

UNIVERSITÉ LARBI TEBESSI, TEBESSA  
Département de Mathématiques et Informatique  
Laboratoire de Mathématiques, d'Informatique et Systèmes (LAMIS)



## Thèse

Présentée pour l'obtention du diplôme de

**DOCTORAT LMD**

en

**Informatique**

---

# Approche Décisionnelle basée Cloud et Négociation Multi-Acteurs pour la Planification Territoriale Urbaine

---

Spécialité : **Systemes d'information coopératifs**

Présentée et soutenue publiquement le 06 novembre 2016 par

**Imene BENATIA**

### Jury

Pr. Nacira GHOUALMI	Université Badji Mokhtar–Annaba	Présidente
Pr. Robert LAURINI	LIRIS - INSA de Lyon – France	Examineur
Dr. Makhlouf DERDOUR	Université Larbi Tebessi – Tébessa	Examineur
Pr. Sadok BEN YAHIA	Faculté des Sciences de Tunis – Tunisie	Examineur
Pr. Mohamed Ridda LAOUAR	Université Larbi Tebessi – Tébessa	Directeur de thèse
Dr. Hakim BENDJENNA	Université Larbi Tebessi – Tébessa	Co-directeur de thèse

# Dédicace

Je dédie ce modeste travail,

A celui qui m'a toujours soutenu, supportant mes sauts d'humeur et mon stress, inconditionnellement présent auprès de moi, à toi mon cher mari ;

A celle qui m'a aidée et s'est sacrifiée pour moi, celle qui a toujours trouvé les mots pour me reconforter et su être à mes côtés ; à celle qui m'a soutenu tout au long de mon parcours, ma douce et adorable mère ;

A mon très cher père qui a su toujours être présent pour moi, à tout moment et en toutes circonstances ;

A ma très chère sœur avec qui j'apprends toujours de la vie, des acquis que je n'aurais jamais su avoir sans toi, merci Lamia ;

A ma chère amie et véritable sœur, Asma Kerouaz, symbole de tendresse et de fidélité ;

A tous mes amies avec qui je partage d'agréables moments : Mounia et Hana ;

A vous mes camarades de parcours les précieuses amies du LAMIS : Manel et Afrah.

Encore merci du fond du cœur,

Imene

# Remerciements

Je tiens premièrement à me prosterner en remerciant Allah le tout puissant de m'avoir donné le courage et la patience pour terminer ce travail.

Ce travail n'aurait jamais abouti sans l'aide, les encouragements et l'implication de certaines personnes à qui j'exprime à travers ces quelques phrases modestes toute ma gratitude, en particulier :

Je tiens tout d'abord à remercier et à exprimer toute ma gratitude approfondie à mon directeur de thèse Monsieur Mohamed Ridda LAOUAR, Professeur à l'Université de Tébessa et directeur du laboratoire LAMIS-Tébessa pour m'avoir permis de réaliser cette thèse dans de bonnes conditions. Je le remercie aussi pour sa patience et ses conseils, ainsi que pour toutes les riches réunions de travail, sans quoi je ne serai jamais arrivée à bout de ce travail, je ne le remercierai jamais assez pour m'avoir fait profiter de son expérience et de m'avoir témoigné tant de bienveillance. Je suis constamment impressionnée de constater à quel point il m'a aidée à atteindre mes buts, identifier et stimuler mon potentiel.

Mes remerciements vont aussi à Monsieur Hakim BENDJENNA, Docteur à l'Université de Tébessa qui m'a co-encadré durant toutes ces années de thèse pour sa disponibilité, son soutien, son aide précieuse, ses conseils et ses remarques constructives.

Je tiens à exprimer toute ma reconnaissance à Monsieur Sean B EOM, Professeur à l'Université Southeast Missouri en USA d'avoir accepté de répondre à mes e-mails sur le sujet d'étude, je le remercie pour son immense aide, pour son attention de tout instant sur mes travaux et pour ses conseils avisés. Je suis ravie d'avoir travaillé en sa compagnie.

Je remercie Monsieur Robert LAURINI, Professeur à l'INSA de Lyon pour m'avoir accueillie au sein du laboratoire de recherche LIRIS de Lyon en France et pour l'aide et les conseils qu'il m'a fournis tout au long de mon séjour au sein de son équipe. Je le remercie également pour l'honneur qu'il m'a fait pour sa participation à mon jury, pour le temps consacré à la lecture critique de cette thèse et les remarques judicieuses qu'il m'a indiqué.

Je remercie chaleureusement Madame Nacira GHOUALMI, Professeur à l'Université d'Annaba pour avoir accepté de présider le jury de la soutenance. Je la remercie pour l'intérêt et la considération qu'elle a porté à mes travaux. Je lui adresse ma profonde gratitude.

Je remercie Monsieur Sadok BEN YAHIA, Professeur à la Faculté des Sciences de Tunis pour l'honneur qu'il me fait en acceptant d'examiner ce travail et de siéger parmi les membres de ce jury de thèse. Je tiens à l'assurer de ma sincère reconnaissance et de mon profond respect.

Je remercie également Monsieur Makhlof DERDOUR, Docteur à l'Université de Tébessa pour toute l'attention dont il a fait preuve en examinant cette thèse et pour l'honneur qu'il me fait en participant à ce jury.

Je tiens à remercier Imene HANNACHE, Ingénieur d'état en architecture au bureau d'étude General Art & Technique (GART) à Annaba, qui n'a pas hésité à me donner toutes les informations sur projets urbains proposés.

Enfin, je tiens à remercier tous les membres de notre laboratoire de recherche LAMIS.

# Résumé

La prise de bonnes décisions en planification territoriale urbaine est une tâche qui devient de plus en plus difficile. Cela est dû, par le fait que les problèmes territoriaux sont de nature complexe et qui nécessitent la définition de plusieurs critères et font intervenir plusieurs entités aux intérêts conflictuels avec des points de vue différents.

Dans cette étude, le travail présenté a pour but de développer des systèmes décisionnels afin d'apporter une aide pertinente aux décideurs de la ville dans l'évaluation des projets urbains. Dans ce contexte, nous avons proposé deux contributions décisionnels, la première contribution consiste à une utilisation combinée du système d'aide à la décision (DSS) et le Cloud Computing où le système décisionnel est composé de deux phases essentiels : la phase d'analyse multicritère et la phase de négociation. La seconde contribution consiste en une proposition d'un nouveau protocole de négociation basé sur le protocole le plus célèbre qui est le Contract net.

**Mots clé :** Planification urbaine, Projet urbain, Système d'aide à la décision, Aide à la décision multicritère, Négociation, Cloud Computing.

# Abstract

Making good decisions in territorial urban planning becomes more and more difficult. This is due to the fact that territorial problems are complex and require the definition of several criteria and involve several entities with conflictual interests and different viewpoints.

In this study, the presented work aims at developing decision systems to provide relevant assistance to decision-makers of the city in the evaluation of urban projects. In this context, we proposed two decision contributions; the first contribution is a combined use of a decision support system (DSS) and Cloud Computing in which the decision-making system is composed of two main phases: multi-criteria analysis and negotiation phase. The second contribution is a proposal of a new negotiation Protocol based on the most famous protocol *i.e.* the Contract Net protocol.

**Keywords:** Urban planning, urban project, decision support system, multi-criteria decision system, Negotiation, Cloud Computing.

## الملخص

اتخاذ قرارات جديدة في التخطيط الحضري أصبحت مهمة صعبة, و يرجع إلى حقيقة أن القضايا الإقليمية معقدة و تتطلب تعريف العديد من المعايير و تدخل عدة شخصيات تمتلك وجهات نظر مختلفة و متعاكسة في هذه الدراسة, يهدف هذا إلى العمل تطوير نظام دعم اتخاذ القرار لتقديم المساعدات لصناع القرار في تقييم المشاريع الحضرية.

في هذا السياق, اقترحنا مساهمتين, المساهمة الأولى هي الجمع بين نظام دعم القرار و الحوسبة السحابية حيث يتكون نظام دعم القرار من مرحلتين : مرحلة التحليل المتعدد المعايير و مرحلة التفاوض. المساهمة الثانية هي اقتراح لوضع بروتوكول جديد للتفاوض مبني على أساس البروتوكول كمنتركت نات .

**الكلمات البحث:** التخطيط الحضري, مشروع حضري, نظام دعم القرار, دعم اتخاذ القرار متعدد المعايير, التفاوض, الحوسبة السحابية.

# Table des matières

Dédicace.....	2
Remerciements.....	3
Résumé.....	5
Abstract.....	6
الملخص.....	7
Liste des figures.....	12
Liste des tableaux.....	14
<b>Introduction Générale</b> .....	<b>15</b>
<b>Chapitre 1. Planification urbaine</b> .....	<b>19</b>
1.1. Introduction.....	20
1.2. Principes de la planification urbaine.....	20
1.3. Les objectifs de la planification urbaine.....	22
1.4. La planification urbaine en Algérie.....	22
1.4.1. Les instruments de la planification urbaine.....	22
1.4.2. Evolution de planification urbaine.....	24
1.4.3. Le développement durable.....	24
1.5. Les principes fondamentaux de la planification urbaine.....	25
1.6. L'évaluation dans le cadre de la planification urbaine.....	26
1.6.1. Notion du projet urbain.....	27
1.6.1.1. Types de projets urbains.....	27
1.6.1.2. Acteurs du projet urbain.....	28
1.6.1.3. Principales étapes du projet urbain.....	29
1.6.2. La prise de décision en planification urbaine.....	31
1.7. Conclusion.....	32
<b>Chapitre 2. Aide à la décision multicritère</b> .....	<b>34</b>
2.1. Introduction.....	36
2.2. Aide à la décision.....	36
2.2.1. Définition de la décision.....	36
2.2.2. Définition de l'aide à la décision.....	36
2.2.3. Les différentes approches d'aide à la décision.....	37
2.2.4. Les différentes étapes du processus d'aide à la décision.....	37

2.2.5. Système d'aide à la décision (DSS).....	39
2.3. Aide à la décision multicritère.....	40
2.3.1. De monocritère vers le Multicritère.....	41
2.3.1.1. Le monocritère.....	41
2.3.1.2. Le multicritère .....	42
2.3.1.3. Les frontières entre le monocritère et le multicritère .....	44
2.3.2. L'aide à la décision multicritère .....	44
2.3.2.1. Formulation d'un problème multicritère de décision .....	44
2.3.2.2. Les approches multicritères .....	44
2.3.2.3. Les étapes d'aide à la décision multicritère .....	46
2.3.3. Les problématiques multicritères.....	47
2.3.4. Les méthodes multicritères.....	50
2.3.4.1. Les Méthodes décisionnelles multicritère de l'approche du critère unique de synthèse...50	
2.3.4.2. Les méthodes de sur-classement selon l'approche du sur-classement de synthèse .....	54
2.3.4.3. Les méthodes interactives selon l'approche du jugement local interactif .....	61
2.3.5. Algorithme de sélection d'une méthode d'aide à la décision multicritère.....	61
2.3.5.1. Caractérisation des inputs .....	62
2.3.5.2. Caractérisation des outputs .....	63
2.4. Etat de l'art : les systèmes décisionnels en planification urbaine.....	64
2.5. Conclusion.....	66
<b>Chapitre 3. Négociation entre acteurs</b>	<b>67</b>
3.1. Introduction.....	68
3.2. Définition de la négociation .....	68
3.3. Les différentes formes de la négociation .....	68
3.3.1. Les systèmes de vote .....	69
3.3.2. Les enchères .....	70
3.3.3. Les systèmes contractuels.....	70
3.3.4. Les systèmes de négociation basés sur l'argumentation .....	71
3.4. Les différents systèmes de négociation automatisée.....	72
3.4.1. Le Kasbah.....	72
3.4.2. L'AuctionBot .....	73
3.4.3. Le Fishmarket.....	73
3.4.4. Le système Adept .....	74
3.4.5. Magnet .....	74
3.5. Les différents systèmes de négociation automatisée proposés en planification urbaine.....	74

3.6. Conclusion.....	75
<b>Chapitre 4. Cloud Computing pour l'intégration du DSS</b>	<b>77</b>
4.1. Introduction.....	78
4.2. Présentation du Cloud Computing.....	78
4.3. Les services Cloud.....	79
4.4. Les modèles de déploiement.....	81
4.5. La virtualisation dans le Cloud Computing .....	81
4.6. Les solutions Cloud open source .....	82
4.6.1. Solutions IaaS.....	83
4.6.2. Solutions PaaS.....	88
4.7. Les solutions Cloud propriétaires .....	89
4.7.1. Google.....	89
4.7.2. Salesforce.com .....	90
4.7.3. Amazon Web Services .....	92
4.8. Conclusion.....	94
<b>Chapitre5.Contributions décisionnelles pour l'évaluation des projets urbains</b>	<b>96</b>
5.1. Introduction.....	97
5.2.Contribution1:un système d'aide à la décision basé Cloud pour l'évaluation des projets urbains .	97
5.2.1. Approche proposée [Benatia et al. 2015].....	97
5.2.1.1. Modèle décisionnel [Benatia et al. 2015].....	98
5.2.1.2. Intégration du modèle décisionnel dans un Cloud privé .....	101
5.2.2. Synthèse .....	108
5.3. Contribution 2 : Système de négociation décisionnelle pour l'évaluation des projets urbains .....	108
5.3.1. Présentation du protocole Contract Net .....	108
5.3.2. Adaptation du protocole Contract Net pour le problème de la planification urbaine [Benatia et al. a 2016].....	110
5.4. Conclusion.....	112
<b>Chapitre 6. Implémentation et mise en œuvre</b>	<b>114</b>
6.1. Introduction.....	115
6.2. Etude de cas.....	115
6.3. Validation du système d'aide à la décision basé Cloud .....	117
6.3.1. Application de la méthode PROMETHEE II [Benatia et al. 2015].....	117
6.3.2. Application de la méthode Hare [Benatia et al. 2015] .....	121
6.3.3. Intégration de notre DSS dans un Cloud privé.....	123
6.3.3.1. Installation d'openstack (quick start).....	123

6.3.3.2. Installation de Cloudbees (quick start).....	124
6.3.3.3. Déploiement du DSS dans le Cloud [Benatia et al. b 2016].....	125
6.4. Validation du système décisionnel basé sur la négociation[Benatia et al. a 2016].....	129
6.5. Conclusion.....	134
<b>Conclusion Générale</b>	<b>135</b>
Publications et Communications .....	138
Bibliographie.....	140
Webographie .....	149
Annexes.....	150

# Liste des figures

Figure 1.1. Quelques composantes du système urbain .....	21
Figure 1.2. Les piliers du développement durable .....	25
Figure 1.3. Principes de la planification urbaine .....	26
Figure 1.4. Processus de conduite d'un projet urbain .....	29
Figure 1.5. Relations entre les décideurs, le territoire et le DSS.....	31
Figure 1.6. Processus d'aide à la décision territoriale.....	32
Figure 2.1. Les étapes du processus d'aide à la décision. ....	39
Figure 2.2. Structure d'un DSS .....	40
Figure 2.3. Relation de dominance.....	42
Figure 2.4. Solutions efficaces. ....	43
Figure 2.5. Démarche bottom-up de Roy .....	45
Figure 2.6. Démarche top-down de Keeney .....	45
Figure 2.7. Démarche intermédiaire de Laaribi. ....	46
Figure 2.8. La problématique du choix.....	48
Figure 2.9. La problématique du tri.....	48
Figure 2.10. La problématique du rangement .....	49
Figure 2.11. Problématique de la description .....	49
Figure 2.12. La structure hiérarchique AHP .....	53
Figure 2.13. Différents formes de courbes de PROMETHEE .....	55
Figure 4.1. Cloud Computing .....	79
Figure 4.2. L'architecture pyramidal IaaS, PaaS, SaaS .....	80
Figure 4.3. Différentes couches de la virtualisation.....	82
Figure 4.4. Architecture d'Eucalyptus.....	83
Figure 4.5. Vu interne de l'architecture d'OpenNebula.....	84
Figure 4.6. Vu interne de l'architecture de Nimbus.....	85
Figure 4.7. Architecture de Xen Cloud Platform. ....	85
Figure 4.8. Architecture d'Abi Cloud.....	86
Figure 4.9. Architecture d'OpenStack.....	86
Figure 4.10. Pourcentage d'utilisation d'OpenStack .....	88
Figure 5.1. Modèle décisionnel proposé [Benatia et al. 2015].....	98
Figure 5.2. Nova.....	102
Figure 5.3. Architecture logique de Nova .....	102
Figure 5.4. Architecture globale de Glance .....	104

Figure 5.6. Le PaaS CloudBees vu de l'extérieur .....	105
Figure 5.7. Système décisionnel basé Cloud pour l'évaluation des projets urbains [Benatia et al. 2015] .....	106
Figure 5.8. Algorithme du protocole Contract Net .....	109
Figure 5.9. Le diagramme de séquence du protocole Contract Net .....	109
Figure 5.10. Le scénario de notre protocole proposé.....	110
Figure 5.11. Algorithme de notre protocole de négociation[Benatia et al. a 2016].....	111
Figure 5.12. Diagramme de séquence pour notre protocole de négociation[Benatia et al. a 2016].....	112
Figure 6.1. Affectation des poids.....	117
Figure 6.2. Calcul des flux pour chaque projet [Benatia et al. 2015].....	117
Figure 6.3. Rangement des projets [Benatia et al. 2015] .....	118
Figure 6.4. Spécification des poids des critères par le décideur 1 [Benatia et al. 2015].....	118
Figure 6.5. Spécification des poids des critères par le décideur 2.....	119
Figure 6.7. Spécification des poids des critères par le décideur 4.....	120
Figure 6.8. Spécification des poids des critères par le décideur 5.....	120
Figure 6.9. Rangements des projets d'habitat en utilisant PROMETHEE II [Benatia et al. 2015]. ....	121
Figure 6.10. Processus de négociation en utilisant la méthode Hare [Benatia et al. 2015]. ....	122
Figure 6.11. L'interface principale d'OpenStack [Benatia et al. b 2016].....	124
Figure 6.12. L'interface principale du CloudBees.....	125
Figure 6.13. Création d'une image [Benatia et al. b 2016].....	126
Figure 6.14. Création d'un flavor [Benatia et al. b 2016].....	126
Figure 6.15. Création d'un volume [Benatia et al. b 2016].....	127
Figure 6.16. Lancement d'une machine virtuelle [Benatia et al. b 2016] .....	127
Figure 6.17. Création des machines virtuelles en utilisant OpenStack [Benatia et al. b 2016] .....	128
Figure 6.18. Machine virtuelle lancée [Benatia et al. b 2016] .....	128
Figure 6.19. Intégration du DSS dans CloudBees [Benatia et al. b 2016] .....	129
Figure 6.20. Invitation des décideurs pour l'évaluation des projets urbains [Benatia et al. a 2016]....	130
Figure 6.21. Réponse des décideurs [Benatia et al. a 2016].....	130
Figure 6.22. Demande de proposition.....	131
Figure 6.23. Choix des projets urbains par les décideurs [Benatia et al. a 2016].....	131
Figure 6.24. Evaluation des votes.....	132
Figure 6.25. Demande de vote pour les projets en égalité. ....	132
Figure 6.26. Choix entre les projets en égalité [Benatia et al. a 2016].....	133
Figure 6.27. Evaluation du deuxième tour de vote [Benatia et al. a 2016] .....	133
Figure 6.28. Notification de la fin de la négociation [Benatia et al. a 2016].....	134

# Liste des tableaux

Tableau 2.1. Matrice d'évaluation Alternatives/Critères .....	47
Tableau 2.2. Les différentes problématiques de décision. ....	49
Tableau 2.3. Matrice multicritère. ....	51
Tableau 2.4. Matrice des fonctions de valeur partielle. ....	51
Tableau 2.5. Échelle de Saaty pour la méthode AHP. ....	53
Tableau 2.6. Matrice de l'indice de cohérence moyen (ICM).....	54
Tableau 2.7. Classification des MADMC selon la matrice inputs/outputs.....	62
Tableau 2.8. Caractérisation des entrées.....	63
Tableau 2.9. Caractérisation des sorties.....	64
Tableau 2.10. Les DSS proposés pour la planification urbaine. ....	65
Tableau 4.1. Solution IaaS open source de Cloud .....	87
Tableau 4.2. Solution PaaS open source de Cloud.....	89
Tableau 6.1. Caractéristiques des projets d'habitat [Benatia et al. 2015] [Benatia et al. a 2016].....	116
Tableau 6.2. Rangements des projets selon chaque décideur [Benatia et al. 2015].....	121

# Introduction Générale

De ses modestes origines il ya environ 45 ans, le domaine des systèmes d'aide à la décision (DSS) a accompli des progrès significatifs afin de devenir une discipline académique cohérente et substantielle. Plusieurs sous-domaines de recherche sur le système d'aide à la décision (DSS) ont été identifiés, au cours de plus de quatre décennies notamment les fondations, la conception, l'implémentation, l'évaluation, les interfaces utilisateurs, les systèmes décisionnels de groupe, la gestion du modèle et les différents critères du système d'aide à la décision. Lors de l'élaboration de ces sous-spécialités du DSS, les chercheurs ont pris en considération les théories et les concepts développés pour élaborer des disciplines en science des systèmes, en sciences cognitives, en sciences informatiques, en sciences de gestion, en psychologie, en sciences de la communication, etc.

En plus de l'élaboration des théories sur les systèmes décisionnels, le développement d'un système d'aide à la décision (DSS) spécifique a été une activité importante dans la recherche des DSS pour les chercheurs. Au cours des 45 dernières années, environ 600 applications spécifiques et fonctionnelles du DSS ont été élaborées et publiées dans des journaux en langue anglaise [Eom 2007]. La majorité de ces applications étaient dans le domaine de la gestion fonctionnelle de l'entreprise tel que la gestion des productions, le marketing, le transport et la logistique, la gestion des systèmes d'information, les systèmes multifonctionnels, les finances, la gestion stratégique, et les ressources humaines. Des applications décisionnelles ont été développées également dans d'autres domaines hors entreprise tel que le gouvernement, l'éducation, les ressources naturelles, la santé, le domaine militaire, l'agriculture, la planification et l'administration urbaine.

