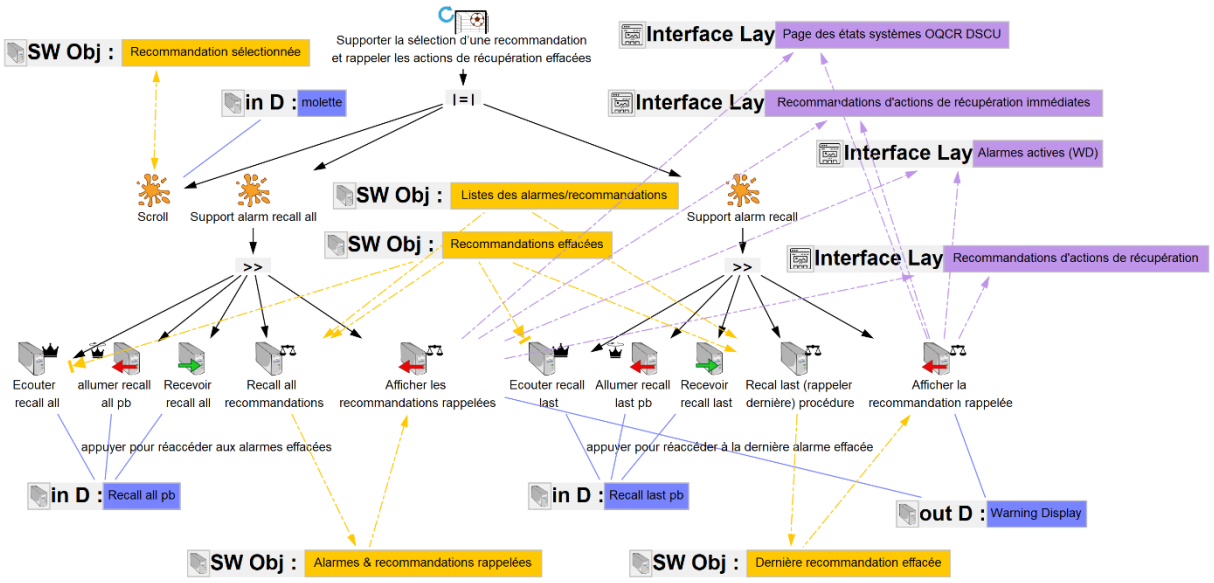


Figure 142. Modèle de fonctions du recommandeur : Soutenir la sélection d'une recommandation.