La planification urbaine représente un ensemble d'études, de démarches, voire de procédures juridiques ou financières, qui permettent le développement physique, social, économique et environnemental des milieux urbains, et la mise en œuvre des options acceptées. Une des plus grandes politiques de la planification urbaine est la ville intelligente. Celle-ci correspond à l'utilisation et l'insertion de nouvelles technologies de l'information et de communications (TIC) dans différents secteurs de la ville, dans l'optique du développement durable, dans le but d'optimiser l'emploi des infrastructures existantes à savoir : le transport, le bâtiment, la gouvernance ou l'environnement. Les nouvelles technologies peuvent contribuer à répondre aux défis urbains actuels. Selon Carlo Ratti [Ratti 2012] la ville intelligente a pour but de rendre le territoire urbain technologique et interconnecté, propre, attractif, rassurant, efficace, ouvert et collaboratif, créatif, digital et verte(e-green). L'introduction de l'intelligence dans le territoire urbain (ville) permet aux acteurs d'acquérir une meilleure connaissance du territoire, de projeter, de définir, d'animer et d'évaluer les politiques et les actions de développement territorial durable.

La planification territoriale urbaine est un domaine très vaste, dont les problématiques sont nombreuses et diverses ; mais dans cette étude, nous nous intéressons, plus particulièrement à la problématique de l'évaluation des projets urbains proposés qui consiste à choisir le projet le plus satisfaisant.

Cependant, ce type de problématique est réputé être complexe en faisant intervenir plusieurs entités aux intérêts et opinions conflictuels. En effet le même projet urbain est vu différemment par des décideurs différents tels que : écologiste, homme politique, sociologue, économiste, ...etc. Chacun de ces acteurs possède une perception différente selon ses objectifs et ses préoccupations, de plus les projets urbains proposés dans un processus de décision, sont en général caractérisés par des critères d'évaluation multiples (l'effet environnemental, social, économique,...). Cette situation de décision où il existe plusieurs décideurs et plusieurs critères d'évaluation constitue ce qui est connu en littérature sous le vocable d'aide à la décision multicritère de groupe constituant un cadre plus réaliste pour résoudre des problèmes de décision car généralement, il est impossible de pouvoir évaluer la qualité d'une décision par un seul indicateur. De ce fait, il est nécessaire d'apporter une aide efficace aux décideurs de la ville pour l'évaluation des projets urbains.

Les systèmes décisionnels sont de plus en plus mis à contributions dans les projets de la planification urbaine. Diverses recherches sont maintenant orientées vers la modélisation des systèmes complexes et du développement d'instruments décisionnels (aide à la décision). Ces nouveaux modèles prennent en considération les divers piliers du développement durable à savoir économique, social et écologique et représentent un outil efficace pour atteindre l'objectif d'une ville intelligente.

Le territoire représente un système complexe dont la gestion est régulièrement source de conflits. Cela est dû, au fait qu'il existe une corrélation entre des acteurs provenant de domaines différents souvent opposés. Dans ce cadre, les intervenants du système sont les différents décideurs de la ville qui disposent de leurs propres informations, préférences, stratégie de décision, contraintes et objectifs généralement non partagés. Cela veut dire que le processus de décision est partagé entre différents acteurs impliqués et impactés par cette décision de groupe. La résolution de ce genre de problème consiste à trouver une décision commune acceptable à tous les décideurs.

Dans ce contexte, nous proposons des contributions pour la résolution de ce problème décisionnel en utilisant les trois technologies : système aide à la décision multicritère (MDSS), la négociation et le Cloud Computing qui sont de plus en plus mises à contributions dans les projets de la planification territoriale urbaine.

L'aide à la décision multicritère est un paradigme qui prend en compte plusieurs critères dans le processus décisionnel. La plus part des recherches visent à développer des systèmes plus ou moins formalisés dans le but d'améliorer, de faciliter et d'accompagner le décideur dans le déroulement du processus décisionnel. En effet plusieurs méthodes et modèles sont développés en aide à la décision multicritère qui ont un objectif commun c'est aider le décideur à prendre une décision qui le satisfait. L'ensemble des méthodes et des modèles de l'analyse multicritère fournies des décisions automatisées qui peuvent ne pas satisfaire l'ensemble des décideurs et cela engendre un conflit entre eux, en effet un mécanisme de négociation est nécessaire dans ce cas. La négociation est une démarche à l'aide de laquelle de nombreux acteurs arrivent à une décision ou à un accord commun. La négociation a pour but de résoudre les conflits et les débats qui peuvent survenir entre les différents décideurs lors d'un processus décisionnel en employant différentes méthodes et protocoles.

Nos études sur le développement des applications décisionnelles ont indiqué qu'au cours des dernières décennies, les chercheurs de DSS ont continuellement appliqué des théories, des outils et des techniques afin d'améliorer les systèmes décisionnels. Plusieurs décennies d'efforts de développement du système décisionnel soulignent que les décisions de gestion dans les organisations sont invariables. Cependant, les technologies de l'information sont constamment modifiées. Par conséquent, le développement de la nouvelle génération de DSS équipé de nouvelles technologies telles que le Cloud Computing est le rôle principal des chercheurs des systèmes d'information pour fournir aux gestionnaires des outils de décision pratiques.

Le Cloud Computing est défini par le NIST comme étant « une architecture moderne qui offre un accès au réseau à la fois pratique et à la demande. Le Cloud Computing est considéré comme un ensemble partagé de ressources informatiques configurables (réseaux, serveurs, stockage, applications, services...) qui peut être rapidement provisionné et diffusé moyennant un effort d'administration minimale ou une interaction des fournisseurs de services minimale ». La tendance croissante de la virtualisation et du Cloud Computing ont poussé les applications de l'informatique décisionnelle de migrer à partir du système d'information interne de l'organisation vers le Cloud Computing. Les fournisseurs du Cloud Computing offrent un nouveau service qui est l'informatique décisionnelle (Business Intelligent BI) en tant que service (Business Intelligent as a Service BaaS), en plus des trois modèles de service: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), et Software as a Service (SaaS). L'avancement des technologies du Cloud Computing permet aux urbanistes de surveiller tous les détails des processus de planification métropolitaine et les impliquer dans la conception et le processus de prise de décision avec un système d'aide à la décision basé Cloud.

L'utilisation du Cloud Computing pour notre système décisionnel a de nombreux avantages tel que : la facilité de communication et d'accessibilité des différents décideurs sans contraintes temporelles ou géographiques via une connexion Internet, la diminution du temps de développement et de déploiement du système décisionnel, la réduction des coûts de l'achat des serveurs, la sécurité des données, la disponibilité sur de nombreux appareils (ordinateur, smart phone, tablette, ...) ne nécessitant aucune autre installation en dehors d'un navigateur Internet et la facilité de la maintenance du système (la mise à jour du système décisionnel).

Notre première suggestion consiste à proposer un nouveau système décisionnel pour l'évaluation des projets urbains basé sur l'utilisation des méthodes de l'aide multicritère à la décision et les méthodes de négociation. Afin d'améliorer notre système décisionnel (DSS) dédié à l'évaluation des projets urbains nous avons suggéré de l'intégrer dans un Cloud privé. Notre seconde contribution consiste à proposer un nouveau protocole de négociation dédié à l'évaluation des projets urbains. La proposition d'un système décisionnel multicritère basé Cloud et celle d'un nouveau protocole de négociation sont susceptibles d'apporter une aide aux décideurs de la ville dans l'évaluation des projets urbains.

La suite de cette thèse est structurée en six chapitres, après ce chapitre introductif :

### **Chapitre 1. Planification urbaine**

Ce Chapitre donne un aperçu sur la planification territoriale urbaine dans lequel nous présenterons ses différents aspects en abordant ses principes généraux à l'échelle nationale, ses différents instruments et son évolution. Puis nous étudierons l'évaluation dans le contexte de cette planification.

## **Chapitre 2. Aide à la décision multicritère**

Ce chapitre expose en premier les phases d'une méthodologie d'aide à la décision en nous appuyant sur ce qui est proposé dans la littérature. Nous verrons ensuite une présentation générale sur les concepts structurant l'aide à la décision multicritère, ses différentes problématiques et méthodes les plus connues par la suite nous présenterons un état de l'art sur les systèmes décisionnels (DSS) dédiés à la résolution des problèmes de la planification urbaine.

## **Chapitre 3. Négociation entre acteurs**

Au cours de ce chapitre nous présenterons les définitions de la négociation les plus utilisées par les différents chercheurs du domaine, par la suite nous étudierons les diverses formes de la négociation et les systèmes de négociation automatique les plus courants.

## **Chapitre 4. Cloud Computing pour l'intégration du DSS**

Dans ce chapitre nous expliquerons, quelques généralités et concepts fondamentaux concernant le Cloud Computing, puis nous mènerons une étude globale sur les diverses solutions Cloud opensources et propriétaires des plateformes du Cloud Computing, Cela nous permet d'avoir et d'exposer une réflexion riche sur les différentes techniques disponibles afin de créer notre environnement Cloud et de choisir notre plateforme consacrée à l'intégration du DSS.

## **Chapitre 5. Contributions décisionnelles pour l'évaluation des projets urbains**

Ce chapitre décrit les deux modèles décisionnels proposés et qui essayent de satisfaire les besoins de la planification territoriale urbaine. Le premier modèle décisionnel propose une intégration d'un système d'aide à la décision avec la nouvelle technologie Cloud Computing. Et le second modèle propose un nouveau protocole de négociation basé sur le protocole le plus célèbre à savoir Contract Net.

## **Chapitre 6. Implémentation et mise en œuvre**

Nous détaillons, dans ce chapitre les étapes de développement et de mise en œuvre de nos systèmes décisionnels proposés avec une étude de cas réels tout en discutant les résultats obtenus.

Nous terminerons ce manuscrit par une conclusion générale et nos perspectives.

# Chapitre 1

## Planification urbaine

*« L'objectif de ce chapitre est de décrire les réflexions les plus importantes dans le contexte de la planification territoriale urbaine. »*

---

### Sommaire

1.1. Introduction.....	20
1.2. Principes de la planification urbaine.....	20
1.3. Les objectifs de la planification urbaine.....	22
1.4. La planification urbaine en Algérie.....	22
1.4.1. Les instruments de la planification urbaine.....	22
1.4.2. Evolution de planification urbaine.....	24
1.4.3. Le développement durable.....	24
1.5. Les principes fondamentaux de la planification urbaine.....	25
1.6. L'évaluation dans le cadre de la planification urbaine.....	26
1.6.1. Notion du projet urbain.....	27
1.6.1.1. Types de projets urbains.....	27
1.6.1.2. Acteurs du projet urbain.....	28
1.6.1.3. Principales étapes du projet urbain.....	29
1.6.2. La prise de décision en planification urbaine.....	31
1.7. Conclusion.....	32

## 1.1. Introduction

Les milieux urbains constituent une dimension fondamentale dans le monde et la conception de ces milieux devient une tâche très difficile à cause des besoins très divers auxquels l'aménagement des surfaces doit répondre tels que : l'agriculture, la vie sociale, les activités économiques et financières, la protection environnementale, etc.... En effet, cet aménagement est assuré par la discipline nommée : la planification urbaine. Cette dernière représente un ensemble d'instruments et de moyens permettant la mise en place des politiques sectorielles et la rationalisation de la gestion urbaine à travers la définition d'objectifs, de principes de développement et de projets d'aménagement. La planification urbaine constitue, donc une composante importante et un élément moteur dans le processus de développement des villes. Dans ce chapitre, nous allons présenter un aperçu sur le contexte de la planification urbaine en mettant l'accent sur ses différents aspects, objectifs, missions et principes fondamentaux.

## 1.2. Principes de la planification urbaine

L'expression "planification urbaine" a été définie et abordée selon différentes sources, parmi lesquelles nous citons:

- Le dictionnaire d'urbanisme et d'aménagement [Merlin et Choay 1988] définit la planification territoriale urbaine comme étant un « ensemble de méthodes, d'études, et de procédures financières ou juridiques permettant aux collectivités publiques de suivre l'évolution des milieux territoriaux urbains, d'identifier des hypothèses d'aménagement portant sur la nature, l'ampleur ainsi que la localisation des espaces à protéger et les développements urbains, par la suite œuvrer pour l'application des options acceptées ».
- Selon Henri Derycke [Derycke 1982], qui est un chercheur économiste, « pour assurer le développement des villes effacer l'urbanisme chaotique, la planification urbaine réduit les erreurs et les aléas que recèle le futur, dans une démarche prospective qui s'appuie sur les prévisions futures, elle définit des procédures et s'assigne des objectifs et désigne des instruments pour les atteindre. Ces plans d'urbanisme qui sont l'un des outils de la planification sont réducteurs des effets néfastes de l'urbanisation. La planification urbaine par ses instruments qui s'appelle le plan, la prospective, la recherche opérationnelle et la liste sans doute n'est pas close est un réducteur d'incertitude, il ne s'agit pas de deviner l'avenir, mais de contribuer à le construire. Elle traduit le passage de l'urbanisme d'autrefois au développement d'une nouvelle ville ».
- Selon Pierre Laborde [Laborde 1994] la planification urbaine est : « les plans, les institutions, les pratiques et les techniques qui cherchent à organiser la ville. La planification part de la demande sociale et non de la quête d'une quelconque ville idéale. Elle est apparue comme moyen d'empêcher l'anarchie urbanistique, de réaliser l'harmonie entre les besoins en logements, en emplois, en services, en circuits de distribution et en infrastructures de circulation ».
- Enfin, d'après Nigel Taylor [Ineichen 2007] « la planification territoriale urbaine n'est pas considérée comme étant une science qui revêt un caractère social. Il s'agit plutôt d'une action sociale, influencée par certaines valeurs morales, politiques et esthétiques pour aboutir à un environnement physique. Par conséquent ce genre d'action peut exister sous diverse aspects.

Pour déterminer et fixer le choix du lieu il, est impératif que les acteurs impliqués dans le même problème s'entendent sur un consensus élaboré en étroite collaboration ».

La planification urbaine n'est pas donc un acte en soi mais l'outil qui permettra la mise en œuvre du développement et l'expression d'un projet de société.

La planification urbaine couvre un ensemble de secteurs physiques et sociologiques, nous citons la démographie, l'emploi, l'habitat, l'utilisation des sols, les services publics, le budget, l'environnement, le transport...etc.[Henderson 1997]. La figure 1.1 illustre l'interaction entre ces différents secteurs. Robert Laurini [Laurini 2002] a mentionné que ces activités de planification ont les quatre qualités suivantes en commun:

- 1. La planification est orientée vers l'avenir** : Les décisions prises dans le processus de planification sont généralement faites pour affecter une condition future dans l'environnement.
- 2. La planification concerne la définition et l'évaluation des solutions alternatives** pour résoudre les problèmes : Cela est profondément ancré dans les théories de la planification rationnelle qui sous-tendent la pratique de la planification actuelle; elle est basée sur le principe selon lequel il est difficile de retenir une stratégie choisie et la considérer comme la bonne, si des solutions alternatives ne sont pas définies aux fins de comparaison et d'évaluation.
- 3. La planification est politique** : Chaque décision de planification publique a lieu dans un contexte politique. Il est important de réaliser que la majorité des activités de planification impliquent l'utilisation ou la réglementation des terrains sous une certaine forme, et que tout terrain appartenant à quelqu'un lui confère des droits et surtout le droit à une compensation raisonnable.
- 4. La planification a une responsabilité particulière** pour représenter les besoins des minorités : les personnes handicapées, pauvres et les groupes sous-représentés. Les planificateurs doivent prêter une attention particulière aux besoins de ces groupes dans le cadre de leur code professionnel de déontologie.

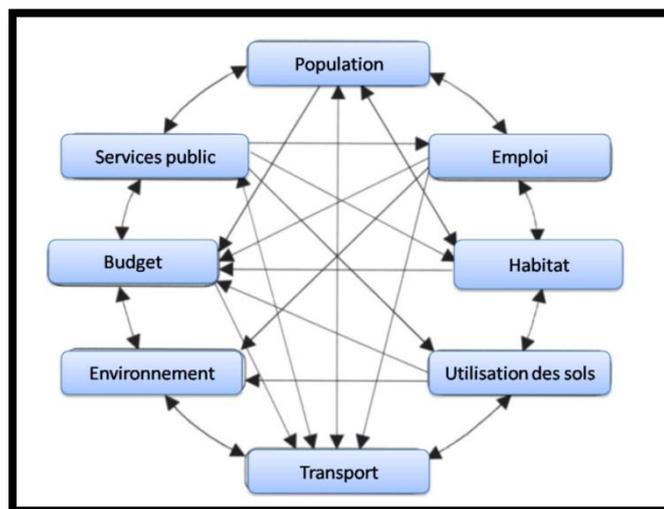


Figure 1.1. Quelques composantes du système urbain [Laurini 2002].

### 1.3. Les objectifs de la planification urbaine

La réalisation des objectifs de la planification urbaine est essentiellement une tâche politique. La planification urbaine traduit une volonté d'intégration et de coordination interdisciplinaire et de coopération entre différentes autorités concernées. Les objectifs de la planification territoriale urbaine sont résumés comme suit [Lévry et al. 2004] [CEMAT 2003]:

- La planification urbaine doit coordonner les différentes fonctions du sol.
  - La planification urbaine doit s'inscrire dans les perspectives du développement durable à savoir la durabilité, principe de prévention voire de précaution, le développement socio-économique équilibré des régions, la préservation environnementale et l'amélioration de la qualité de vie.
  - La planification urbaine doit coordonner les affectations, arbitrer les conflits d'utilisation, etc.
- En effet, la planification territoriale urbaine ne permet pas uniquement de légaliser les zones à bâtir, mais elle propose aussi une approche globale d'urbanisation, de transport, d'environnement, de nature, d'économie...

### 1.4. La planification urbaine en Algérie

L'Algérie est classée deuxième pays le plus grand géographiquement en Afrique. Toutefois elle occupe peu son espace. En effet il y a une armature urbaine déséquilibrée, une densification accentuée de la population dans la région du nord ainsi qu'un développement urbain important et désorganisé engendrant des problèmes de gestion et un manque dans le domaine de l'habitat, transport et des équipements.

Afin d'atteindre cette situation, l'Algérie a tracé une nouvelle vision de la planification urbaine visant plusieurs buts : [MATE 2004]

- proposer un développement durable et adapté du territoire urbain, en conciliant, l'équipe sociale, l'efficacité économique, l'émancipation de l'homme ainsi que la préservation environnementale.
- Comblent les entraves d'ordre géographique et naturels des territoires et des régions.
- Fournir une protection des populations et des territoires contre les dangers liés aux risques d'ordre technologique et naturels.
- Organiser la croissance des villes et favoriser le développement qualitatif des agglomérations.

#### 1.4.1. Les instruments de la planification urbaine

La politique de la planification urbaine en Algérie est menée au moyen d'un ensemble de schémas et de plans d'aménagement situés à différents niveaux d'échelles, dont on peut citer :

##### - Le Schéma National d'Aménagement du Territoire (S.N.A.T)

Il exprime la vision prospective de l'occupation du territoire national en liaison avec la stratégie du développement économique, social et culturel à long terme [CFU 2001]. Le schéma national fixe des paramètres fondamentaux déterminant :

- l'occupation rationnelle de l'espace national (avec stratégie de politique d'aménagement).
- la répartition planifiée de la population et des activités économiques, sociales et culturelles.
- la valorisation et l'exploitation des ressources naturelles.

- la mise en place coordonnée des réseaux d'infrastructures de base.
- la répartition spatiale des établissements humains et la location des grands équipements.
- la protection du patrimoine écologique national.
- la protection du patrimoine culturel.

#### **-Le Schéma Régional d'Aménagement du Territoire (S. R. A. T)**

C'est un instrument d'appui qui assure avec une plus grande précision la définition des choix et des actions d'aménagement du territoire à l'échelle régionale [MATE 2004]. Chaque SRAT détaille l'image prospective du territoire de la région et :

- définit les objectifs essentiels de valorisation du territoire régional.
- précise pour les sous-ensembles de cet espace les bases de la distribution équilibrée des activités et du peuplement.
- détermine la répartition et l'aménagement de la structure urbaine principale.
- détaille les programmes et organise la distribution des infrastructures et équipements structurants.
- définit les zones de forte solidarité interwilaya.
- fixe les stratégies de coordination des différentes initiatives en matière d'action économique.

#### **- Le Plan Directeur d'Aménagement et d'Urbanisme (P.D.A.U)**

C'est un outil permettant la gestion urbaine et la planification spatiale. Il permet de définir les orientations fondamentales de l'aménagement de la commune ou des communes qu'il couvre en prenant en considération les plans de développement et les schémas d'aménagement [CFU 2001].

#### **-Le Plan Communal de Développement (P.C.D)**

C'est un programme financier renouvelable chaque année, son rôle est à l'échelle locale. Le PCD a comme objectifs : le développement de l'industrie et de l'agriculture, l'amélioration des équipements sociaux.

#### **-Le Plan de Modernisation Urbaine (P.M.U)**

Son rôle est d'intervenir sur le tissu urbain existant à l'intérieur des villes de taille moyenne ou grande. Il permet la localisation des équipements et la répartition spatiale des investissements par le (PCD).

#### **-Le Plan d'Urbanisme Directeur (P.U.D)**

Le PUD nouveau instrument de gestion urbaine a été instauré pour faire face à la consommation abusive du sol urbain et de permettre la maîtrise de la croissance urbaine qui est due au développement rapide. Le PUD est un support de programmation et de spatialisation des différents programmes et investissements sur le court, moyen et long terme.