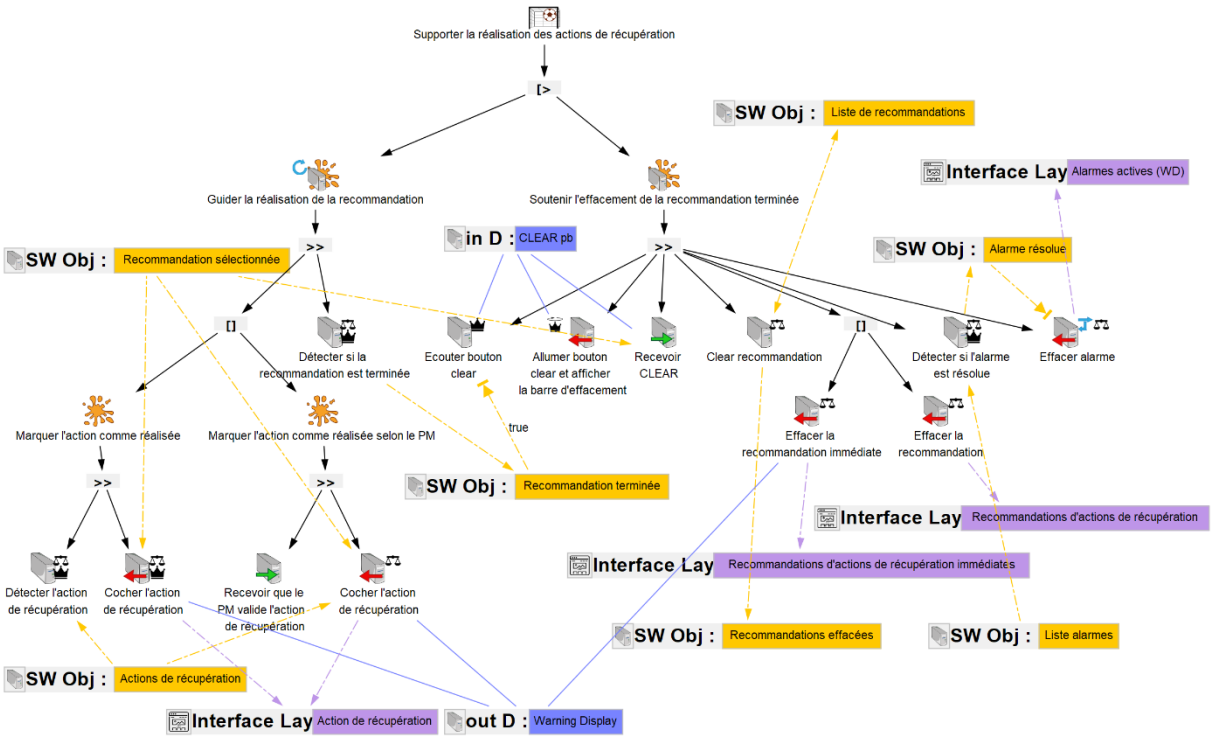


Figure 143. Modèle de fonctions du recommandeur : Soutenir la réalisation d'une recommandation.

4 Colonnes du tableau RCRA décrivant l'allocation des ressources pour la gestion des alarmes avec le système alternatif de gestion des alarmes

Tableau 47. Tableau décrivant l'allocation des ressources entre le système alternatif de gestion des alarmes (Recommandeur) et le pilote *monitoring* (PM) pour la résolution des alarmes.

Allocation des ressources				
Ressource [Partagée]	Description DOD	Type DOD	Possesseur	Acquéreur
Alarmes [P]	Lumière d'alerte	Couche interface	RECOMMANDEUR	PM
	Son d'alerte	Couche interface	RECOMMANDEUR	PM
	Hauts parleurs & casques	Périphérique d'entrée/sortie	RECOMMANDEUR	PM
	Master W/C	Périphérique d'entrée/sortie	RECOMMANDEUR	PM
	Alarmes (WD)	Couche interface	RECOMMANDEUR	PM
	Manifestation de l'alarme perçue	Information	PM	
	Alarme	Information	PM	
	Compréhension de l'alarme	Information	PM	
	Alarme annoncée	Parole	PM	PF
	Confirmation du PF	Parole	PF	PM
	Alarme confirmée	Information	PM	
	Alarmes	Objet Logiciel	RECOMMANDEUR	
	Alarmes inhibées	Objet Logiciel	RECOMMANDEUR	
	Alarmes non-inhibées	Objet Logiciel	RECOMMANDEUR	
	Alarmes combinées	Objet Logiciel	RECOMMANDEUR	
	Liste des alarmes	Objet Logiciel	RECOMMANDEUR	
	Alarmes actives	Objet Logiciel	RECOMMANDEUR	
	EMER CANCEL pb	Périphérique d'entrée	RECOMMANDEUR	PM
	EMER CANCEL annoncé	Parole	PF	PM
	Confirmation du PF	Parole	PF	PM
	EMER CANCEL pb	Information	PM	

Alarmes / recommandations [P]	Recall ALL pb	Périphérique d'entrée	RECOMMANDEUR	PM
	Recall LAST pb	Périphérique d'entrée	RECOMMANDEUR	PM
	Clear pb	Périphérique d'entrée	RECOMMANDEUR	PM
	CLEAR pb	Information	PM	
	Annoncer l'effacement de la procédure	Parole	PM	PF
	Confirmation du PF	Parole	PF	PM
	Recall all annoncé	Parole	PF	PM
	Confirmation du PF	Parole	PF	PM
	RECAL ALL pb	Information	PM	
	RECALL LAST annoncé	Parole	PF	PM
	Confirmation du PF	Parole	PF	PM
	RECAL LAST pb	Information	PM	
	Alarmes/recommandations rappelées	Objet Logiciel	RECOMMANDEUR	
	Listes des alarmes	Objet Logiciel	RECOMMANDEUR	
	Alarme résolue	Objet Logiciel	RECOMMANDEUR	
Actions de récupération [P]	Recommandations d'actions de récupération	Couche interface	RECOMMANDEUR	PM
	Timeline des recommandations immédiates	Couche interface	RECOMMANDEUR	PM
	Action de récupération	Couche interface	RECOMMANDEUR	PM
	Recommandations d'actions de récupérations	Information	PM	
	Recommandations d'actions de récupérations immédiates	Information	PM	
	Actions récupération	Information	PM	
	Pertinence de la recommandation pour la situation	Information	PM	
	Recommandation pertinente	Information	PM	
	Action de récupération sélectionnée	Information	PM	
	Temps restant	Durée	PM, RECOMMANDEUR	