#### **- Le Plan d'Urbanisme Provisoire (P.U.P)**

C'est un plan d'urbanisme qui concerne les groupements qui n'ont pas bénéficié du plan d'urbanisme directeur .Ce plan a été promulgué à travers la loi communale. Il concerne les petits groupements urbains. Le PUP possède les mêmes perspectives et objectifs du PUD et la différence qui existe entre les deux plans réside dans la durée réduite pour l'élaboration du PUP ainsi que son approbation qui est à l'échelle locale loin de l'approbation ministérielle.

#### **-Le Plan d'Occupation des Sols (P.O.S)**

Le Plan d'Occupation des Sols permet de fixer de manière détaillée les droits d'utilisation et de construction des sols. Il est établi progressivement pour couvrir le territoire défini par le PDAU [CFU 2001]. Le POS permet de :

- fixer la forme urbaine, l'organisation ainsi que les droits de construction et d'utilisation des sols de façon détaillée.
- déterminer les instructions relatives à l'aspect extérieur des constructions.
- délimite les espaces publics et verts, les emplacements réservés aux ouvrages, publics et les installations à intérêt général ainsi que les tracés et les caractéristiques de voies de circulation.
- préciser les quartiers, rues, monuments, et sites à protéger, à rénover et à restaurer.
- localiser les terrains agricoles à préserver et à protéger.

### 1.4.2. Evolution de planification urbaine

La planification urbaine est devenue une nécessité devant le développement des métropoles, la rente pétrolière et la croissance démographique. En Algérie, le problème de la planification urbaine et son impact sur l'environnement a été abordé et soulevé dans la période allant de 1974 à 1977 (2ème plan quadriennal de développement), ainsi il a fallu instaurer et mettre en place des structures institutionnelles et des outils (tel que : PUD, PCD, PMU,...) pour la réalisation d'un tel projet. Jusqu'à la fin des années 80 c'est l'Etat qui centralisait l'urbanisme ; ensuite vers les années 90, l'intérêt pour la ville s'est dévoilé peu à peu et la planification urbaine est aperçue comme étant « une forme de mobilisation foncière pour les grandes opérations publiques, d'autorisations de construire à caractère provisoire sur les parcelles acquises par les promoteurs et par l'auto-construction » [Bouimaza 2005]. La libéralisation de l'économie dans les années 90 a fait balancer l'Etat algérien « D'Etat producteur à l'Etat organisateur du cadre, d'Etat pouvoir à un pouvoir d'Etat » [SNAT 1995], changeant ainsi ses objectifs en matière d'aménagement du territoire et par conséquent l'évolution d'instruments de la planification urbaine selon les différents contextes de développement économique. Malgré la panoplie de lois qui existe depuis des années 90 l'urbanisation suit toujours son cours [Rahmoun 2013].

### 1.4.3. Le développement durable

Le concept développement durable a été défini dans le rapport Bruntland [Vaillancourt 1998] en 1987. Il doit garantir l'amélioration de l'environnement, la préservation des ressources naturelles et la croissance naturelle. Il doit satisfaire les besoins présents tout en tenant compte des besoins des générations ultérieures.»

La définition proposée par la « stratégie mondiale de la conservation » est la suivante [Lequin 1992] : «le développement durable peut être défini comme étant la transformation de la biosphère et l'utilisation des moyens humains, financiers, vivants et non vivants afin de contenter les besoins de la population et améliorer la qualité de leur quotidien. Dans le but de garantir la continuité des ressources, il y a de prendre en considération les facteurs d'ordre social, écologique et économique, de la base des moyens vivants et non vivants ainsi que les bienfaits et les inconvénients à long et à court terme des autres solutions prévues».

Ainsi, le développement durable peut être considéré comme étant un développement qui prend en compte la satisfaction des besoins des générations présentes et futures. Il repose sur les trois piliers [Destais 2011] : le pilier écologique, économique, et social réaliserait à leurs

intersections ; un compromis doit alors se faire entre les dimensions écologiques, sociales et économiques [Wiliquet 2011].

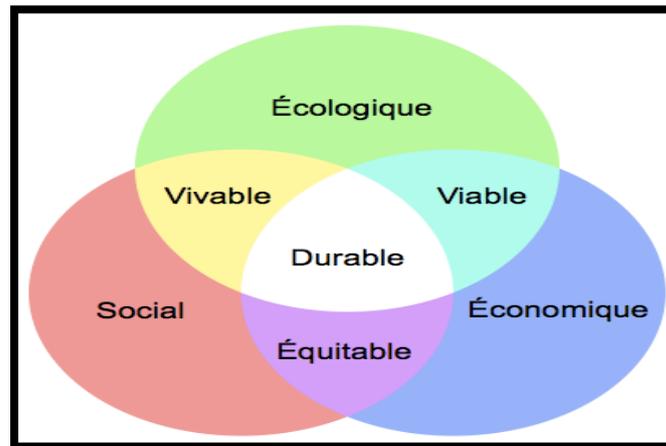


Figure 1.2. Les piliers du développement durable [Destais 2011].

En Algérie, la prise en charge de l'environnement sous ses différents aspects a été pendant fort longtemps méprisé : la pollution de l'air, la désertification, les changements climatiques, la remontée des eaux dans le Sud, l'appauvrissement de la diversité biologique, etc. Le souci, de mieux prendre en charge le secteur de l'environnement se fait sentir de manière plus sérieuse, plus pressante et plus éminente ces dernières années [Labeled 2007].

En effet, la création du « ministère de l'aménagement du territoire et de l'environnement » (MATE) en août 2000 est venue pour engager parallèlement à la politique nationale de l'aménagement du territoire une stratégie nationale de l'environnement [MATE 2004] permettant d'assurer les principes de développement durable en Algérie.

## 1.5. Les principes fondamentaux de la planification urbaine

La planification urbaine s'appuie sur trois grands principes : la coordination, la coopération et la participation : [MIAT 2005]

### -La coordination

Une des principales missions de la planification territoriale urbaine, elle consiste à coordonner les différentes demandes formulées en matière d'affectation des sols compte tenu de ses propres objectifs et des exigences des politiques sectorielles. La coordination a pour but de synchroniser les actions dans le temps en exploitant un référentiel temporel commun, et à gérer la cohérence des activités des individus en fonction de l'ensemble des actions.

### -La coopération

La coopération une action collective par laquelle des sujets contribuent à un même résultat. C'est aussi un moyen de dépasser les limites de l'action individuelle. En ce sens, les processus optant pour un mode de fonctionnement coopératif attendent en retour une minimisation des risques et une réduction de l'incertitude.

En qualité de politique transversale interdisciplinaire, la planification territoriale urbaine est souvent confrontée à divers problèmes résultant de la compartimentation entre compétences, niveaux d'organisation et secteurs. La coordination, telle qu'elle est décrite ci-dessus ne peut être garantie que si les acteurs concernés sont disposés à surmonter cette compartimentation et à s'intégrer dans une démarche commune.

### -La participation

La participation est un moyen indispensable pour la validation d'un projet en planification urbaine. Elle permet l'intégration des différents acteurs (individus ou groupes) pour aider le décideur à établir un choix et améliorer les procédures décisionnelles.

Les trois principes fondamentaux sur lesquels s'articule la planification urbaine sont modélisés sur la Figure 1.3.

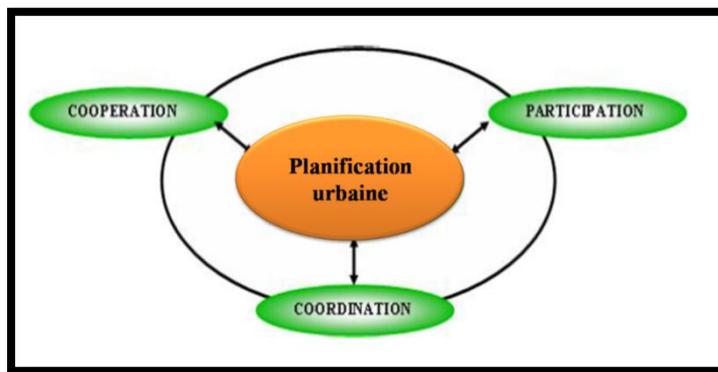


Figure 1.3. Principes de la planification urbaine [MIAT 2005].

## 1.6. L'évaluation dans le cadre de la planification urbaine

L'évaluation constitue un concept faisant l'objet de façons de faire, de disciplines et de qualificatifs variés comme le mentionnent Jacob et Varone [Jacob et Varone 2003] : « L'évaluation est un objet polysémique traversé par plusieurs courants. Il s'agit d'une pratique où sont appliquées diverses démarches des sciences sociales. Les méthodes et techniques pluridisciplinaires sont empruntées à la sociologie, aux sciences économiques et politiques, à la psychologie, etc. Les définitions de cette pratique ainsi que ses finalités sont nombreuses et reflètent les débats théoriques en cours mais aussi des pratiques concrètes très différentes ».

Malgré les différentes approches en évaluation, des caractéristiques générales peuvent tout de même lui être attribuées. À ce titre, l'évaluation peut être vue comme étant une démarche à la fois objective et subjective [Risse 2004]. La notion d'objectivité représente une démarche permettant d'assister le planificateur le plus objectivement possible, d'inventorier, de classer et d'arranger l'information nécessaire à un choix [Voogd et Faludi 1985]. Par contre, la notion de subjectivité constitue une démarche impliquant un jugement ou encore « l'évaluation d'une valeur » [Risse 2004]. Elle repose sur la comparaison des mérites ou encore des avantages et désavantages de différentes options dans le but d'aboutir à des conclusions logiques [Lichfield et al. 1975].

L'évaluation dans le cadre de la planification urbaine est vue donc, comme une démarche décisionnelle [Risse 2004]. Elle conduit à des recommandations contribuant à orienter une décision [Voogd et Faludi 1985]. L'évaluation, alors n'est pas synonyme de la prise de décision mais elle assiste cette dernière en fournissant aux participants au processus de planification urbaine des informations factuelles nécessaires à la prise de décision [Lichfield et al. 1975]. Enfin, l'évaluation peut être considérée comme une démarche d'accompagnement et d'enrichissement du processus de la planification urbaine [Risse 2004].

L'évaluation dans le domaine de la planification comprend l'évaluation des politiques, des plans, des projets urbains, des contextes, des acteurs et des parties concernées [Oliveira et Pinho 2008]. Dans le contexte de notre thèse, nous nous intéressons plus particulièrement à l'évaluation des projets urbains. Nous présentons dans les parties suivantes la notion des projets urbains ainsi que la prise de décision pour leur évaluation dans la cadre de la planification urbaine.

### **1.6.1. Notion du projet urbain**

La notion de « projet urbain » apparaît dans les années 1970 comme une réponse à de nombreux bouleversements [Tomas 1998]. Selon Masboungi [Masboungi 2002], le projet urbain « organise un territoire afin d'en améliorer l'usage, la qualité, le fonctionnement, la dynamique économique et culturelle et les relations sociales. Il doit assurer à tous l'accessibilité à l'espace public, à l'habitat, aux équipements, aux transports ; se préoccuper de la qualité des espaces publics, de l'architecture, des paysages, de l'environnement naturel, de la mise en valeur du patrimoine ; servir les enjeux du développement durable, avec une utilisation économe de l'espace tout en garantissant le fonctionnement des infrastructures, ainsi que des réseaux de transports et de distribution. Il ne se réalise que grâce à un engagement politique fort, pour lutter contre le laisser-faire et les coupures urbaines physiques, fonctionnelles et sociales »

Le projet urbain, alors représente une stratégie « d'agir en formation qui marque un moment de transition entre la manière traditionnelle de penser l'urbanisme et une nouvelle approche, moins figée et plus ouverte aux transformations et aux débats » [Ingallina 2001].

#### **1.6.1.1. Types de projets urbains**

Selon Ascher [Ascher 1991], il existe trois types de projets urbains: le projet urbain politique, opérationnel et architectural :

- Le projet urbain politique regroupe les projets dits «de territoire», «de ville», « d'agglomération», «de développement» ; il couvre tout le territoire communal ou intercommunal [Arab 2004]. Ce type est compris comme une intention de ville résultant d'une réflexion stratégique, indispensable pour promouvoir des solutions alternatives face aux mutations profondes des villes, aux risques sociaux et environnementaux qu'elles semblent engendrer [Rabinovich 2008].
- Le projet urbain opérationnel est considéré comme étant un ensemble d'opérations urbaines dont les fonctions sont multiples et munies d'une certaine étendue persistant au moins une dizaine d'années. Ces opérations impliquent plusieurs intervenants publics et privés et suscitent une conception et une gestion d'ensemble [Ascher 1995]. Ce sont des opérations urbaines de

nature complexe regroupant des projets diversifiés dans un plan, un programme ainsi que des formes d'ensemble [Ascher 1995].

- Le projet urbain architectural est réservé à l'approche du « design urbain », en lien avec sa réhabilitation faite par les architectes-urbanistes [Rabinovich 2008]. Il intéresse un ou plusieurs bâtiments. C'est un processus urbanistique et architecturale inséré. Le projet urbain architectural est étroitement lié aux éléments de la forme urbaine environnante.

### 1.6.1.2. Acteurs du projet urbain

Le projet urbain est vu comme une démarche, à la fois pensée et spatialisée. Il existe plusieurs groupes d'acteurs qui ont un rôle important pour l'avancement du projet urbain, nous citons : [Dind et Cunha 2011]

- **Les acteurs impliqués** : il s'agit des acteurs qui interviennent dans la marche du projet urbain et on a :

- **Les décideurs** : ce sont les élus impliqués, les chefs de service des administrations qui ont un pouvoir de décision sur le projet urbain, soit sur le plan de financement, d'orientation stratégique ou bien sur le plan de la validation. Ils doivent donner une orientation au projet urbain et fournir les moyens financiers qu'il faut.

Il est à signaler l'importance de l'association de tous ces derniers au pilotage du projet urbain et ce même à titre symbolique de manière à dévier éventuellement des entraves sur le plan politique.

- **Les opérationnels**: il s'agit des acteurs qui s'occupent de la conduite réelle du projet : Le responsable du projet, les coopérateurs des administrations concernés dans la structure opérationnelle (équipe du projet) et même les délégués des associations. Le rôle des opérateurs est de bien conduire le projet urbain, en atteignant les objectifs définis par les décideurs. Il s'agit là également de faire intervenir les différents services impliqués dans une optique transversale du projet. Il faut que le projet ait un sens dans le cadre des missions de ces collaborateurs qui doivent avoir les moyens qu'il convient (objectif net, temps, moyens financiers) afin de bien conduire les diverses tâches du projet.

- **Les acteurs intéressés**: il s'agit des acteurs qui interviennent de temps à autre sur l'un ou l'autre aspect du projet urbain : On a :

- **Les mandataires**: leur mission est d'agir sur un aspect ou un autre du projet urbain : Architectes, consultants stratégiques, animateurs des démarches participatives, sociologues.

Il faut souligner l'importance de l'évaluation du bénéfice de l'implication des collaborateurs des administrations dans l'accomplissement de leurs tâches.

- **Les associations**: sont des partenaires essentiels dans la gestion des projets dans la mesure où elles apportent de très importantes informations sur l'environnement local et aussi elles suggèrent dans la majorité du temps une vision pointue qui peut compléter sur des thématiques particulières comme : La protection de l'environnement, la gestion de la mobilité, les commerces, la vie du quartier, les équipements sociaux ...Elles ont entre autres un rôle dans l'alimentation de la réflexion sur le projet urbain.

- **Les acteurs touchés** : il s'agit des acteurs qui sont directement touchés par le projet urbain et on a :

- **Les propriétaires:** ils peuvent être des propriétaires privés voulant simplement valoriser leur parcelle ou bien il peut s'agir de propriétaires institutionnels comme les assurances et les caisses de pension. Le comportement adopté vis-à-vis des conduites de négociation : Entente optimale pour la coopération qui peut aller jusqu'au partenariat public- privé, immobilisme par rapport aux sollicitations de l'administration ou un comportement de méfiance dans le but de faire juger par la loi les différences entre les intérêts privés ou publics .

- **Les habitants:** ne composent pas un bloc d'acteurs identiques. Ils se différencient par leur comportement (d'accord ou non avec le projet urbain), par leur degré de contribution (présents ou non), par les problématiques qu'ils soutiennent. La représentativité constitue le challenge essentiel : De quelle manière peut-on connaître que les décisions prises par les habitants concernés par le projet urbain reflètent-elles la position de tous ? Une des voies à favoriser est de fixer des « acteurs-clés » qui représentent des intérêts différents.

Il est essentiel de signaler qu'il y a des relations entre les divers acteurs. D'après Laurini [Laurini 2002] ces relations peuvent être :

- en coopération : dans ce cas, les acteurs travaillent ensemble pour résoudre les mêmes types de problèmes, ils sont d'accord sur la même solution, ils partagent les mêmes plans d'action, et ils mettent en commun leurs ressources et moyens.
- en conflit : dans ce cas, les acteurs ont des intérêts divergents dans la résolution du problème; le conflit peut être un conflit d'objectifs ou de signification ; le plus fréquent est celui des objectifs.
- en négociation : dans ce cas, les acteurs savent que leur intérêt est de travailler ensemble pour un objectif commun; ils acceptent partiellement ou temporairement de partager certaines ressources pour résoudre un problème.

### 1.6.1.3. Principales étapes du projet urbain

La littérature du domaine a montré que plusieurs processus ont été proposés pour établir un projet urbain. Le déroulement d'un projet urbain s'articule autour d'un processus d'étapes séquentielles et complémentaires classées en trois niveaux [Laouar 2005] (Figure 1.4) :

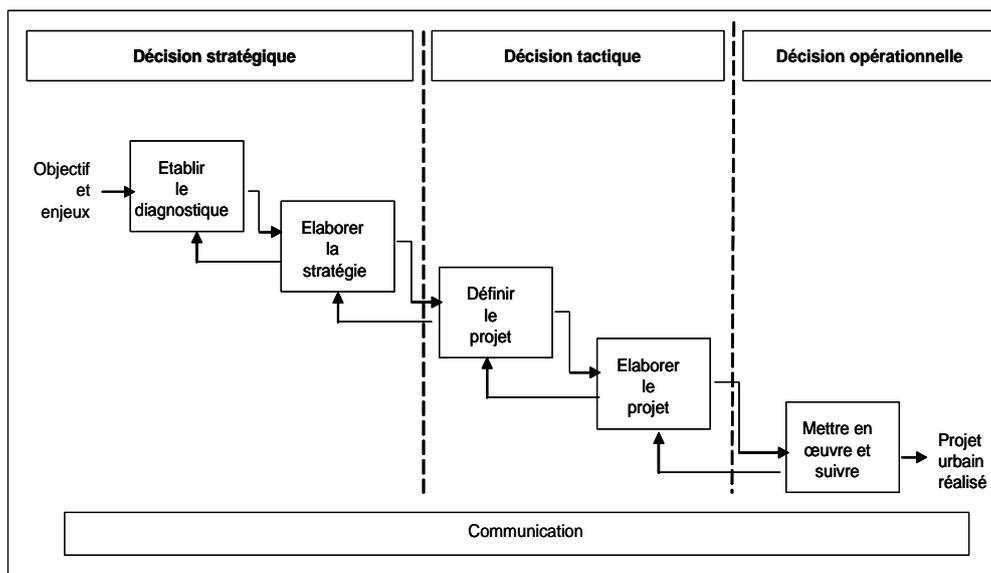


Figure 1.4. Processus de conduite d'un projet urbain [Laouar 2005].

### A. Décision stratégique

La décision stratégique vise à déterminer les grandes missions, les politiques générales, ainsi que les objectifs poursuivis. Le niveau stratégique permet la définition des grandes orientations à long terme. Les étapes incluses dans ce niveau sont :

• **Etablir le diagnostic** : le diagnostic permet d'apporter des précisions :

- les recherches effectuées dans le domaine de l'urbanisme,
- les informations et les études disponibles.
- la consultation des divers intervenants,

Il est nécessaire pour définir le contexte, la problématique et le périmètre d'intervention.

• **Elaborer la stratégie** : c'est l'étape la plus difficile à franchir et la plus sensible politiquement. En effet elle oblige les responsables politiques à s'avancer, à se positionner vis-à-vis d'un avenir par définition incertain. L'adoption d'un scénario se fait selon l'ensemble des actions qu'il comporte, des principaux objectifs ainsi que des ressources financières dont il a besoin.

### B. Décision tactique

La décision tactique ou administrative consiste à répartir de manière rationnelle les ressources à disposition. Ce niveau contient les objectifs dont la raison d'être est de permettre de se rapprocher d'un objectif. Il concerne donc la définition des objectifs qui doivent poursuivre les actions concrètes pour concourir aux objectifs stratégiques. Les étapes incluses dans ce niveau sont :

• **Etablir le projet** : la phase habituelle qui consiste à mettre en relief (documents graphiques, textes ...) la méthode définie précédemment en matière de grandes politiques d'aménagement et de développement social, économique, écologique, ...

• **Elaborer le projet** : à la suite du diagnostic, les objectifs et les grandes lignes du projet urbain se formalisent dans le cadre d'un plan de référence. Ce document est composé de plans synthétiques traduisant les grandes options de développement du projet.

### B. Décision opérationnelle

Cette décision qui est liée au processus concret de travail nécessaire à la réalisation des missions. Ce niveau met en œuvre des actions concrètes et des décisions prises sur des données à grande précision. Les étapes incluses dans ce niveau sont :

• **Mettre en œuvre et suivre** : lors de l'élaboration du cahier des charges, il est nécessaire de bien clarifier les missions du concepteur urbain. Cette étape commence à partir de la collecte des informations jusqu'à la détermination claire des divers domaines (secteurs) étudiés ainsi que leur organisation par rapport au temps.

• **Communication du projet** : c'est une phase capitale, mais elle est souvent négligée. Cette étape permet de communiquer le projet urbain à tous les acteurs du projet urbain (institutions, entreprises, associations et habitants) ce qui permet d'obtenir des critiques, des avis, des propositions, une excellente efficacité sur plan administratif et politique du projet urbain, et une bonne acceptation des propositions d'actions figurant dans le projet adhésion par les forces vives de l'agglomération.

## 1.6.2. La prise de décision en planification urbaine

L'approche de prise de décision pour la planification territoriale urbaine est multidisciplinaire, elle prend en considération la diversité bio-physique des humains ainsi que la diversité des territoires urbains [Graillot et Waaub 2006]. Dans le large choix des procédés de décision, quelques décisions ont la spécificité de maintenir un lien étroit avec le territoire urbain, on parle donc de décisions dont l'objet est un projet territorial ; ce dernier a des effets caractéristiques sur le territoire : des impacts directs qui renvoient à l'emprise du projet (exemple :L'espace d'un endroit naturel qui bénéficie d'une protection ou une superficie prise par une construction) ou alors indirects qui font un écho aux conséquences de la décision (exemple :Méfaits acoustiques, un effet sur la vision, des conséquences d'ordre économique, un accroissement des mouvements des gens) [Pelletier 2009].

Pour une bonne assimilation de la prise de décision dans le cadre de la planification urbaine, il est exigé une analyse approfondie ; il s'agit alors d'apporter des réponses aux questionnements ci-après: [Fouchet et Lopez 2000]

- Qu'est-ce qui caractérise une opération de la planification urbaine ?
- Comment la situer?, et par ailleurs comment se présente une décision en la matière ?

Cependant, ces questions ne sont pas neutres. En effet, une décision en matière de planification urbaine est aussi une décision publique, elle est prise sur un espace donné, concerne différents acteurs et résulte d'une participation plurielle d'acteurs politiques, administratifs, techniques, financiers, etc [Fouchet et Lopez 2000].

Selon Laurini [Laurini 2002], la prise de décision urbaine comporte trois éléments essentiels : les décideurs en conflit, le territoire urbain et le système d'aide à la décision (DSS) lui-même. La Figure 1.5 montre l'interaction et la relation entre ces différents éléments : à l'aide des simulations, les différents décideurs peuvent percevoir comment la ville évolue en fonction de leurs actions. Une fois les évolutions simulées semblent satisfaisantes, le plan d'action est susceptible d'être trouvé.

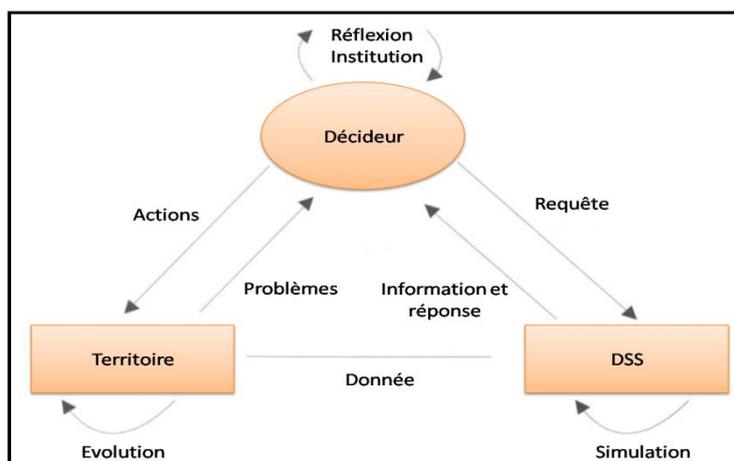


Figure 1.5. Relations entre les décideurs, le territoire et le DSS [Laurini 2002].

L'aide à la décision territoriale intervient donc sur des processus basés sur la récolte, l'analyse et l'échange d'informations qui permettent aux acteurs concernés par la décision de construire, de renforcer ou de modifier leurs préférences [Labeled 2007] comme illustré par la Figure 1.6.

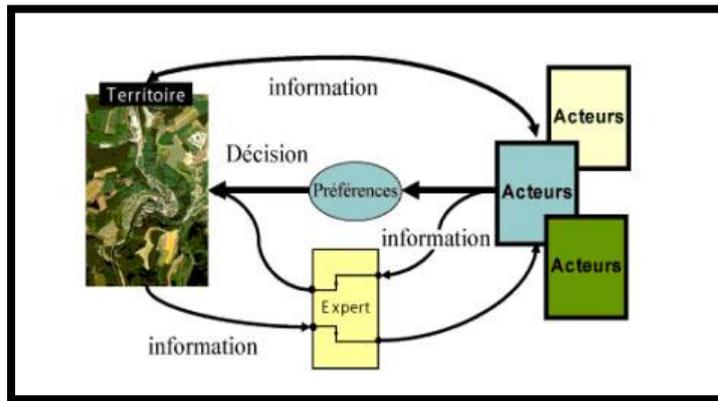


Figure 1.6. Processus d'aide à la décision territoriale [Labeled 2007].

Selon Pelletier [Pelletier 2009], on associe à la décision territoriale trois espaces distincts : (1) l'espace d'intervention qui correspond à l'emprise objective du projet urbain, (2) l'espace des conséquences qui couvre les impacts réels ou supposés et finalement, (3) l'espace des acteurs, soit le territoire défini par les lieux de résidence (ou de siège social) des acteurs ayant un rôle dans le processus décisionnel. Certaines décisions portent sur des projets urbains de grande envergure aux conséquences tout aussi importantes et n'impliquant que des acteurs régionaux. D'autres décisions ont pour objet des aménagements très localisés mais dont les conséquences se manifestent au-delà de ces espaces et n'impliquant que des acteurs régionaux. Enfin, certaines décisions s'intéressent à l'aménagement d'espaces restreints, mais dont les conséquences sont régionales et impliquent un large éventail d'acteurs locaux et régionaux voire même internationaux.

Donc, le plus souvent, les acteurs ont des positions différentes, du fait d'intérêts ou de systèmes de valeurs divergents. Or, ces différents avis nécessitent d'être pris en considération pour la décision publique. Nous allons par conséquent trouver une entente, en fonction des différents modes, en nous dirigeant vers un arrangement ou un consensus. Nous passons donc par une étape de négociation, au cours de laquelle les intervenants établissent des arguments et les communiquent entre eux dans le but d'appuyer leur position ou bien combattre celle de l'autre [Ferrand 1997]. Dans la plupart des projets urbains, la négociation constitue le cœur du processus de décision. Alors le paradigme de la négociation sera abordé plus en détail lors du chapitre 3.

## 1.7. Conclusion

Le domaine de la planification territoriale urbaine est un domaine très riche et vaste. Il inclut de nombreuses et différentes problématiques. L'informatisation de ces problèmes apparaît une tâche très intéressante telle qu'une meilleure gestion du volume d'information, évaluation d'un grand nombre de projets urbains, visualisation des cartes géographiques des futurs bâtiments, traitement et analyse de données...

En effet, grâce à ces outils, les intervenants dans un projet de planification urbaine bénéficieront d'une aide importante à la prise de décision.

---

Cependant, dans un projet de ce type, différents acteurs sont intégrés, ces derniers ont souvent des buts contradictoires qui les mettent généralement en situation de débat et de conflit. En plus, le même projet peut avoir des solutions diverses selon le lieu et le contexte.

De ce fait, le développement d'une application en planification urbaine pour résoudre la problématique d'évaluation des projets urbains nécessitera l'intégration de différentes disciplines, à savoir l'aide à la décision multicritère et la négociation qui prennent en considération la multitude des critères des différents projets urbains et les points de vue.

# Chapitre 2

## Aide à la décision multicritère

*«L'objectif de ce chapitre est d'explicitier les concepts de décision, d'aide à la décision et l'aide à la décision multicritère en mettant l'accent sur ses diverses problématiques et méthodes puis de présenter les différentes solutions décisionnelles en planification urbaine. »*

---

### Sommaire

2.1. Introduction.....	36
2.2. Aide à la décision.....	36
2.2.1. Définition de la décision.....	36
2.2.2. Définition de l'aide à la décision.....	36
2.2.3. Les différentes approches d'aide à la décision.....	37
2.2.4. Les différentes étapes du processus d'aide à la décision.....	37
2.2.5. Système d'aide à la décision (DSS).....	39
2.3. Aide à la décision multicritère.....	40
2.3.1. De monocritère vers le Multicritère.....	41
2.3.1.1. Le monocritère.....	41
2.3.1.2. Le multicritère.....	42
2.3.1.3. Les frontières entre le monocritère et le multicritère.....	44
2.3.2. L'aide à la décision multicritère.....	44
2.3.2.1. Formulation d'un problème multicritère de décision.....	44
2.3.2.2. Les approches multicritères.....	44
2.3.2.3. Les étapes d'aide à la décision multicritère.....	46
2.3.3. Les problématiques multicritères.....	47
2.3.4. Les méthodes multicritères.....	50
2.3.4.1. Les Méthodes décisionnelles multictère de l'approche du critère unique de synthèse....	50

2.3.4.2. Les méthodes de sur-classement selon l'approche du sur-classement de synthèse .....	54
2.3.4.3. Les méthodes interactives selon l'approche du jugement local interactif .....	61
2.3.5. Algorithme de sélection d'une méthode d'aide à la décision multicritère .....	61
2.3.5.1. Caractérisation des inputs .....	62
2.3.5.2. Caractérisation des outputs .....	63
2.4. Etat de l'art : les systèmes décisionnels en planification urbaine.....	64
2.5. Conclusion.....	66

## 2.1. Introduction

Que se soit dans la vie domestique ou professionnelle, l'homme a et sera toujours amené de prendre des décisions. Ces dernières, bien qu'elles soient prises le plus souvent par une seule personne, sont le plus souvent la résultante d'une interaction entre les préférences des participants d'un groupe (Conseil d'administration, gouvernement, ...). Ces intervenants (ou acteurs) font des choix qui les amènent à exécuter ou non une action, la faire d'une manière ou d'une autre. Ces choix peuvent être sans importance ou, à l'opposé, influencer sur la vie d'un collectif (famille, société, pays, ...). Ces décisions, bien qu'elles soient la résultante d'une réflexion (ou processus de décision), sont également le résultat de la confrontation permanente entre ces différents acteurs et ne sont pas toujours les meilleures [Aupaix 2010].

Tout au long de ce chapitre, nous clarifions les outils méthodologiques qui seront employés dans la démarche d'aide à la décision multicritère constituant la base de notre approche qui sera présentée au chapitre 5. À la section 2.2 nous présenterons les étapes de la méthodologie d'aide à la décision qui est utilisée pour formaliser le processus décisionnel. À la section 2.3, nous nous intéresserons plus en détail à l'aide à la décision multicritère. Enfin à la section 2.4, nous présenterons les approches décisionnelles proposées en planification urbaine.

## 2.2. Aide à la décision

La décision consiste à choisir une alternative basée sur la mise en œuvre d'un ensemble de ressources ou bien à déterminer un ensemble de buts en prenant en considération un ou plusieurs facteurs afin d'évaluer des solutions. La prise de décision ainsi que son application dans toute structure représentent des objectifs importants pour celle-ci ; alors chaque structure dépend forcément du type de la décision prise en son sein par les différents décideurs (collectifs ou individuels). L'ensemble de décisions prises sont généralement fortuites ou basées sur des essais passés alors qu'en réalité ce genre de démarches s'applique uniquement pour les situations familières. Lorsqu'on est face à des nouveaux cas, la procédure de décision sera plus délicate ainsi que le cadre du décideur devient plus compliqué et se développe d'une manière plus rapide. La détermination des données adéquates pour la prise de décision est une tâche difficile dans le domaine d'aide à la décision, alors il est indispensable d'employer des systèmes d'aide à la décision (DSS) pour l'évaluation d'un problème décisionnel, des différentes alternatives ainsi que leurs influences.

### 2.2.1. Définitions de la décision

En effectuant une recherche sur dictionnaire le petit Larousse, nous avons trouvé que la décision est vu comme étant un « acte par lequel quelqu'un opte pour une solution, décide quelque chose ; résolution, choix ». Il y a plusieurs définitions qui ont été proposées pour expliquer le concept de décision :

- Une décision est « une initiative prise afin de résoudre un problème ou bien répondre à une modification de l'environnement, cela veut dire, pour solutionner un problème posé à l'organisation ou à l'individu [Lévine et Pomerol 1989] ».
- D'après Roy et Bouyssou [Roy et Bouyssou 1993], la décision « est généralement représentée par le fait qu'un décideur effectue librement un choix entre plusieurs alternatives à un moment

donné dans le temps ».

- Selon Mintzberg [Mintzberg 1979] : « une décision, qu'elle soit résultante d'un travail de groupe ou individuelle, elle est déterminée par l'engagement dans une action, ce la veut dire qu'il existe une volonté explicite d'agir ».
- Décider, « c'est choisir entre plusieurs alternatives (actions) parmi celles qui sont possibles. L'action sélectionnée ne permet pas l'effectuation complète ou parfaite des buts ; elle représente uniquement la meilleure solution possible dans des conditions données » [Simon 1983].

### 2.2.2. Définitions de l'aide à la décision

Zoller et Béguin [Zoller et Béguin 1989] ont défini l'aide à la décision comme suit : « aider à la décision, c'est donner l'information qui permet une évaluation la plus proche des différentes possibilités envisagées et une prospective plus appropriée des solutions attendues de façon à ce que la négociation se fait autour d'une table et non sur le terrain ».

En faisant référence aux travaux de Roy [Roy et Bouyssou 1993], l'aide à la décision représente « une activité d'un analyste qui utilise des modèles bien expliqués mais pas totalement formalisés, qui essaye de trouver des réponses aux interrogations posées par un décideur dans une démarche de décision, et enfin qui intervient dans le processus de mise en œuvre de toutes les options qui ont été retenues ».

Selon cette hypothèse, on déduit que l'aide à la décision ne répond pas directement au problème décisionnel mais elle se limite à quelques interrogations posées aux décideurs, ces questions représentent leurs préoccupations. Donc, il est nécessaire de répondre aux préoccupations des décideurs, tout en répondant à la préoccupation finale (problème décisionnel).

### 2.2.3. Les différentes approches d'aide à la décision

D'après la définition présentée précédemment, nous avons déduit que l'aide à la décision est une activité basée sur des approches qui représentent la réalité de façon formelle ou bien explicite. Dans cette partie nous intéressons aux approches coexistant fondée sur ces modèles. Il existe différents types d'approches ou modèles qui sont représentés ci-dessous : [Mammeri 2013]

#### • **Modèle normatif**

Les modèles normatifs ce sont des modèles qui ont été construits en se basant sur une rigueur économique [Fishburn 1970] [Luce et Raiffa 1957]. Généralement le décideur n'est pas appelé à construire ces modèles. Dans ce cas, la validation des résultats obtenus par ce modèle normatif repose sur leur harmonie avec les principes de la rationalité économique.

#### • **Modèle descriptif**

Contrairement au modèle précédent, ces modèles [Poulton 1994][Humphreys et al. 1983] sont de type empirique et non pas théorique. Ils sont le plus souvent basés sur des informations ou des observations existantes, c'est-à-dire d'appliquer au problème traité, des solutions qui sont jugées très bien adaptées dans des problèmes similaires.

#### • **Modèle prescriptif**

Le modèle prescriptif [Belton et Stewart 2002] [Keeney 1992] est différent des deux modèles vus précédemment. Il n'est pas basé sur des données existantes. Alors, l'homme d'étude collecte

et structure les données pour élaborer ce modèle. Lors de l'élaboration du modèle, il n'est pas nécessaire que le décideur intervient. Néanmoins, il peut intervenir lors de la validation.

### • **Modèle constructif**

Dans ce modèle [Bouyssou et al. 2000] [Genard & Pirlot 2002], nous avons à construire le problème et la solution au même temps. Dans ce cas, le décideur est appelé lors de l'élaboration et la validation du modèle. Les modèles sont alors basés sur les informations des différents décideurs. Le rôle de l'analyste ici, est l'explicitation ou la formalisation des données.

## **2.2.4. Les différentes étapes du processus d'aide à la décision**

Au cours cette section, nous présentons les diverses étapes du processus décisionnel. D'après A. Tsoukiàs le processus décisionnel se déroule selon les étapes suivantes : [Tsoukiàs 2008]

### **1- Représenter le problème décisionnel**

Cette première phase consiste à analyser le processus décisionnel pour déterminer :

- les participants au processus décisionnel,
- leur mission dans le processus décisionnel,
- le participant (ou les participants) avec qui l'analyste interagit,
- les objectifs de chacun des participants concernant le problème décisionnel.

Certains objectifs sont plus importants que d'autres ou dépendent les uns des autres. Alors, pour que l'aide à la décision soit pertinente, Tsoukiàs [Tsoukiàs 2008] a proposé de déterminer l'importance de chacun de ces objectifs et les possibles dépendances qui existent entre eux

### **2- Formuler le problème décisionnel**

Selon Roy [Roy 1985], dans cette étape, nous identifions avec le décideur l'ensemble des actions ou des alternatives possibles. Cet ensemble sera noté l'ensemble  $A$ . Selon Tsoukiàs [Tsoukiàs 2008], il existe différents types de traitements que nous pouvons réalisés sur  $A$  :

- a- La définition de  $A$  d'une façon formelle sans évaluation.
- b- La construction d'un modèle dans le but de la description d'un sous-ensemble d'alternatives dans un ensemble de points de vue.
- c- Le partitionnement de l'ensemble  $A$ .

Le troisième traitement (partitionner l'ensemble  $A$ ) est appelé problématique [Roy 1985], il est souvent en aide multicritère à la décision. Il existe trois manières de diviser l'ensemble  $A$  ou trois problématiques d'après Roy [Roy 1985] : problématique de choix, de tri, et de rangement. Nous présenterons ces problématiques en détail dans la section 2.3.3.

### **3- Construire le modèle d'évaluation**

Après la détermination de l'ensemble des actions (Alternatives)  $A$  ainsi que les préoccupations du décideur, l'analyste construit un modèle à travers lequel il va répondre au problème décisionnel posé et donner ses consignes. Pour l'explicitation et la formalisation d'un modèle d'évaluation, l'homme d'étude construit les « dimensions » qui représente une information sur  $A$ , elle permet de déterminer l'ensemble  $A$  ; les différents « critères » qui permettent d'évaluer l'ensemble des éléments de  $A$ , et enfin les « méthodes d'évaluation » qui représentent les outils nécessaires pour effectuer le traitement souhaité sur l'ensemble  $A$  dans le but d'apporter une

solution au modèle.

#### 4- Construire la recommandation finale

Si les résultats fournis par les méthodes d'évaluation sont harmonieux avec le modèle, cela ne signifie pas qu'ils sont cohérents avec les objectifs (préoccupations) des participants au processus décisionnel. Alors avant de donner une quelconque consigne, l'analyste doit vérifier en utilisant une analyse de sensibilité que les résultats obtenus par la méthode d'évaluation ne sont pas altérés par des fluctuations non significatives des données employées pour l'élaboration du modèle. Egalement, il doit effectuer une interprétation des résultats obtenus par la méthode d'évaluation en prenant en considération toutes les propositions sur le modèle, les données et l'évolution du processus décisionnel à partir desquelles il a été produit.

Nous avons résumé les étapes du processus décisionnel par le schéma (Figure 2.1) suivant :

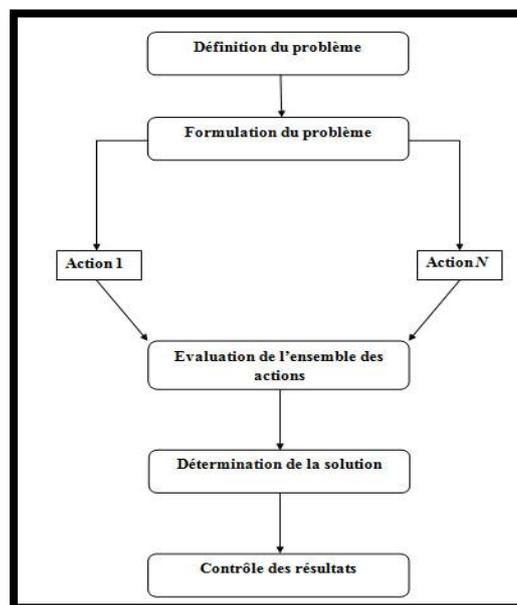


Figure 2.1. Les étapes du processus d'aide à la décision.

#### 2.2.5. Système d'aide à la décision (DSS)

Un système d'aide à la décision (DSS) représente un mécanisme qui permet de soutenir et accompagner le décideur tout au long de son processus de prise de décision. En effet, un processus de décision ne peut pas être entièrement automatisable [Pomerol 1992]. Le DSS est aperçu comme étant un Système de Traitement de l'Information [Lévine et Pomerol 1989], son rôle est l'extraction et la fourniture des données nécessaires au décideur mais cela est insuffisant pour la caractérisation d'un DSS et le distinguer des autres systèmes d'information conventionnels. Il est important de rajouter qu'un DSS peut être considéré comme un résolveur de problème.

Un DSS est composé de trois modules : le module de dialogue, base de données et modèle. Comme le montre la Figure 2.2, le module dialogue est interconnecté avec les deux autres modules (le module base de données et le module modèle).

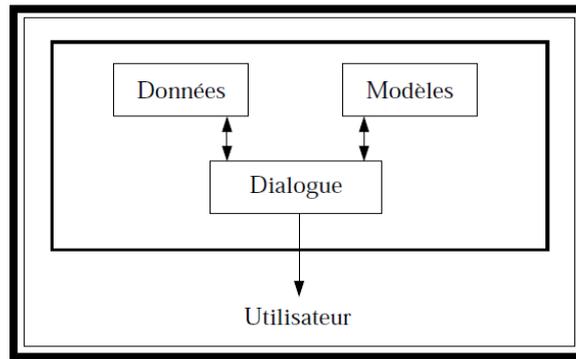


Figure 2.2. Structure d'un DSS [Sprague 1987].