	Prochaine recommandation	Information	PM	
	Bouton de validation	Périphérique d'entrée	RECOMMANDEUR	PM
	Réactions	Information	PM	
	Clear pb	Périphérique d'entrée	RECOMMANDEUR	PM
	CLEAR pb	Information	PM	
	Annoncer l'effacement de la procédure	Parole	PM	PF
	Confirmation du PF	Parole	PF	PM
	Actions de récupération	Objet Logiciel	RECOMMANDEUR	
	Actions de récupération filtrées	Objet Logiciel	RECOMMANDEUR	
	Actions de récupérations recommandées	Objet Logiciel	RECOMMANDEUR	
	Actions de récupération recommandées	Objet Logiciel	RECOMMANDEUR	
	Actions de récupération recommandées non immédiates	Objet Logiciel	RECOMMANDEUR	
	Actions de récupération recommandées immédiates	Objet Logiciel	RECOMMANDEUR	
	Actions de récupération non immédiates ordonnancées	Objet Logiciel	RECOMMANDEUR	
	Actions de récupérations recommandées groupées par but utilisateur	Objet Logiciel	RECOMMANDEUR	
	Temps restants pour actions de récupération immédiates	Objet Logiciel	RECOMMANDEUR	
	Alarmes & recommandations rappelées	Objet Logiciel	RECOMMANDEUR	
	Dernière recommandation effacée	Objet Logiciel	RECOMMANDEUR	
	Recommandations effacées	Objet Logiciel	RECOMMANDEUR	
	Listes des alarmes/recommandations	Objet Logiciel	RECOMMANDEUR	

	Recommandation sélectionnée	Objet Logiciel	RECOMMANDEUR	
	Recommandation terminée			
Stratégie de filtrage	Contraintes de filtrage	Objet Logiciel	RECOMMANDEUR	
	Stratégie d'inhibition	Objet Logiciel	RECOMMANDEUR	
Stratégie de vérification des alarmes	Les capteurs utilisés pour l'overhead panel sont différents des capteurs qui déclenchent l'alarme	Connaissance	PM	
	Les capteurs utilisés pour les états systèmes sont différents des capteurs qui déclenchent l'alarme	Connaissance	PM	
	Connaissance de pilotage	Connaissance	PM	
Systèmes [P]	Page des états systèmes OQCR	Couche interface	RECOMMANDEUR	PM
	Systèmes impactés	Information	PM	
	Etats opérationnels des systèmes	Information	PM	
	Etat opérationnel de l'avion	Information	PM	
	Etats OQCR des systèmes	Information	PM	
	état de l'overhead panel	Information	PM	
	Conséquences visibles dans le cockpit	Information	PM	
	Phase de vol	Objet Logiciel	RECOMMANDEUR	
	Système impacté	Objet Logiciel	RECOMMANDEUR	
	Futurs états OQCR	Objet Logiciel	RECOMMANDEUR	
Autre éléments contextuels	Elément contextuel (sensible seulement par les pilotes (fumée, malaise, …))	Information	PM	