Le rôle de chaque module est présenté ci-dessous : [Pelletier 1999]

- Le module « dialogue » permet au décideur d'accéder aux fonctions de calcul et aux données, il lui permet également de communiquer les résultats de manipulation réalisée via les interfaces dirigées par ce module. Les représentations des résultats sont moins favorisés que les échanges ce qui permet au décideur de réaliser sa recherche heuristique et son contrôle dans de meilleures circonstances.
- Le module « Base de données » fournit un stockage permanent et temporaire des données. Il offre aussi la sauvegarde des informations volatiles (représentent les résultats obtenus pendant la phase du traitement des données) et l'effacement de celles-ci selon le besoin de l'utilisateur.
- Le module « Modèle de données », ce module comprend une multitude de procédures de calcul et des modèles tels que les modèles de prédiction et de simulation, les outils de programmation mathématiques, les procédures de représentation des données, et les modèles financiers et de calcul standards. Ces outils sont mis à la disposition de l'utilisateur afin qu'il les utilise pour la réalisation des traitements standards des données.

### 2.3. Aide à la décision multicritère

Dans la partie précédente, nous avons vu que la prise de décision constitue l'activité principale et essentielle des décideurs. Cette activité devient de plus en plus complexe, car ces décideurs cherchent à intégrer dans leurs décisions plusieurs facteurs de nature contradictoire. Les situations de choix sont nombreuses où les actions sont évaluées sur la base de plusieurs objectifs ou critères.

La problématique de décision multicritère est souvent présente dans la vie pratique. En effet la détermination de la meilleure solution représente un grand défi. L'aide à la décision multicritère s'est développée afin de fournir des concepts, des approches, des modèles et des méthodes qui ont pour but d'aider le décideur à décrire, évaluer, ranger, choisir ou rejeter un ensemble d'actions, pouvant être exercées sur des alternatives qui peuvent être des candidats, des produits ou des projets. Cet exercice est basé sur l'évaluation à l'aide de notes ou scores, de valeurs, d'intensité de préférence, et ce, en fonction d'un ensemble de critères. Ces derniers peuvent représenter divers aspects tels que les objectifs, les buts, les cibles, les valeurs de préférence, les degrés d'aspiration et les fonctions d'utilité.

### 2.3.1. De monocritère vers le Multicritère

Dans cette partie, nous allons présenter le paradigme monocritère et le multicritère ainsi que les frontières entre eux.

#### 2.3.1.1. Le monocritère

Un problème de décision monocritère est un problème du type :

$$\text{Optimiser } \{f(x) / x \in A\}$$

Cette expression représente clairement les trois étapes de la modélisation: [Schärli 1985]

(1) La définition de l'espace  $A$  de solution.

(2) La modélisation des préférences du décideur se fait au moyen d'un critère d'évaluation  $f(x)$ . Pour tout  $x \in A$ ,  $f(x)$  est un nombre réel représentant soit un profit, soit un coût.

(3) Le processus d'investigation mathématique consiste à optimiser  $f(x)$  (maximiser ou minimiser) sur l'espace  $A$ . Il s'agit souvent d'un algorithme, plus ou moins compliqué selon le cas.

Selon la définition, le monocritère est un problème d'optimisation. Plusieurs problèmes de recherche opérationnelle sont de ce type et nous citons : la programmation linéaire, non linéaire, dynamique, la théorie des graphes et des réseaux, l'optimisation combinatoire, la théorie des jeux, les problèmes de localisation, les problèmes de transport, la théorie de file d'attente, la gestion des stocks, les problèmes d'ordonnancement, la gestion de la production,...

D'après Schärli, les problèmes monocritères ne sont pas adaptés aux traitements de la réalité humaine. Prenons comme exemple une personne souhaitant s'acheter un téléphone portable. Si cette personne raisonne selon un seul critère à savoir le prix, elle prendra le téléphone portable le moins cher ; alors qu'en réalité il existe plusieurs types de téléphone portable du moins cher au plus onéreux : une personne préférerait un téléphone tactile sans s'intéresser au prix, une autre préférerait un téléphone portable avec clavier coulissant....

Le monocritère possède certaines propriétés : [Vilain 2007]

#### Propriété 1: problème bien posé

Dans le paradigme monocritère, la notion de solution optimale a en général un sens. Il s'agit d'une solution  $x'$  telle que:  $f(x') \geq f(x)$ ,  $x \in A$  C'est une solution optimale que l'on soumettra au décideur dans le cas d'une problématique de choix.

#### Propriété 2: relation de dominance ( $I, P$ )

Le critère  $f(x)$  permet de différencier les actions de  $A$ . Il implique de façon naturelle une relation de dominance ( $I, P$ ) sur les éléments de  $A$  pris deux à deux et tels que:

$$\begin{cases} f(a) > f(b) \leftrightarrow aPb \\ f(a) = f(b) \leftrightarrow aIb \\ f(a) < f(b) \leftrightarrow aRb \end{cases}$$

Où  $P$  désigne la préférence,  $I$  l'indifférence et  $R$  l'incomparabilité.

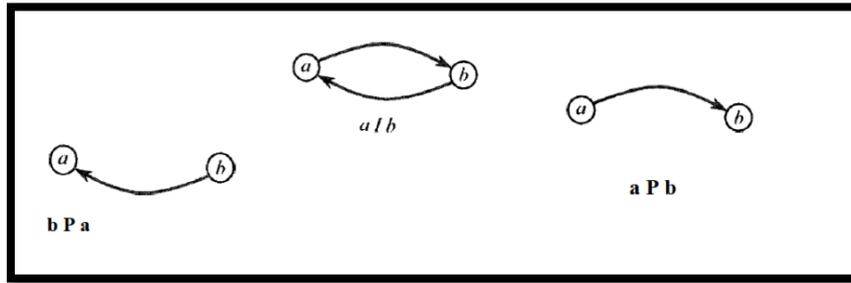


Figure 2.3. Relation de dominance.

**Propriété 3: préordre complet**

La relation  $(I, P)$  n'est pas une relation quelconque. Elle permet aussi de ranger les actions de la moins bonne à la meilleure, avec éventuellement la présence d'ex aequo. On parle dans ce cas de préordre complet.

Ce préordre sera soumis au décideur en cas de problématique de rangement. Le préordre est complet car toutes les actions sont comparables deux à deux.

**Propriété 4: transitivité**

Les relations  $I$  et  $P$  sont transitives. En effet:

$$\begin{cases} aPb, bPc \rightarrow aPc \\ aIb, bIc \rightarrow aIc \end{cases}$$

**2.3.1.2. Le multicritère**

Un problème de décision multicritère est un problème du type : [Perny 1998]

$$\mathbf{Max} \{ f_1(x), f_2(x), \dots, f_k(x) / x \in A \}$$

On suppose ici que les  $k$  critères  $f_j(x), j=1,2,\dots,k$  sont des applications de  $A$  sur l'ensemble des nombres réels. Ces critères sont en général exprimés dans des unités distinctes. On considère ici le cas où le décideur souhaite maximiser tous les critères, ce qui facilite l'énoncé des propriétés. Ce cas n'est cependant pas restrictif et les résultats énoncés s'étendent directement au cas plus général dans lequel certains critères sont à maximiser et d'autres à minimiser.

Le paradigme multicritère possède certaines propriétés :

**Propriété 1: problème mal posé**

Un modèle multicritère correspond à une certaine réalité économique, industrielle ou autre, le décideur souhaitant optimiser plusieurs critères simultanément. Conformément aux vœux du décideur, le problème est donc bien posé, économiquement bien posé. Malheureusement, et c'est ici la difficulté majeure de l'approche multicritère, ce problème n'admet généralement pas de solution  $x'$  telle que: [Landry 1998]

$$f_j(x') \geq f_j(x), \forall x \in A, \forall j = 1, 2, \dots, k$$

Dès lors, il représente un problème sans solution, donc mal posé mathématiquement.

Bien entendu, s'il existe une solution optimale sur l'ensemble des critères (cas rare), cette solution sera soumise au décideur dans le cas d'une problématique de choix. Sinon, il faudra se contenter d'une solution de compromis. Chaque méthode multicritère s'efforce de proposer les meilleures solutions de compromis possibles. [Landry 1998]

### Propriété 2: relation de dominance ( $I, P, R$ )

Les  $k$  critères induisent de façon naturelle la relation de dominance suivante sur  $A$  : [Vansnick 1995]

$$a P b \leftrightarrow \begin{cases} f_j(a) \geq f_j(b), \forall j = 1, \dots, k \\ \exists h: f_h(a) > f_h(b) \end{cases}$$

$$a I b \leftrightarrow f_j(a) = f_j(b), \forall j = 1, \dots, k$$

$$a R b \leftrightarrow \begin{cases} \exists h: f_h(a) > f_h(b) \\ \exists h': f_{h'}(a) < f_{h'}(b) \end{cases}$$

Où  $P$  désigne la préférence,  $I$  l'indifférence et  $R$  l'incomparabilité, c'est-à-dire l'absence de relation entre  $a$  et  $b$ . la relation de dominance, si l'on exclut  $R$ , est donc une relation partielle. Observons que la préférence et l'indifférence sont fondées sur l'unanimité des points de vue (de tous les critères).

### Propriété 3: Pareto optimalité (efficacité)

Une solution  $a \in A$  est appelée Pareto optimale ou efficace si elle n'est dominée par aucune autre solution. Si  $E$  désigne l'ensemble des solutions efficaces, le décideur est logiquement invité à décider parmi les solutions de  $E$  dans le cas d'une problématique de choix [Mareschal 2014].

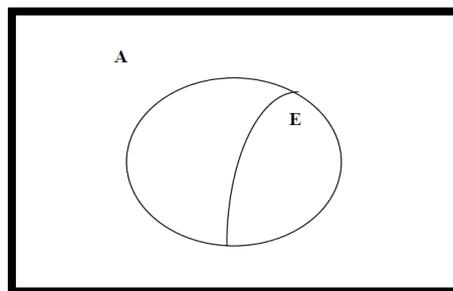


Figure 2.4. Solutions efficaces.

Il faut cependant observer que la détermination de l'ensemble  $E$ , à laquelle se limite d'ailleurs un certain nombre de méthodes multicritères, ne résout en général pas le problème. En effet, d'une part  $E$  contient souvent un nombre élevé de solutions (le cas n'est d'ailleurs pas rare où toutes les solutions sont efficaces) et d'autre part les solutions efficaces sont souvent de nature opposée: quand une solution est bonne selon un critère, elle est généralement moins bonne sur d'autres.

Nous sommes donc confrontée à une situation fort différente du cas monocritère dans lequel une solution optimale s'impose (ou éventuellement plusieurs solutions optimales indifférentes). Il importe donc de ne pas confondre les notions d'optimalité (cas monocritère) et de Pareto-optimalité (cas multicritère). Dans ce dernier cas, décider sur  $E$  reste un problème délicat: les solutions efficaces, bien qu'incomparables, ne sont pas indifférentes.

#### **Propriété 4: transitivité**

Certaines méthodes multicritères respectent la transitivité de la préférence ou de l'indifférence, d'autres non. Ceci dépend de la technique proposée par chaque méthode pour la recherche de solutions de meilleur compromis. Observons à ce sujet qu'en cas d'évaluation multiple, il n'y a pas lieu d'imposer la propriété de transitivité [Roy et Bouyssou 1993].

#### **2.3.1.3. Les frontières entre le monocritère et le multicritère**

Savoir choisir en s'appuyant sur un unique critère est une bonne chose. Or, dans la réalité, les choix auxquels nous sommes confrontés ne se résument pas la plupart du temps à classer ces actions selon un seul critère. Reprenons encore une fois l'exemple du téléphone portable. Pour le choix d'un téléphone on n'a pas qu'un seul critère ; il existe d'autres critères sur lesquels juger un téléphone par exemple : la qualité et la longueur de la diagonale de l'écran, le fait qu'il possède ou non un clavier physique, son prix, ... On voit bien sur cet exemple que choisir son téléphone est bien plus compliqué que de faire un choix en se basant sur un seul critère. De plus, cette multiplicité vient avec un problème récurrent, celui-là même qui fait la difficulté du choix multicritère : les critères peuvent conduire à des choix conflictuels. En effet, si on ne considère que la qualité générale et le prix des téléphones, on se rend compte bien vite que plus un téléphone est de bonne qualité, plus il coûte cher. Comment faire lorsqu'une utilité plus grande sur un attribut implique une plus faible sur un autre ?

#### **2.3.2. L'aide à la décision multicritère**

Selon Maystre [Maystre et al. 1994], « l'aide à la décision multicritère est vue comme étant une analyse qui a pour objectif l'explicitation d'une famille harmonieuse de caractéristiques permettant l'appréhension des diverses conséquences d'une alternative (action) ».

Vincke [Vincke 1989] a défini l'aide à la décision multicritère ainsi: « L'aide multicritère à la décision vise, comme son nom l'indique, à fournir à un décideur des outils lui permettant de progresser dans la résolution du problème de décision à plusieurs points de vue, souvent contradictoires, doivent être pris en compte ».

L'aide à la décision multicritère apporte des explications à une catégorie de problèmes où un ensemble de caractéristiques qualitatives et quantitatives est pris en compte ; ces caractéristiques sont généralement hétérogènes, conflictuelles et considérées d'inégale importance [Laaribi 2000].

##### **2.3.2.1. Formulation d'un problème de décision multicritère**

Selon Vansnick [Vansnick 1990], la formulation d'un problème de décision multicritère est représenté selon le modèle suivant " $A, A/F, E$ " où:

- **A** représente un ensemble fini ou infini des alternatives (actions) potentielles.
- **A/F** représente un ensemble fini de critères qui sont souvent conflictuels ou contradictoires, à partir desquels les alternatives sont évaluées;
- **E** représente un ensemble d'évaluations des performances des alternatives selon chaque critère.

### 2.3.2.2. Les approches multicritères

Afin de faire face à un problème d'aide à la décision multicritère, il existe diverses démarches. On trouve :

- **la démarche de Roy** [Roy 1990 b] qui est basée sur « une approche de bas vers le haut » (bottom-up). Cette approche identifie « les conséquences » provenant de la mise en œuvre de l'ensemble des actions qui sont structurées en « dimensions » par la suite en « axes de signification » à travers desquels les « critères » sont construits.

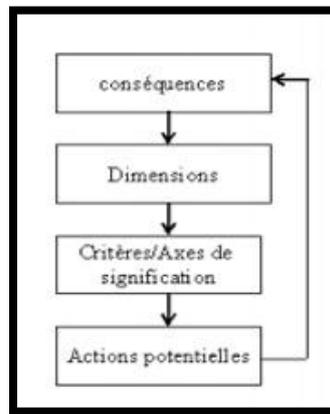


Figure 2.5. Démarche bottom-up de Roy [Roy 1990 a].

- **la démarche de Keeney and Raiffa** [Keeney 1992] est basée sur « l'approche du haut vers le bas » (top-down). Cette approche construit une organisation hiérarchique où le niveau premier de celle-ci représente « l'objectif global » qui sera divisé en « sous-objectifs » jusqu'à ce qu'on arrive à un niveau mesurable que l'on nomme « un niveau d'attributs ».

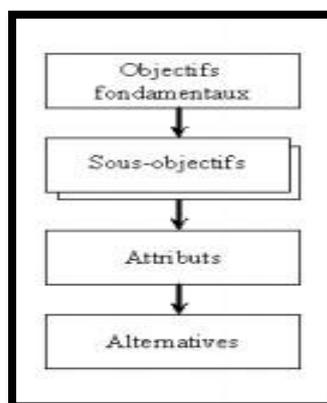


Figure 2.6. Démarche top-down de Keeney [Keeney 1992].

• **la démarche de Laaribi** [Laaribi 2000] est une approche intermédiaire qui, selon lui, est de portée générale. Cette approche démarre par « une situation de décision quelconque » où un ensemble « d'objectif à atteindre » est dégagé. En prenant en considération ces « objectifs », un ensemble de « points de vue » qui exprime les classes des critères sera dégagé. Egalement, ces « objectifs » permettent la définition « des scénarios (ensemble d'actions fragmentées) ou bien des actions globales ». L'ensemble de « points de vue » représente le plus souvent les « critères » permettant « l'évaluation » « des actions ou des scénarios ». Une fois l'évaluation effectuée, on procédera à l'investigation par une procédure multicritère d'agrégation appropriée pour parvenir à une recommandation.

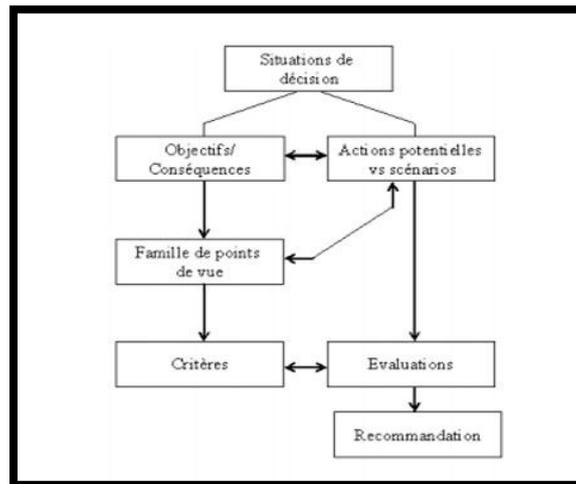


Figure 2.7. Démarche intermédiaire de Laaribi [Laaribi 2000].

### 2.3.2.3. Les étapes d'aide à la décision multicritère

Brans [Brans 1993] propose une démarche en trois étapes. Dans un premier temps, les alternatives potentielles sont identifiées. Dans un deuxième temps, les préférences du décideur sont modélisées, ce qui permettra de différencier les actions potentielles. Pour finir, une procédure d'agrégation, est définie ou choisie afin de fournir des éléments de réponse au problème de décision.

**1- Identifier les alternatives (actions) potentielles:** cette étape permet la définition d'un ensemble d'actions qui devrait être examiné. Cet ensemble peut être soumis à multitude de révisions où des actions vont apparaître ou bien disparaître.

**2- Modéliser les préférences du décideur:** comparer les actions en se basant sur leurs conséquences est une tâche difficile à cause de l'incertitude, l'imprécision, ainsi que la mauvaise définition de ces conséquences. Pour cela la notion d'attribut (critère) a été introduite. Le rôle des critères est la comparaison des actions entre elles. Un critère représente une fonction  $C$  qui porte sur des valeurs réelles définies sur l'ensemble des alternatives potentielles. Cette fonction est définie de telle sorte que deux alternatives  $a_1$  et  $a_2$  puissent être comparées en se basant sur les nombres  $C(a_1)$  et  $C(a_2)$ . Les préférences du décideur sont alors modélisées par un ensemble de critères. Un poids peut aussi être associé à chaque critère, selon l'importance du critère par rapport aux autres. L'évaluation des alternatives se fait sur la famille des critères. Un tableau de performances qui présente les résultats de cette évaluation sur chaque critère donne

une vue de l'ensemble des alternatives. Les critères sont sur les colonnes et les alternatives sont sur les lignes. La case  $C_j$  ( $Ca_i$ ) donne la performance de l'alternative  $a_i$  selon le critère  $C_j$ .

		Critères					
		$C_1()$	$C_2()$	...	$C_j()$	...	$C_k()$
Alternatives	$a_1$						
	$a_2$						
	...						
	$a_i$						
	...						
	$a_n$						
		$w_1$	$w_2$	...	$w_i$	...	$w_k$
		Poids					

Tableau 2.1. Matrice d'évaluation Alternatives/Critères.

**3- La procédure d'agrégation:** une procédure d'agrégation multicritère doit être définie afin de répondre à la problématique de décision posée en s'appuyant sur le tableau des performances qui caractérise les alternatives potentielles à évaluer.

Il existe différentes méthodes d'agrégation multicritère que nous allons présenter par la suite.

### 2.3.3. Les problématiques multicritères

La problématique en aide à la décision multicritère représente la manière d'envisagement de l'aide à la décision [Roy 1985]. Elle exprime les différents termes selon lesquels l'analyste ou le décideur a défini le problème et détermine le type de la solution qu'il souhaite avoir. Quatre problématiques ont été définies par Roy [Roy 1985], on trouve: la problématique de choix, de rangement et de tri qui ont un objectif perspectif, et la problématique de description qui est plus modérée dans son but car elle est à destination descriptive. Nous présenterons donc ci-après ces quatre problématiques en détail.

#### • La problématique de choix ( $P.\alpha$ )

Cette problématique consiste à aider à « choisir » une meilleure alternative (action) ou bien à établir d'une « procédure de sélection ». Mathématiquement, la problématique  $\alpha$  pose le problème en matière de recherche/exploration d'un « sous-ensemble  $A'$  de  $A$  » qui contient la ou les meilleures actions avec un cardinal faible [Ginting 2000].

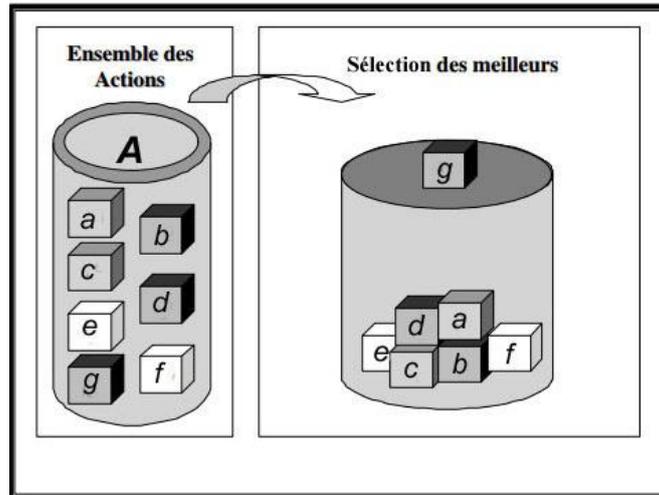


Figure 2.8. La problématique du choix [Ginting 2000].

### • La problématique de tri ( $P.\beta$ )

Cette problématique consiste à aider à « trier » les alternatives (actions) en suivant certaines normes ou à établir une « procédure d'affectation » en utilisant des catégories d'actions préalablement élaborées. Cette problématique recherche un « protocole d'affectation » qui distribue les alternatives (actions) de l'ensemble  $A$  sur les différentes catégories prédéfinies. Cependant nous pouvons avoir une alternative qui appartient à plusieurs classes (catégories) ou bien qui n'appartient à aucune des classes [Ginting 2000].