5 Meta-modélisation des modèles de tâches ségrégués et des modèles de fonctions

5.1 Meta-modélisation des modèles de tâches ségrégués

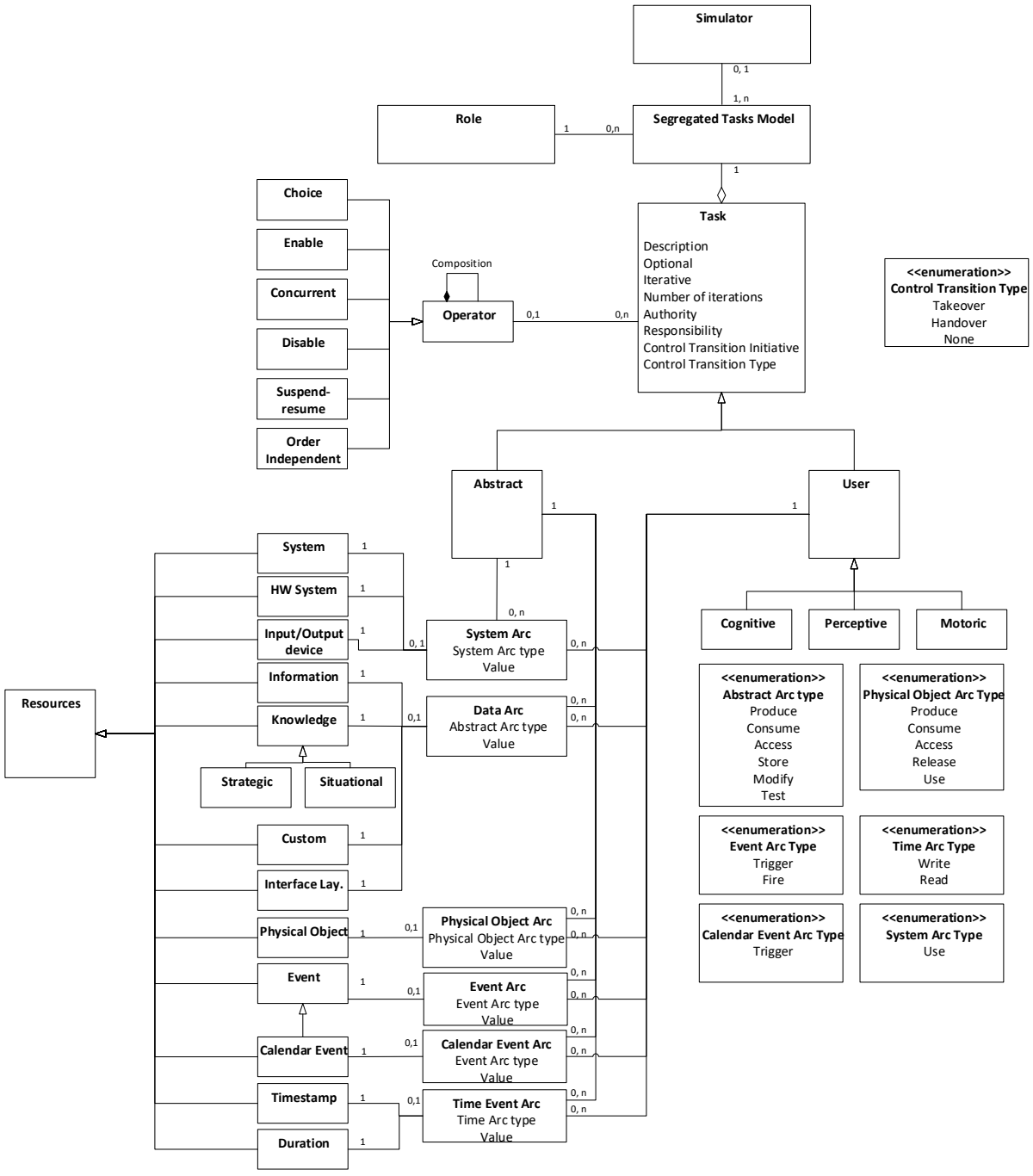


Figure 144. Méta-modèle d'un modèle de tâches ségrégué.