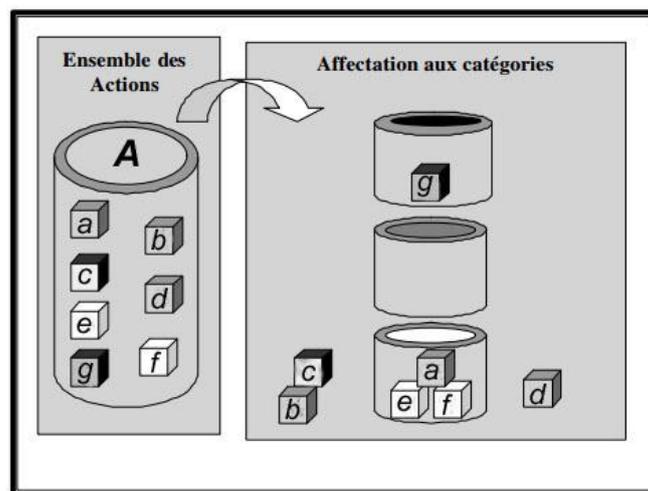


Figure 2.9. La problématique du tri [Ginting 2000].

### • La problématique de rangement ( $P.\gamma$ )

Cette problématique permet de « ranger » les alternatives (actions) de la meilleure action à la moins bonne. La situation idéale de cette problématique est d'avoir un ordre complet. Mais, les différents critères sont de nature contradictoire ou conflictuelle pour cela généralement on présente au décideur un ordre partiel. [Mousseau 1993].

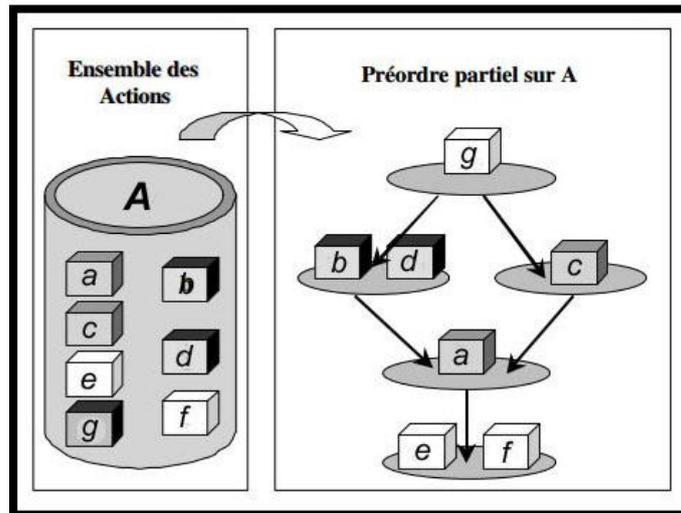


Figure 2.10. La problématique du rangement [Ginting 2000].

### • La problématique de description ( $P.\delta$ )

Cette problématique consiste à « décrire » les alternatives (actions) ainsi que leurs conséquences et non pas les comparer entre elles comme dans les problématiques précédentes. Dans ce cas, la procédure d'investigation est cognitive et il n'existe pas une prescription [Roy 1985].

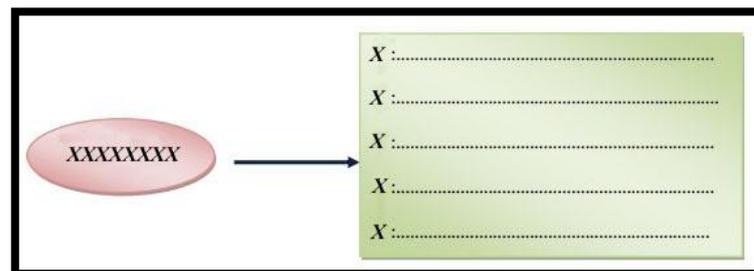


Figure 2.11. Problématique de la description [Azzabi 2010].

Les différentes problématiques multicritères sont résumées sur le tableau ci-après.

Problématique	Procédure d'investigation	Prescription	Résultat	Objectif
$(P.\alpha)$	Sélection	Sous ensemble de $A$	Un choix ou une procédure de sélection	sélectionner un sous ensemble aussi restreint que possible de $A$ contenant les meilleures actions.
$(P.\beta)$	Segmentation	Partition de $A$	Un tri ou une procédure d'affectation	Affecter chaque action de $A$ à une et une seule des catégories définies selon de normes préétablies.
$(P.\gamma)$	Classement	Ordre partiel sur $A$	Un rangement ou une procédure de classement	Ordonner les actions de $A$ de la meilleure à la moins bonne.
$(P.\delta)$	Cognitive	Aucune prescription	Une description ou une procédure cognitive	Décrire les actions et leurs conséquences.

Tableau 2.2. Les différentes problématiques de décision [Roy 1985].

### 2.3.4. Les méthodes multicritères

Dans cette partie nous nous intéressons à la troisième composante du processus décisionnel dans le cas d'aide multicritère à la décision: les méthodes ou les procédures d'évaluation. Il y a deux grandes classes de méthodes d'aide à la décision multicritère. La première classe représente les « méthodes d'aide à la décision multicritère qui utilise un attribut (critère) unique de synthèse », cette classe agrège les performances d'une action en un seul critère. La deuxième classe est celle des « méthodes d'aide à la décision multicritère de sur-classement » cette classe compare les différentes actions par paires. Nous étudierons dans cette partie ces deux classes de méthodes d'aide à la décision multicritère et par la suite nous présenterons une autre classe de méthodes qui est la famille des « méthodes d'aide à la décision multicritère à base de règles ». Ces catégories de méthodes se présentent comme suit :

#### 2.3.4.1. Les méthodes décisionnelles multicritère de l'approche de l'attribut unique de synthèse

Selon Roy cette classe est la plus classique. Les méthodes d'aide à la décision multicritère qui appartiennent à cette classe portent souvent le nom de « méthodes d'aide à la décision multicritère d'agrégation complète ». Elles consistent à agréger l'ensemble des critères, de manière à obtenir une fonction critère unique qui synthétise cet ensemble. Ainsi, cette fonction à optimiser, qui peut être par exemple une fonction de valeur ou d'utilité, agrège les différentes préférences de chacun des critères [Martel 1999]. En d'autres termes, ceci revient, selon Schärli, à transformer un problème multicritère en un problème monocritère. Cependant, il est important de ne pas confondre l'analyse multicritère et l'analyse monocritère. Roy souligne, à ce sujet, que même lorsqu'une analyse multicritère s'achève par l'agrégation d'un ensemble d'attributs en un critère unique, celle-ci diffère de l'analyse monocritère. Il considère que cette dernière prend a priori comme référence un critère unique en faisant l'économie de la détermination de l'ensemble des critères pertinents eu égard au contexte décisionnel en présence [Kazi 2009].

##### I. La méthode SMART (Simple Multi Attribute Rating Technique) [Edwards 1971]

###### a. Notion de base de la méthode

Cette méthode a été proposée par Edwards en 1971. Elle utilise « la forme additive » pour agréger les évaluations sur les divers critères. En utilisant la « la forme additive », nous obtiendrons de bonnes approximations. Les phases de la méthode SMART sont les suivantes :

- Phase 1 : trier les différents attributs (critères) du meilleur au moins bon selon leurs importances.
- Phase 2 : définir les « poids » des attributs (critères).
- Phase 3 : effectuer une normalisation des « coefficients d'importances » relative entre 0 et 1 en additionnant les « coefficients d'importance » puis en divisant chaque « poids » par cette somme.
- Phase 4 : calculer la distance de chaque alternative sur chacun des attributs ( $u_j(a_i)$ ).
- Phase 5 : définir la valeur de chacune des alternatives en utilisant la « somme pondérée »

$$U(a_i) = \sum_{j=1}^n \pi_j \cdot u_j(a_i), \text{ où } i = 1, 2, \dots, m$$

- Phase 6 : ranger les alternatives suivant un ordre décroissant de  $U(a_i)$ .

### b. Outils logiciels

- Criterium Decision Plus 3.0
- Decide right

## II. La méthode MAVT (Multiple Attribute Value Theory) [Keeney et Raifa 1976]

### a. Notion de base de la méthode

Cette méthode a été proposée par Keeney et Raifa en 1976. Le principe de la méthode est le suivant: chaque décideur cherche à maximiser une « fonction  $V = V[g_1, \dots, g_n]$  » pour l'agrégation de tous les critères. Cette méthode permet de construire une « fonction de valeur partielle » pour chaque critère. La meilleure alternative sur un critère obtiendra une « valeur partielle » égale à 1 et la moins bonne alternative obtiendra la « valeur partielle » égale à 0. En questionnant le décideur, on construit au fur et à mesure les « fonctions de valeur partielle ». Par la suite en fonction des « caractéristiques des préférences » du décideur, on construit la « fonction de valeur  $V$  ». Les phases de la méthode MAVT sont les suivantes :

- Phase 1 : effectuer une évaluation de chaque alternative (action) selon chacun des critères en construisant la « matrice multicritère ».

	$g_1$		$g_j$		$g_n$
$a_1$	..	..	..	..	..
	..	..	..	..	..
$a_i$	..	..	$g_j(a_i)$	..	..
	..	..	..	..	..
$a_m$	..	..	..	..	..
	$v_1(g_1)$	..	$v_j(g_j)$	..	$v_n(g_n)$

Tableau 2.3. Matrice multicritère.

- Phase 2 : pour chaque critère, élaborer des « fonctions de valeur partielle »

	$v_1(g_1)$		$v_j(g_j)$		$v_n(g_n)$
$a_1$	..	..	..	..	..
	..	..	..	..	..
$a_i$	..	..	$v_j(g_j(a_i))$	..	..
	..	..	..	..	..
$a_m$	..	..	..	..	..

Tableau 2.4. Matrice des fonctions de valeur partielle.

- Phase 3 : déterminer les poids des différents attributs (critères).
- Phase 4 : pour chaque alternative (action), calculer « l'évaluation globale » en employant une « forme d'agrégation appropriée ».

$$V[g_1(a_i), \dots, g_n(a_i)] = f\{v_1[g_1(a_i)], \dots, v_n[g_n(a_i)]\}$$

Par exemple : décomposition de la fonction de valeur en une forme additive

$$V[g_1(a_i), \dots, g_n(a_i)] = \sum_{j=1}^n \pi_j v_j g_j(a_i)$$

• Phase 5 : définir l'alternative qui maximise la valeur  $V$  c'est-à-dire l'alternative qui a eu le plus grand score (c'est la meilleur alternative).

**b. Outils logiciels**

- Web Hipre
- LDW Software

**III. La méthode MAUT (Multiple Attribute Utility Theory) [Keeney et Raifa 1976]**

**a. Notion de base de la méthode**

Cette méthode a été proposée par Keeney et Raifa en 1976. Elle a le même principe que la méthode précédente (MAVT). Mais, cette méthode s'applique uniquement dans le cas où les évaluations des alternatives par rapport aux critères sont aléatoires. Donc ici nous parlons d'une « fonction d'utilité » et non pas d'une « fonction de valeur ». Cette fonction considère séparément les critères et détermine « l'utilité » dégagée de chacun de ces critères. Alors « l'utilité »  $V_A(x_1, x_2, \dots, x_n)$  pour l'alternative  $A$ , dégagée sur les critères  $1, 2, \dots, N$  se décompose comme suit :

$$V_A(x_1, x_2, \dots, x_n) = \sum_{j=1}^n (x_j)$$

**b. Outils logiciels**

- LDW Software

**IV. La méthode AHP (Analytic Hierarchy Process) [Saaty 1980]**

**a. Notion de base de la méthode**

Cette méthode a été proposée par Saaty en 1980 ; elle permet de représenter un problème décisionnel sous une forme hiérarchique tout en montrant les interconnexions qui existent entre les différents éléments de ce problème, par la suite elle procède à comparer paire par paire tous les éléments de la structure hiérarchique, et à la fin elle détermine les différentes priorités des alternatives. Les phases de la méthode AHP sont les suivantes :

• Phase 1 : effectuer une décomposition du problème décisionnel en une structure hiérarchique qui sera composée d'un ensemble d'éléments interconnectés. Au sommet de cette structure, on trouve le « but », et dans les sous niveaux, « les éléments qui contribuent pour arriver à ce but ». Le dernier niveau de l'hiérarchie est celui des « alternatives (actions) ».

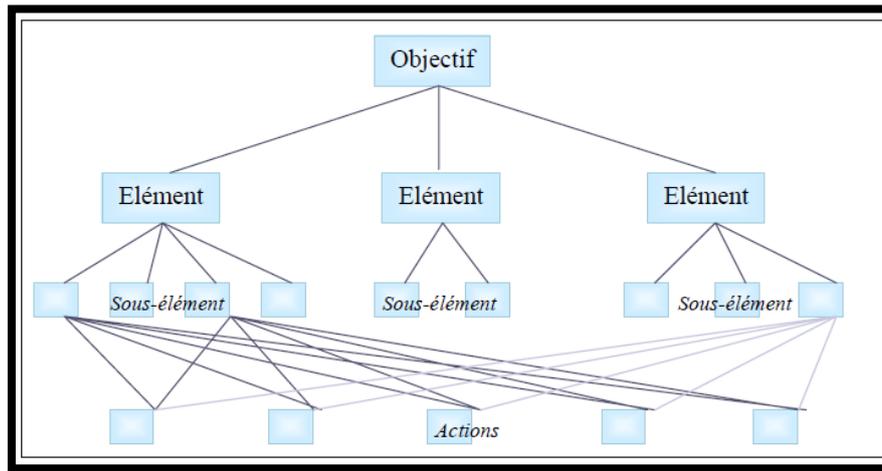


Figure 2.12. La structure hiérarchique AHP [Hammami 2003].

- Phase 2 : comparer par paires les éléments de chaque niveau de la structure hiérarchique par rapport à un élément du niveau au-dessous. Cette phase construit un ensemble de tableaux de comparaisons où les valeurs de ces tableaux sont calculées en utilisant « la transformation des jugements en valeurs numériques selon l'échelle de Saaty », tout en prenant en considération la réciprocité :

$$P_c(E_A, E_B) = \frac{1}{P_c(E_B, E_A)}$$

Degré d'importance	Définition
1	Importance égal à deux éléments
3	Faible importance d'un élément par rapport à un autre.
5	Importance forte ou déterminante d'un élément par rapport à un autre.
7	Importance attestée d'un élément par rapport à un autre.
9	Importance absolue d'un élément par rapport à un autre.
2, 4, 6, 8	Valeurs intermédiaires entre deux appréciations voisines.
Réciprocité	Si l'élément $i$ se voit attribuer l'un des chiffres précédents lorsqu'elle est comparée à l'élément $j$ , $j$ aura donc la valeur inverse lorsqu'on la compare à $i$ .

Tableau 2.5. Échelle de Saaty pour la méthode AHP [Saaty 1980].

- Phase 3 : définir l'importance liée aux éléments de la hiérarchie par le calcul des « vecteurs propres » qui correspondent aux « valeurs propres maximales » des tableaux de comparaisons.
- Phase 4 : procéder à la vérification de « la cohérence des jugements ».
  - Tous d'abord, calculer « IC qui est l'Indice de Cohérence ».

$$IC = \frac{\lambda_{max} - n}{n-1}$$

«  $\lambda_{max}$  » représente « la valeur propre maximale » qui correspond au tableau des comparaisons par paires et  $n$  représente le nombre des éléments de la hiérarchie comparés.

➤ Par la suite, calculer le « RC qui est le Ratio de Cohérence » comme suit :

$$RC = 100 \times \frac{IC}{ACI}$$

$ACI$  représente « l'indice de cohérence moyen » résultant de la génération aléatoirement des « tableaux de jugement » ayant des tailles similaires.

Dimension de la matrice	1	2	3	4	5	6	7	8	9	10
Cohérence aléatoire ( $ACI$ )	0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49

Tableau 2.6. Matrice de l'indice de cohérence moyen (ICM).

Si la valeur de «  $RC$  » est < à 10% donc elle favorable (acceptable), sinon, il faut réduire ces incohérences en comparant les éléments par paires.

• Phase 5 : effectuer la « performance » lié à chaque alternative (action).

$$P_K(e_i^k) = \sum_{j=1}^{nk-1} P_{k-1}(e_i^{k-1}) \cdot P_k((e_i^k) / e_i^{k-1}), \text{ avec } \sum_{j=1}^{nk} P_k(e_i^k) = 1$$

Où  $nk - 1$  représente le nombre des éléments d'un même niveau de la structure hiérarchique  $k-1$ , et  $P_K(e_i^k)$  représente la priorité d'un élément  $e_i^k$  au niveau  $k$  de la structure hiérarchique.

#### b. Outils logiciels

- Criterium Decision Plus 3.0
- aliah Think
- Expert-Choice

### 2.3.4.2. Les méthodes d'aide à la décision multicritère de surclassèrent selon l'approche du sur-classement de synthèse

Les méthodes de cette famille commencent, le plus souvent, par paires d'alternatives. Alors, les alternatives sont comparées paires par paires afin de déterminer l'existence ou pas d'une « relation de sur-classement ». Après la comparaison de toutes les alternatives, une synthèse sur les différentes relations binaires sera réalisée pour répondre au problème décisionnel posé. Il important de noter que ce genre de méthodes est appliqué dans le cas où l'ensemble des actions est fini [Kazi 2009].

#### I. Les méthodes PROMETHEE (Preference Ranking Organization Method for Enrichment Evaluation) [Brans et Vincke 1985]

##### a. Notion de base de la méthode

Les méthodes PROMETHEE ont été développées par Brans et Vincke en 1985 ; ces méthodes sont basées sur l'utilisation d'une fonction qui permet au décideur d'exprimer ces préférences pour une alternative  $a_i$  par rapport à une autre alternative  $a_k$ . Pour chacun des attributs (critères), le décideur choisit une des six « formes de courbes » présentées dans la figure 2.13.

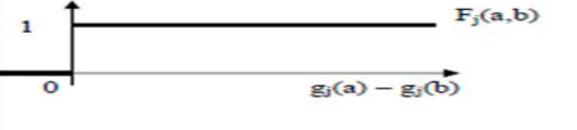
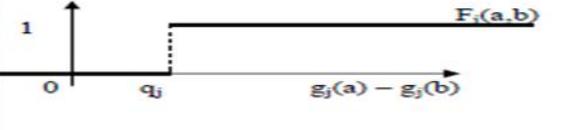
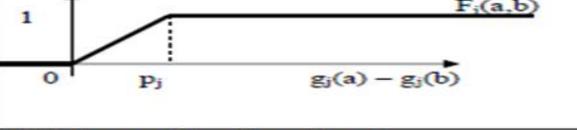
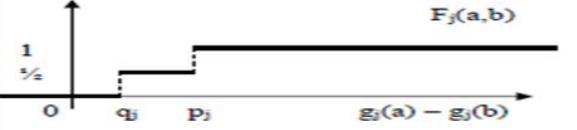
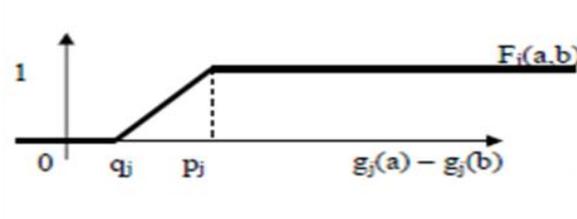
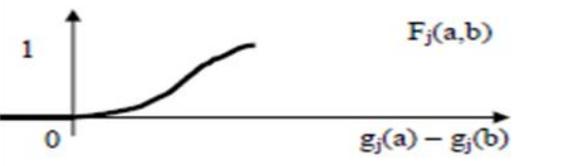
<p><b>1<sup>ère</sup> forme : Vrai-critère</b></p> 	<ul style="list-style-type: none"> <li>• Préférence stricte immédiate.</li> <li>• Pas de paramètres à déterminer.</li> </ul> $F_j(a,b) = \begin{cases} 1 & \text{si } g_j(a) > g_j(b) \\ 0 & \text{si } g_j(a) \leq g_j(b) \end{cases}$
<p><b>2<sup>ème</sup> forme : Quasi-critère</b></p> 	<ul style="list-style-type: none"> <li>• Il existe un seuil d'indifférence (quasi-critère) qui doit être fixé.</li> </ul> $F_j(a,b) = \begin{cases} 1 & \text{si } g_j(a) - g_j(b) > q_j \\ 0 & \text{si } g_j(a) - g_j(b) \leq q_j \end{cases}$
<p><b>3<sup>ème</sup> forme : Pré-critère</b></p> 	<ul style="list-style-type: none"> <li>• La préférence croît jusqu'à un seuil de préférence qui doit être fixé.</li> </ul> $F_j(a,b) = \begin{cases} 1 & \text{si } g_j(a) - g_j(b) > p_j \\ \frac{g_j(a) - g_j(b)}{p_j} & \text{si } 0 < g_j(a) - g_j(b) \leq p_j \\ 0 & \text{si } g_j(a) - g_j(b) \leq 0 \end{cases}$
<p><b>4<sup>ème</sup> forme : Pseudo-critère 1</b></p> 	<ul style="list-style-type: none"> <li>• Il existe un seuil d'indifférence et un seuil de préférence à fixer (pseudo-critère); entre les deux, la préférence est moyenne.</li> </ul> $F_j(a,b) = \begin{cases} 1 & \text{si } g_j(a) - g_j(b) > p_j \\ \frac{1}{2} & \text{si } q_j < g_j(a) - g_j(b) \leq p_j \\ 0 & \text{si } g_j(a) - g_j(b) \leq q_j \end{cases}$
<p><b>5<sup>ème</sup> forme : Pseudo-critère 2</b></p> 	<ul style="list-style-type: none"> <li>• Il existe un seuil d'indifférence et un seuil de préférence à fixer; entre les deux la préférence est croissante.</li> </ul> $F_j(a,b) = \begin{cases} 1 & \text{si } g_j(a) - g_j(b) > p_j \\ \frac{g_j(a) - g_j(b) - q_j}{p_j - q_j} & \text{si } q_j < g_j(a) - g_j(b) \leq p_j \\ 0 & \text{si } g_j(a) - g_j(b) \leq q_j \end{cases}$
<p><b>6<sup>ème</sup> forme : Critère gaussien</b></p> 	<ul style="list-style-type: none"> <li>• La préférence croît suivant une loi gaussienne dont il faut fixer l'écart type.</li> </ul> $F_j(a,b) = 1 - \exp\left(\frac{g_j(a) - g_j(b)}{-2\sigma^2}\right)$ <p>(avec <math>\sigma</math> = écart-type)</p>

Figure 2.13. Différentes formes de courbes de PROMETHEE[Vincke 1989].

Les étapes de la méthode PROMETHEE I et la méthode PROMETHEE II sont les suivantes :

- Phase 1 : fixer pour chacun des attributs (critères), une courbe parmi les six présentées précédemment avec les éléments (paramètres) qui lui sont attribués.
- Phase 2 : calculer pour chaque couple d'alternatives  $(a_i, a_k)$  « la préférence globale » qui représente le degré de sur-classement :

$$P(a_i, a_k) = \sum_{j=1}^n \pi_j \cdot F(a_i, a_k)$$

- Phase 3 : on calcule les « flux sortants » et les « flux entrants » pour chaque alternative  $a_i$ .

$\varphi^+(a_i) = \sum_{a_k \in A, a_k \neq a_i} P(a_i, a_k)$  ; le flux  $\varphi^+$  représente le « flux sortant », qui est la force de  $a_i$

$\varphi^-(a_i) = \sum_{a_k \in A, a_k \neq a_i} P(a_k, a_i)$  ; le flux  $\varphi^-$  représente le « flux entrant » qui est la faiblesse de  $a_i$

- Phase 4 : on détermine les deux « préordres totaux » par la suite on procède au classement des alternatives (actions) :

- Le premier « préordre total » range les alternatives dans un ordre décroissant des  $\varphi^+$ .
- Le deuxième « préordre total » range les alternatives dans un ordre croissant des  $\varphi^-$ .
- Afin d'obtenir les rangements (« préordre partiel ») par la méthode PROMETHEE I, il suffit de faire l'intersection des deux « préordres totaux ».
- Et pour obtenir les rangements (« préordre total ») par la méthode PROMETHEE II, il suffit de calculer les scores  $\varphi(a_i) : \varphi(a_i) = \varphi^+(a_i) - \varphi^-(a_i)$  puis les classer selon un ordre décroissant des scores  $\varphi(a_i)$ .

### b. Outils logiciels

- Promcalc – Gaia
- Decision Lab - 2000

## II. La méthode Electre I (Elimination Et Choix Traduisant la REalité)[Roy 1968]

### a. Notion de base de la méthode

Elle a été proposée par Roy en 1968. Cette méthode est considérée comme étant une problématique de choix ( $P\alpha$ ). Elle est basée sur deux concepts : une condition de « concordance » qui impose que la plupart des attributs se dégagent avantageant l'alternative surclassante et une condition de « non discordance » qui impose le fait qu'il n'y ait pas assez d'influence, dans un des critères de la minorité, pour favoriser un sur-classement inverse [Schärli 1985].

Le but de la méthode Electre I est l'obtention d'une division de  $A$  en 2 sous ensembles «  $N$  » et «  $A \setminus N$  ».  $N$  représente l'ensemble des alternatives non surclassées donc, la meilleure alternative sera dans  $N$ .

Les étapes de la méthode Electre I sont les suivantes :

- Phase 1 : on calcule tous les « indices de concordance »

Pour chacun des couples d'alternatives  $(a_i, a_k)$ , associer un « indice de concordance » comme suit :

$$C(a_i, a_k) = \sum_{j: e_{ij} \geq e_{kj}} \pi_j \text{ avec } \sum_{j=1}^n \pi_j = 1$$

- Phase 2 : on calcule « les indices de discordance »

Pour chacun des couples d'alternatives  $(a_i, a_k)$ , associer un « indice de discordance » comme suit :

$$D(a_i, a_k) = \begin{cases} 0 & \text{si } e_{ij} < e_{kj} = \emptyset \\ \frac{1}{E} * \text{Max} | e_{ij} - e_{kj} | & \text{sinon} \end{cases}$$

$E$  : représente « l'étendue de la plus grande échelle » attribuée à l'une des alternatives.

• Phase 3 : on construit les « liens de sur-classement » :

Si le test sur « l'indice de concordance » et le test sur « l'indice de non discordance » sont satisfaisants, alors on procède au sur-classement de  $a_k$  par  $a_i$ .

$$\text{Si } (D(a_i, a_k) \geq c \text{ et } D(a_i, a_k) \leq d) \leftrightarrow a_i S a_k$$

Si un des tests ou bien les deux tests ne sont pas satisfaits, alors on est dans le cas « d'incomparabilité ».

$$\text{Si } (C(a_i, a_k) \leq c \text{ et } D(a_i, a_k) \geq d) \leftrightarrow a_i R a_k$$

$c$  relativement grand, il appartient à l'intervalle  $[1/2, 1]$  et  $d$  relativement petit.

• Phase 4 : on étudie les « liens de sur-classement » :

Cette phase détermine un sous ensemble d'alternatives  $N$  nommé « le noyau » dont les alternatives sont incomparables. Toute alternative qui n'appartient pas à l'ensemble  $N$  est surclassée par au moins une alternative de l'ensemble  $N$ .

### b. Outils logiciels

- Electre Tri (version démo gratuite)

## III. La méthode Electre II [Roy et Bertier 1971]

### a. Principe de la méthode

La méthode Electre II a été développée par Roy et Bertier en 1971. Cette méthode est considérée comme étant une problématique de rangement ; elle range les alternatives de la meilleure à la moins bonne. Elle emploie le même « indice de concordance » et de « discordance » que la méthode Electre I. Cependant, nous associons à l'indice de concordance trois seuils ( $0.5 < c_3 < c_2 < c_1 \leq 1$ ) et deux seuils par attribut à l'indice de discordance ( $0 < d_{j1} < d_{j2} < E_j$ ). Cette méthode distingue deux types de sur-classement : sur-classements fort et faible. Les étapes de la méthode Electre II sont les suivantes : Tout d'abord, associer à chaque attribut  $j$  un poids  $\pi_j$ .

• Phase 1 : associer un « indice de concordance » à chacun des couples d'alternatives  $(a_i, a_k)$  :

$$C(a_i, a_k) = \sum_{j: (e_{ij} > e_{kj})} \pi_j \quad \text{avec } \sum_{j=1}^n \pi_j = 1$$

- Phase 2 : associer un « indice de discordance » à chacun des couples d'alternatives  $(a_i, a_k)$  et pour tout attribut  $j$  :

$$D_j(a_i, a_k) = \begin{cases} 0 & \text{si } e_{ij} \geq e_{kj} \\ e_{kj} - e_{ij} & \text{si } e_{ij} < e_{kj} \end{cases}$$

- Phase 3 : on construit les « liens de sur-classement » :

Si le test sur la « concordance » et le test sur « non discordance » sont satisfaisants, alors on procède au *sur-classement fort* de  $a_k$  par  $a_i$  ( $a_i S^F a_k$ )

$$\text{Si } \left( \frac{\sum_{j:\Delta_j > 0} \pi_j}{\sum_{j:\Delta_j < 0} \pi_j} \right) > 1 \text{ et } \begin{cases} \text{si} & (C(a_i, a_k) \geq c_1 \text{ et } (D(a_i, a_k) \leq d_{j2} \forall j)) \\ & \text{ou} \\ \text{si} & (C(a_i, a_k) \geq c_2 \text{ et } (D(a_i, a_k) \leq d_{j1} \forall j)) \end{cases} \leftrightarrow a_i S^F a_k$$

Si les tests sur les « indices de concordance » et les indices de « non discordance » sont satisfaisants, alors on procède au *sur-classement faible* de  $a_k$  par  $a_i$  ( $a_i S^f a_k$ )

$$\text{Si } \left( \frac{\sum_{j:\Delta_j > 0} \pi_j}{\sum_{j:\Delta_j < 0} \pi_j} \right) > 1 \text{ et si } (C(a_i, a_k) \geq c_3 \text{ et } (D(a_i, a_k) \leq d_{j2} \forall j)) \leftrightarrow a_i S^f a_k$$

Si ces deux tests ne sont pas satisfaisants, donc on procède à l'incomparabilité des alternatives  $a_i$  et  $a_k$  ( $a_k R a_i$ ).

- Phase 4 : on étudie les « liens de sur-classement » :

Déterminer les « préordres totaux »  $R1$ ,  $R2$  et le « préordre partiel »  $R$ .

- On détermine le « préordre total »  $R1$  en employant le « sur-classements fort » : la première catégorie contient les alternatives non surclassées (qui ont eu un chemin de longueur nulle). La seconde catégorie contient les alternatives qui ont eu un chemin de longueur 1, et ainsi de suite, on appelle ce classement : un « classement /rangement direct ». la longueur d'un chemin c'est le nombre d'arcs qu'il le constitue. Afin de départager les alternatives au sein des classes (catégories) on emploie les sur-classements faibles.
- On détermine le « préordre total »  $R2$  en classant les alternatives par rapport à longueur de leurs trajets en « sur-classement fort », on appelle ce classement : un « classement /rangement inverse ». Afin de départager les alternatives au sein des classes (catégories) on emploie le « sur-classement faibles ».
- Enfin, on détermine le « préordre partiel »  $R$  en appliquant une intersection entre  $R1$  et  $R2$ .

#### IV. La méthode Electre III [Roy 1978]

##### a. Notion de base de la méthode

Cette méthode a été proposée aussi par Roy en 1978. Elle est considérée comme étant une problématique de rangement. La méthode Electre III établit pour chaque couple d'alternatives

un « degré de crédibilité du sur-classement »  $S(a_i, a_k)$  qui est inclus dans l'intervalle  $[0,1]$  ; tant que la « solidité du sur-classement » de  $a_i$  sur  $a_k$  est grande, « le degré de crédibilité » sera grand.

Cette méthode se différencie des deux autres Electre I et Electre II par l'utilisation de « pseudo-critères » : quand on compare les différents écarts de préférence de deux alternatives selon un attribut (critère) on se retrouver soit dans une situation préférence stricte ou de préférence faible ou bien dans une situation d'indifférence.

Les étapes de la méthode Electre III sont les suivantes : Tout d'abord, associer à chaque attribut  $j$  un poids  $\pi_j$ .

- Phase 1 : on calcule les « indices de concordance »

$$C(a_i, a_k) = \sum_{j=1}^n \pi_j \cdot \delta_j(a_i, a_k) \quad \text{Où } \delta_j(a_i, a_k) = \begin{cases} 1 & \text{si } -q_j \leq \Delta_j \\ \frac{\Delta_j + p_j}{p_j + q_j} & \text{si } -p_j < \Delta_j \leq -q_j \\ 0 & \text{si } \Delta_j \leq -q_j \end{cases}$$

$\Delta_j = e_{ij} - e_{kj}$  et  $q_j, p_j$  représentent les « seuils de discrimination »

- Phase 2 : on calcule par attribut les « indices de discordance »

$$D_j(a_i, a_k) = \begin{cases} 0 & \text{si } -p \leq \Delta_j \\ \frac{\Delta_j + p_j}{p_j + q_j} & \text{si } -v_j < \Delta_j \leq -p_j \\ 1 & \text{si } \Delta_j \leq -v_j \end{cases}$$

$v_j$  est le « seuil de veto » de l'attribut (critère)  $j$

- Phase 3 : on exploite les « degrés de crédibilité des sur-classements » :

$$\sigma(a_i, a_k) = \begin{cases} C(a_i, a_k) & \text{si } D_j(a_i, a_k) \leq C(a_i, a_k) \quad \forall j \\ C(a_i, a_k) \times \prod_{j: D_j(a_i, a_k) > C(a_i, a_k)} \frac{1 - D_j(a_i, a_k)}{1 - C(a_i, a_k)} & \text{si } \exists j : D_j(a_i, a_k) > C(a_i, a_k) \end{cases}$$

Cette étape détermine les « liens de sur-classement valués » compris entre 0 et 1, pour tout couple d'alternatives  $(a_i, a_k)$ .

- Phase 4 : on détermine les « liens de sur-classement » :

Cette étape détermine un classement des alternatives. Pour ce faire, on joue sur le niveau de signification du degré de crédibilité et on procède à des distillations successives descendantes et ascendantes qui aboutissent à deux rangements (préordre complet). L'intersection de ces deux rangements conduit à un rangement final (généralement partiel).

## b. Outils logiciels

- Electre III (version démo gratuite)

## V. La méthode ELECTRE IV

### a. Notion de base de la méthode

La méthode ELECTRE IV est identique à la méthode ELECTRE III ; la seule différence qui existe entre elles c'est que la méthode ELECTRE IV ne prend pas en considération les poids des différents attributs (critères). Cette méthode est utilisée alors dans cas où le décideur veut accorder aux critères la même importance. Dans cette méthode, nous évaluons chaque couple d'alternative selon chaque attribut sans avoir à définir un « indice de concordance » ou de « discordance ». [Shanian et Savadogo 2006].

### b. Outils logiciels

- Electre III—IV de Lamsade

## VI. La méthode Electre Is [Hammami 2003]

### a. Notion de base de la méthode

Cette méthode est très identique à la méthode Electre I, hors qu'elle est applicable dans les situations où le problème concerne les « pseudo critères ». Cette méthode permet de déterminer un « noyau »  $N$ . Les étapes de celle-ci sont les suivantes :

- Phase 1 : on calcule les « indices de concordance »

$$C(a_i, a_k) = \sum_{j=1}^n \pi_j \cdot \delta_j(a_i, a_k) \quad \text{où} \quad \delta_j(a_i, a_k) = \begin{cases} 1 & \text{si } -q_j \leq \Delta_j \\ \frac{\Delta_j + p_j}{p_j + q_j} & \text{si } -p_j < \Delta_j \leq -q_j \\ 0 & \text{si } \Delta_j \leq -q_j \end{cases}$$

Avec  $\Delta_j = e_{ij} - e_{kj}$  et  $q_j, p_j$  représentent les « seuils de discrimination »

- Phase 2 : on calcule les « indices de discordance » par attribut

$$C(a_i, a_k) = \sum_{j: (e_{ij} \geq e_{kj})} \pi_j \quad \text{avec} \quad \sum_{j=1}^n \pi_j = 1$$

- Phase 3 : on construit les « liens de sur-classement »

$$\begin{cases} \text{si } (C(a_i, a_k) \geq c) \text{ et } D_j(a_i, a_k) \leq (v_j - D), \forall j & \leftrightarrow (a_i S a_k) \\ \text{sinon} & \leftrightarrow (a_i R a_k) \end{cases}$$

$$\text{Où } D = q_j * \frac{1 - C(a_i - a_k)}{1 - c}$$

- Phase 4 : on exploite les « liens de sur-classement » : dans cette phase on détermine le « noyau » du « graphe de sur-classement ».

### b. Outils logiciels

- ELECTRE IS

### **2.3.4.3. Les méthodes d'aide à la décision multicritère interactives selon l'approche du jugement local interactif**

Les méthodes interactives sont également appelées méthodes d'agrégation locale et itérative. Cette appellation renvoie au fait que ces dernières procèdent, en premier lieu, par la détermination d'une solution de départ. Elles effectuent ensuite une recherche dans l'environnement de cette solution pour essayer d'aboutir à un meilleur résultat, d'où le qualificatif et progressif, le terme itératif a été également utilisé pour qualifier les méthodes interactives. Ainsi ces dernières permettent de simuler les préférences de chacun des décideurs d'une façon séquentielle et interactive. En effet, elles s'attachent à révéler progressivement des phases de calcul et de dialogue. Cette succession d'étapes a pour finalité d'arriver à un compromis final qui puisse satisfaire le décideur [Kazi 2009].

### **2.3.5. Algorithme de sélection d'une méthode d'aide à la décision multicritère**

Dans la section précédente, nous avons étudié les méthodes multicritère d'aide à la décision et nous concluons que celles-ci se distinguent par la manière dont elles modélisent les préférences des différents décideurs, l'affectation ou non des poids aux critères ainsi que les attentes du décideur qui correspondent aux résultats de l'application d'une méthode décisionnelle multicritère (tri, classement, choix,..). Cela veut dire que ces méthodes se distinguent par les données employées en Input (entrée) et les résultats fournis en Output (sortie). En utilisant des entrées et des sorties, nous pouvons ranger les méthodes multicritère d'aide à la décision, ainsi nous créons un algorithme qui permet de choisir et de sélectionner la méthode d'aide à la décision multicritère appropriée. Cette idée a été proposée par Guitouni, Martel J-M et Vincke en 1999 [Guitouni et al. 1999]. Guitouni et al. [Guitouni et al. 1999] ont défini 24 entrées et 7 sorties. C'est l'équivalent d'un tableau de 24 lignes et 7 colonnes où chacune des méthodes multicritère est située dans la cellule appropriée. La distribution des méthodes classification est présentée par le Tableau 2.7.

Inputs	Outputs						
	01	02	03	04	05	06	07
I1							
I2							
I3				ELECTRE IV			
I4							
I5							
I6							
I7							
I8							
I9							
I10							
I11							
I12							
I13							
I14							
I15							
I16							
I17				ELECTRE II		ELECTRE I	
I18			PROMETHEE II	PROMETHEE I			
I19				ELECTRE III		ELECTRE IS	
I20	MAUT		MAVT SMART				
I21	AHP						

Tableau 2.7. Classification des MADMC selon la matrice inputs/outputs [Vincke 1989] [Guitouni et al. 1999].

La description des différents inputs et outputs de la matrice est représentée ci-après. Nous nous limitons à expliciter les inputs et outputs correspondant à des cases pleines de la matrice.

### 2.3.5.1. Caractérisation des entrées

Le Tableau 2.8 donne un aperçu sur les diverses caractérisations des entrées suggérés par Guitouni et al. [Guitouni et al. 1999]

Inputs	Signification
11	$n$ structures $\{P_j, I_j\}$ ( $\{P_j$ : relation de préférence stricte, $Q_j$ : relation de préférence faible, $I_j$ : relation d'indifférence}) de préordres.
13	$n$ structures $\{P_j, I_j\}$ de semi ordres et/ou structures $\{P_j, I_j, Q_j\}$ ( $\{P_j, I_j, Q_j\} \equiv \{>, \sim, >^f\}$ ) de pseudo-ordres plus des seuils de veto $v_j$ .
14	$n$ fonctions d'utilité ( $u_j$ ) (d'utilité (valeur) partielle) exprimées sur des échelles intervalles.
110	$n$ structures $\{P_j, I_j\}$ de semi ordres et/ou structures $\{P_j, I_j, Q_j\}$ de pseudo-ordres et une relation de préordre complet ( $f$ ) sur les attributs (critères).
112	$n$ fonctions d'utilité ( $u_j$ ) (d'utilité (valeur) partielle) exprimées sur des échelles intervalles et une relation de pré ordre complet ( $f$ ) sur les attributs.
116	$n$ structures $\{P_j, I_j\}$ de semi ordres et/ou structures $\{P_j, I_j, Q_j\}$ de pseudo-ordres plus des seuils de veto $v_j$ définies sur des évaluations distributionnelles plus une relation de pré ordre complet ( $f$ ) sur les attributs.
117	$n$ structures $\{P_j, I_j\}$ de pré ordres plus un vecteur de coefficients d'importance relative ( $\Pi$ ) des attributs.
118	$n$ structures $\{P_j, I_j\}$ de semi ordres et/ou structures $\{P_j, I_j, Q_j\}$ de pseudo-ordres plus un vecteur de coefficient d'importance relative ( $\Pi$ ) des attributs.
119	$n$ structures $\{P_j, I_j\}$ de semi ordres et/ou structures $\{P_j, I_j, Q_j\}$ ( $\{P_j, I_j, Q_j\} \equiv \{>, \sim, >^f\}$ ) de pseudo ordres plus des seuils de veto $v_j$ plus un vecteur de coefficient d'importance relative ( $\Pi$ ) des attributs.
120	$n$ fonctions d'utilité ( $u_j$ ) (d'utilité (valeur) partielle) exprimées sur des échelles intervalles plus un vecteur de coefficient d'importance relative ( $\Pi$ ) des attributs.
121	$n$ fonctions d'utilité ( $u_j$ ) (d'utilité (valeur) partielle) exprimées sur des échelles ratios plus un vecteur de coefficient d'importance relative ( $\Pi$ ) des attributs.
124	$n$ structures $\{P_j, I_j\}$ de semi ordres et/ou structures $\{P_j, I_j, Q_j\}$ de pseudo-ordres plus des seuils de veto $v_j$ définies sur des évaluations distributionnelles plus un vecteur de coefficient d'importance relative ( $\Pi$ ) des attributs.

Tableau 2.8. Caractérisation des entrées.

### 2.3.5.2. Caractérisation des sorties

Le Tableau 2.9 donne un aperçu sur les diverses caractérisations des entrées suggérée par Guitouni et al. [Guitouni et al. 1999].

Outputs	Signification
01	<b>Évaluation globale (un score global par exemple)</b> : cette situation traduit la volonté de construire un critère unique de synthèse. Le score global est de niveau cardinal.
02	<b>Un rangement global des actions en considérant un seuil d'indifférence</b> ( $\{>, \sim\}$ : une structure de semi ordre total) : l'introduction de seuils permet d'introduire des nuances ou limiter certaines conclusions (de surclassement) par exemple.
03	<b>Un rangement total des actions avec possibilité d'ex æquo</b> ( $\{>, \sim\}$ : une structure de pré ordre total) : l'objectif dans ce cas est souvent le rangement des actions de la meilleure à la moins bonne. Dans ce cas, toutes les actions sont comparables; il est possible de les discriminer.
04	<b>Un rangement partiel des actions en considérant l'incomparabilité</b> : ( $\{>, \sim, >^f, ?\}$ : une structure de pré ordre partiel qui signifie {Préférence forte, Indifférence, Préférence faible, incomparabilité}) : cet output reprend l'idée de l'output précédent tout en considérant qu'il est possible de ne pas pouvoir discriminer entre certaines actions : on accepte l'incomparabilité.
05	<b>Choix de la meilleure action</b> ou d'une classe d'équivalence des meilleures actions.
06	<b>Choix d'un sous-ensemble d'actions</b> parmi lesquelles se trouve(nt) la (les) meilleure(s) action(s).
07	<b>Tri ordonné</b> : affecter les actions à des catégories pré-définies et ordonnées.