5.2 Meta-modélisation des modèles de fonctions

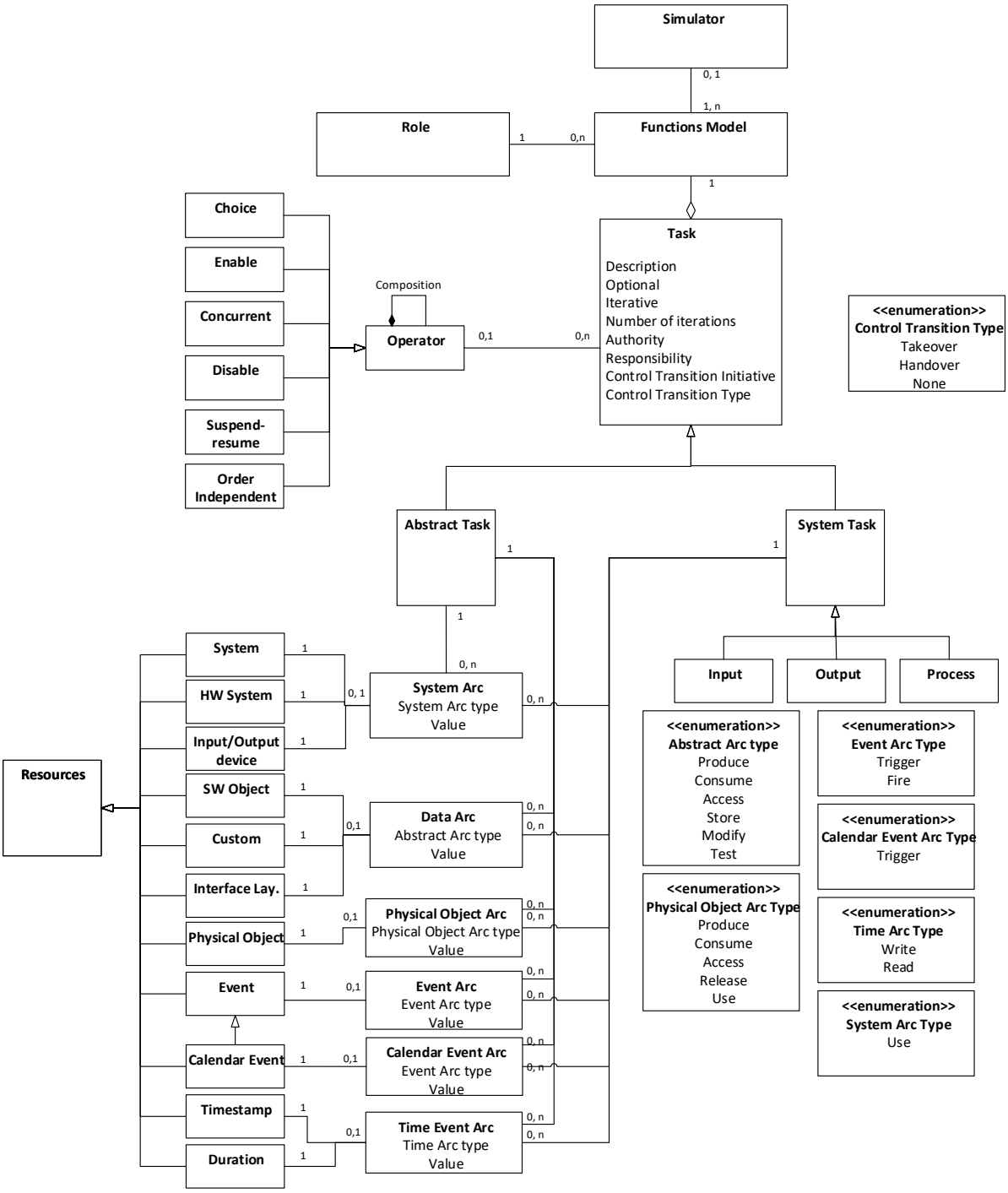


Figure 145. Méta-modèle d'un modèle de fonctions.

5.3 Meta-modélisation du protocole de coopération

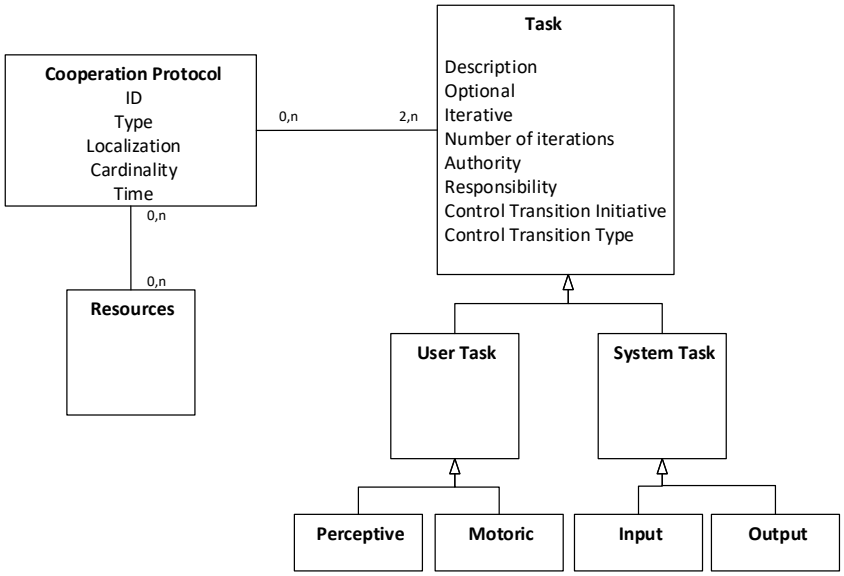


Figure 146. Méta-modèle d'un protocole de coopération.

6 ERRATA

Présentation des changements post-publication.

Modifications dans l'état de l'art : Dans le chapitre Etat de l'art, certains éléments sous forme de listes à points auraient dû rester des notes et ne pas apparaitre dans la version finale, ces éléments n'étant pas des écrits originaux mais provenant de travaux précédents. Cette nouvelle version de la thèse corrige cela. Les sections de l'état de l'art impactées sont indiquées par une note de bas de page.

Modifications générales dans le document : Cette nouvelle version corrige également quelques erreurs orthographiques ayant échappées à ma vigilance et présente une version française des figures qui contenaient originellement des termes anglais.

Ces modifications n'impactent en aucun cas les analyses et la contribution de la thèse et ont pour but d'améliorer la forme du manuscrit publié.

Je vous souhaite une bonne lecture.