Tableau 2.9. Caractérisation des sorties.

## 2.4. Etat de l'art : les systèmes décisionnels en planification urbaine

Dans le contexte de la planification urbaine, différentes recherches ont été effectuées pour résoudre les problèmes dans les divers secteurs tel que l'habitat, le transport, l'évaluation, l'environnement... etc. Nous citons :

Yang et Chen [Yang et Chen 2002] ont proposé un système décisionnel basé sur l'intégration d'un système d'information géographique (SIG) pour la planification du transport public afin de réduire le trafic urbain en Chine. Fayeche, Maouche, Hammadi et Borne [Fayeche et al. 2002] ont présenté un système d'aide à la décision multi-agent (SMA) afin de planifier le réseau de transport urbain. Dans [Natividade et al. 2007], les auteurs ont proposé un système d'aide à la décision qui assiste les décideurs dans le processus d'évaluation et de choix des logements. Ce système décisionnel est basé essentiellement sur l'intégration d'une multitude de méthodes multicritère d'aide à la décision et des outils de système d'information géographique. Jun et Yikui [Jun et Yikui 2009] ont développé un système décisionnel intelligent pour la planification du réseau des routes urbaines. Longfei et Hong [Longfei et Hong 2009] ont suggéré une approche décisionnelle en utilisant un processus de négociation afin de résoudre le problème de stationnement dans une ville urbaine. Hasan [Hasan 2010] a proposé un système décisionnel intelligent (IDSS) pour la gestion du système de congestion du trafic. Dawood et Alshawi [Dawood et Alshawi 2009] ont développé un système d'aide à la décision multicritère qui aide les décideurs à évaluer les projets de construction de nouvelles villes destinés à la résolution du problème de l'habitat précaire dans le monde islamique. Dans [Ahris et al. 2009], les auteurs ont suggéré un système décisionnel basé sur l'utilisation des SIG et l'aide à la décision multicritère qui permettent de planifier et gérer le développement durable. Shi et Li [Shi et Li 2010] ont

décrit le développement d'un système d'aide à la décision basé sur l'intégration des systèmes experts pour l'évaluation et le contrôle de la pollution atmosphérique liée à la circulation urbaine. Dans [Maktav et al. 2011], les auteurs ont développé un système d'aide à la décision spatial basé sur l'analyse multicritère pour la gestion des espaces vides dans une ville urbaine. Dans [Yujing et al. 2011], les auteurs ont développé un modèle décisionnel qui aide les décideurs gouvernementaux à évaluer et améliorer la performance des projets urbains. Ce dernier est basé sur l'aide à la décision multicritère plus particulièrement sur l'utilisation de la méthode AHP. Dans [Moreira de Oliveira et al. 2012], les auteurs ont suggéré un système d'aide à la décision pour la planification du transport public, ce système est basé sur l'utilisation des systèmes d'information géographique (SIG) ce qui assure une réglementation, une surveillance et une planification efficace des lignes de bus. Takahashi, Kanamori et Ito [Takahashi et al. 2013] ont proposé un système décisionnel basé sur la négociation pour le changement de route en cas d'embouteillage.

Chacun de ces travaux décisionnels tente de résoudre un problème dans un secteur (domaine) précis en utilisant une multitude d'outils comme illustré dans le tableau 2.10.

	Domaine	Outils					
		SIG	MDAM	SE	IA	SMA	Négociation
Yang et Chen (1993)	Transport	X					
Fayech et al. (2002)	Transport					X	
Natividade et al. (2007)	Habitat Evaluation	X	X				
Jun et Yikui (2009)	Transport				X		
Longfei et Hong (2009)	Transport						X
Hasan (2010)	Transport				X		
Alshawi et Dawood (2009)	Evaluation Habitat		X				
Ahris et al. (2009)	Economique Sociale Environnemental	X	X				
Shi et Li (2010)	Environnemental			X			
Maktav et al. (2011)	Environnemental	X	X				
Yujing et al. (2011)	Evaluation		X				
Moreira de Oliveira et al. (2012)	Transport	X					
Takahashi et al. (2013)	Transport						X

Tableau 2.10. Les DSS proposés pour la planification urbaine.

Ces travaux montrent que le système d'aide à la décision peut être un outil efficace pour la planification urbaine dans différents domaines en utilisant une large gamme d'outils de DSS tels que : l'intelligence artificielle, les systèmes multi-agent, les systèmes experts, les techniques d'aide à la décision multicritère (MADM), les systèmes d'information géographique (SIG), la négociation ...etc.

Dans notre travail de recherche, nous nous sommes concentrée plus particulièrement sur le domaine d'évaluation des projets urbains. Peu de recherches ont été effectuées dans ce domaine : les auteurs dans [Dawood et Alshawi 2009] ont proposé leur propre processus décisionnel dans lequel ils se sont limités à une évaluation financière des projets urbains en utilisant uniquement des critères économiques afin de choisir le projet urbain le plus

économique. Les auteurs dans [Yujing et al. 2011] ont élaboré une évaluation multicritère en combinant la théorie de la logique floue avec la méthode AHP pour choisir le meilleur projet urbain. Les critères des projets ont été classés en trois catégories : économique, écologique et sociale ; durant l'évaluation les critères de la même catégorie ont eu le même poids alors qu'en réalité ils peuvent être vus différemment par les décideurs donc ils n'auront pas le même poids. Par conséquent, nous proposerons un système d'aide à la décision (DSS) qui effectue une évaluation des projets urbains en prenant en considération tous les critères du développement durable (économique, environnemental et social). Notre système décisionnel sera basé sur l'utilisation de deux outils : l'aide multicritère à la décision qui permet aux décideurs d'affecter les poids des critères selon leurs préférences et connaissances et la négociation qui trouve un compromis entre eux.

Dans le but d'améliorer notre DSS en terme de développement, d'accessibilité et de communication, nous suggérons de l'intégrer dans une architecture de Cloud Computing ce qui peut améliorer l'efficacité d'évaluation des projets urbains.

## 2.5. Conclusion

L'aide à la décision multicritère permet d'offrir au décideur un ensemble d'outils et de méthodes lui permettant d'évoluer dans la résolution d'un problème décisionnel où différents points de vue conflictuels sont pris en compte. L'analyse multicritère ne permet pas forcément de trouver des solutions optimales, mais elle aide les différents décideurs à déterminer au moins une solution satisfaisante.

Dans cette partie, nous avons donné un aperçu sur l'aide à la décision, quelques notions fondamentales en problème multicritère, les différentes problématiques multicritères, les principales méthodes multicritères et nous avons terminé ce chapitre par les travaux liés aux systèmes d'aide à la décision en planification urbaine.

Les méthodes multicritère d'aide à la décision sont des techniques assez récentes et en plein développement. Nous utiliserons donc dans une partie de notre processus décisionnel une de ces méthodes à savoir la méthode PROMETHEE II, nous développerons l'utilisation de celle-ci par la suite.

Les méthodes multicritère ce sont des méthodes qui donnent des décisions automatiques, alors que notre problématique inclut plusieurs acteurs avec des points de vue contradictoires et les décisions résultantes des méthodes multicritères peuvent ne pas satisfaire la plupart des décideurs ce qui engendre un conflit et un débat entre eux. Afin de trouver un compromis entre les différents décideurs, nous faisons appel au paradigme de la négociation qui sera abordé plus en détail lors du prochain chapitre.

# Chapitre 3

## Négociation entre acteurs

*« L'objectif de ce chapitre est de présenter un aperçu sur la négociation ainsi que ses diverses formes. Aussi, une taxonomie des systèmes de négociation automatique est abordée dans ce chapitre. »*

---

### Sommaire

3.1. Introduction.....	68
3.2. Définition de la négociation .....	68
3.3. Les différentes formes de la négociation .....	68
3.3.1. Les systèmes de vote .....	69
3.3.2. Les enchères .....	70
3.3.3. Les systèmes contractuels.....	70
3.3.4. Les systèmes de négociation basés sur l'argumentation .....	71
3.4. Les différents systèmes de négociation automatisée.....	72
3.4.1. Le Kasbah.....	72
3.4.2. L'AuctionBot .....	73
3.4.3. Le Fishmarket.....	73
3.4.4. Le système Adept .....	74
3.4.5. Magnet .....	74
3.5. Les différents systèmes de négociation automatisée proposés en planification urbaine.....	74
3.6. Conclusion.....	75

### 3.1. Introduction

La négociation désigne la stratégie de résolution qui utilise le dialogue pour parvenir à un accord visant à résoudre des conflits de croyances ou de buts. Ces conflits sont produits par l'existence de contradiction entre les différentes opinions (points de vue) des acteurs. Ils sont dû au fait que les acteurs possèdent des connaissances incomplètes voire erronées [Bouron 1992]. La négociation est donc un processus de communication d'un groupe d'agent en conflit permettant de trouver une solution satisfaisante entre eux. La négociation a soulevé l'intérêt de plusieurs chercheurs dans différents domaines d'application- tels que l'économie, le commerce, la santé, l'environnement... Dans cette partie, nous allons présenter le principe de la négociation, les différentes formes de la négociation et les systèmes de négociation automatisée existants.

### 3.2. Définition de la négociation

Selon le point de vue de Kraus [Kraus et Wilkenfeld 1991], la négociation se définit comme étant le processus par lequel au moins deux parties avec des motivations, des préférences et des contraintes, qui peuvent être similaires ou conflictuelles, arrivent conjointement à un accord des échanges plus au moins structurés et dans des temps plus au moins longs.

D'après Guttman [Guttman et al. 1998], la négociation est définie comme étant : « un processus de prise de décision où deux parties ou plus cherche conjointement un consensus dans un espace de solutions potentielles ».

Selon le point de vue de Verrons [Verrons 2004], la négociation aura lieu quand il y a un dialogue, un ensemble de suggestions entre les acteurs et quand il y a un compromis final qui satisfait l'ensemble des intervenants. Verrons a donné aussi une autre définition dans [Verrons 2004]: « une négociation met en jeu des ressources, qui seront rassemblées afin d'être négociées dans un contrat et un ensemble de personnes qui participent à cette négociation. Il y a toujours un ou plusieurs manageurs (vendeur ou autre) et un ou plusieurs contractants (acheteurs ou autre) ».

En général les chercheurs donnent des définitions de la négociation mais qui sont similaires car la notion de négociation n'est pas récente, elle est apparue même avant l'apparition de l'informatique. Les chercheurs n'ont rien apporté de nouveau, ils décrivent seulement la réalité.

### 3.3. Les différentes formes de la négociation

Les différentes formes de la négociation sont regroupées en quatre catégories [Verons 2004] ; la première catégorie regroupe les « systèmes de vote », leur rôle est de sélectionner une action (solution) parmi un ensemble d'actions. La seconde catégorie concerne les « enchères » dont les types sont l'enchère anglaise, l'enchère hollandaise et l'enchère de Vickrey. La troisième famille concerne les « contractuelles » dont le plus utilisé est le protocole Contract Net. La dernière catégorie est celle des « protocoles d'argumentations ».

### 3.3.1. Les systèmes de vote

Ces systèmes permettent de choisir une solution parmi les diverses solutions possibles. Il existe plusieurs systèmes de vote, parmi les plus connus, nous citons :

#### I. La méthode Borda

La méthode Borda [Verrons 2004] a été développée en 1781 par Jean Charles Borda. Le principe général de cette méthode est d'affecter des scores à chaque action (alternative). Alors, pour chacune des listes de préférence, l'action classée première obtiendra (n-1) scores, les seconds (n-2) scores et ainsi de suite jusqu'à la dernière action qui remportera 0 score. Par la suite, pour chaque action, nous calculons la somme des scores qu'elle a remporté, l'action qui a eu les meilleurs points est déclarée comme étant le « choix social ».

On est certain d'obtenir un « choix social », vu que toutes les actions sont rangées par les électeurs ; donc elles ont toutes un score. Dans le cas d'égalité entre les scores, la méthode refait la procédure d'affectation de point avec un sous ensemble d'actions.

#### II. La méthode de Hare

Cette méthode a été développée en 1861 par Thomas Hare [Debord 1987]. Elle permet de retirer successivement les actions les moins désirées dans le but de trouver le « choix social ». Les étapes de la méthode Hare sont les suivantes :

- Dans le cas où, une action est rangée première selon au moins la moitié des séries de préférence, il existe une meilleure action (choix social) et la méthode est achevée.
- Dans le cas contraire où il n'y a pas une action qui est rangée première selon au moins la moitié des séries, on choisit alors l'action rangée première sur le moins de séries et on la supprime de toutes les séries. Si une multitude d'actions sont à égalité, on les supprime des séries. Les actions qui suivent celle qui a été supprimée sur les séries remportent une place.
- On recommence la méthode pour déterminer si une action est classée première selon au moins la moitié des séries et on supprime l'action la moins souhaitée dans le cas contraire.
- La méthode cesse lorsqu'une action se présente en premier selon au moins la moitié des séries ou bien dans le cas où toutes les actions persistantes se présentent en premier sur précisément le même nombre de séries.

#### III. La méthode de Condorcet

La méthode Condorcet [Mangin 2002] a été développée au XVIIIe siècle par le marquis de Condorcet. Cette méthode permet de déterminer pour chaque couple d'actions le nombre de votants qui ont choisit l'une ou l'autre tout en contrôlant comment une alternative est classée par rapport à l'autre sur chacune des listes de préférences,

Le vote sera remporté par l'action qui gagnera toutes les comparaisons avec les autres. Il est nécessaire de noter qu'il est possible qu'aucune action ne soit choisit. Condorcet a constaté que cette méthode possède un inconvénient dans le cas où on a une action  $a$  privilégiée à  $b$ , et  $b$  est

privilégiée à  $c$ , et  $c$  est privilégiée à  $a$ . Alors, il faut concevoir une procédure afin de résoudre cette situation conflictuelle.

### 3.3.2. Les enchères

Les enchères ont vu le jour grâce aux sites Internet. Ils représentent la forme la plus connue dans la négociation. Les enchères sont employées pour vendre les objets d'antiquités et d'art, les produits, les véhicules et l'immobilier et aussi elles sont employées dans quelques jeux comme le « bridge ». Une enchère est composée, généralement deux types d'acteurs : un acteur initiateur et plusieurs acteurs participants. Il existe différentes formes d'enchères dont les plus diffusées sont : les enchères « anglaises », « hollandaise », les « offres cachées au meilleur prix » et les « offres cachées au second meilleur prix » [Vickrey 1961].

#### •Les enchères anglaises

Les enchères anglaises sont appelées également les enchères ascendantes parce que le tarif préposé pour l'objet en vente s'élève dans le temps. Ce type d'enchères s'exécute en plusieurs étapes successives, il permet à l'acteur initiateur d'annoncer le « tarif de réservation » qui représente le tarif minimal pour lequel il approuve la vente du bien. Chacun des acteurs participants déclare ouvertement son offre. Dans le cas où aucun des acteurs participants n'augmente plus l'offre alors, l'enchère s'achève et l'intervenant (participant) qui a proposé le plus grand prix gagne.

#### •Les enchères hollandaises

Les enchères hollandaises sont appelées également les enchères descendantes parce que le tarif préposé pour l'objet en vente baisse dans le temps. Ce type d'enchères s'exécute en plusieurs étapes successives, il permet à l'acteur initiateur de suggérer un tarif qui baisse jusqu'à ce qu'un des acteurs participants achète le bien au tarif proposé.

#### •Les offres cachées au meilleur prix

Ce type d'enchères s'exécutent en une seule étape, chaque acteur participant (acheteur) suggère un tarif unique, sans qu'il connait le tarif suggéré par les autres (d'où le terme offres cachées). L'acteur qui gagnera l'offre est celui qui a suggéré le tarif le plus élevé et qui peut le régler. On conclut donc qu'il n'existe pas une deuxième suggestion possible, alors ces enchères ne sont pas considérées comme étant une négociation. La procédure de ces enchères se déroule comme suit : l'initiateur (vendeur) va suggérer un objet à vendre à un ensemble de participants (acheteurs). Ces derniers acceptent ou refusent l'offre. Dans le cas d'acceptation, les acheteurs suggèrent un prix. Celui qui a suggéré le meilleur tarif (supérieur au tarif de réserve), remportera l'enchère.

#### •Les offres cachées au second meilleur prix [Vickrey 1961]

Ce type d'enchères est nommé les enchères de Vickrey. Il se déroule de la même manière que les enchères aux offres cachées au meilleur prix; la seule différence est que le gagnant paie le second meilleur prix suggéré. Dans ce cas aussi, il n'existe pas une deuxième suggestion possible et donc le principe de la négociation est absent. La procédure d'exécution est similaire à celle des offres cachées au meilleur prix, la seule diversité étant le tarif à régler par le vainqueur du contrat.

### 3.3.3. Les systèmes contractuels

D'après Smith et David la négociation est un processus caractérisé par trois éléments [David et Smith 1980], l'échange d'informations qui se fait de façon bilatérale, l'évaluation de la négociation et l'accord final. Dans ce contexte Smith et David [David et Smith 1980] ont développé un protocole de négociation appelé le Contract Net Protocol (CNP).

Le protocole Contract Net est un processus de négociation qui se déroule entre deux familles d'acteurs : un acteur appelé le « contractant » et un autre acteur appelé le « gestionnaire ». Le protocole Contract Net permet au « gestionnaire » d'effectuer un échange avec les différents acteurs participants (contractants) puis de choisir le contrat le plus avantageux parmi un ensemble de contrats proposés par les participants. Cette situation s'appelle « la sélection mutuelle ». Ce protocole donc, est composé de trois phases essentielles : la première phase consiste en l'« annonce de l'offre », la deuxième phase concerne la « soumission de la proposition » et la dernière phase c'est l'« affectation du contrat ».

Une extension du protocole Contract Net a été proposée dans [Verrons 2004], cette extension est basée sur un fractionnement décentralisé des activités. Donc ici, l'acteur « gestionnaire » divise une activité en plusieurs sous-activités puis il les annonce à un acteur ou à un ensemble d'acteurs (contractants). Ce protocole étendu se distingue du protocole Contract Net précédent par le fait que l'activité peut être approuvée ou rejetée d'une façon temporaire, ou bien approuvée ou rejetée d'une façon définitive par les acteurs participants. Ce protocole permet à l'acteur « gestionnaire » d'annoncer une commande à un ensemble d'acteurs participants par la suite il choisit le meilleur acteur (contractant) à qui il soumet une acceptation temporaire. Tous les autres acteurs vont recevoir un refus temporaire. Dans le cas où, la meilleure offre ne satisfait pas toute la commande, l'acteur « gestionnaire » ré-annonce ce qui reste de la commande jusqu'à ce que elle soit totalement satisfaite. Dans ce contexte, les acceptations et les refus temporaires se transforment en acceptations et refus définitifs. Lorsqu'un acteur participant obtient une acceptation temporaire, il va concevoir un exemplaire de son ancienne stratégie qui sera récupérée dans le cas d'un refus négatif.

### 3.3.4. Les systèmes de négociation basés sur l'argumentation

La négociation basée sur l'argumentation est employée chez les acteurs logiques possédant une « base de connaissances » qui contient un ensemble de « prédicats » et de « règles d'inférence ». L'argumentation donc permet la modification des opinions des autres acteurs afin qu'ils partagent les mêmes idées, objectifs et avis que l'acteur « argumentant ». Dans ce cas, les acteurs utilisent divers types d'arguments. Chacun de ces types correspond à des « pré-exigences » afin qu'il soit employé. L'argument sera utilisé par un acteur si et seulement si toutes ces conditions sont satisfaisantes. Dans ce cas l'acteur utilise une méthode pour choisir quel ensemble d'arguments il doit choisir dans une situation donnée.

Comme nous avons mentionné il existe différentes catégories d'arguments, nous citons : [Verrons 2004]

- « **Appels à une promesse passée** » : dans ce cas, l'acteur (négociateur) X fait rappeler à l'acteur (négociateur) Y un engagement antérieur relatif au sujet de négociation. C'est à dire

l'acteur *Y* s'est engagé dans une négociation passée à l'acteur (négociateur) *X* de fournir ou d'effectuer « un objet de négociation ».

Préconditions : il faut que l'acteur *X* procède à une vérification si un engagement relatif à un objet de négociation a été reçu antérieurement suite à une négociation établie avec succès.

- « **Promesse d'une récompense future** » : l'acteur *X* s'est engagé de réaliser l'objet de négociation pour un autre acteur *Y* à un moment à venir.

Préconditions : l'acteur *X* doit à l'obligation de chercher un désir de l'acteur *Y* pour un moment à venir, si possible un désir pouvant être contenté à l'aide d'une activité que *X* a la capacité de le faire alors que *Y* ne pouvant pas la réaliser.

- « **Appels au propre intérêt** » : l'acteur *X* pense que l'accord qui a été conclu sur l'objet de négociation est au profit de *Y* et essaye de persuader *Y* de ceci.

Préconditions : l'acteur *X* a l'obligation de chercher (ou déduire) un des désirs de *Y* qui donnera satisfaction si l'acteur *Y* a l'objet de négociation bien *X* a l'obligation de chercher un autre objet de négociation *ON'* qui a été fourni au préalable sur le marché et prouver que l'objet de négociation soumis est plus important que *ON'*.

- « **Appel à une pratique fréquente** » : l'acteur *X* pense que *Y* n'a pas accepté la suggestion vu que *Y* pense que la suggestion est en contradiction avec un de ses objectifs. Si tel est le cas alors, l'acteur *X* donne à *Y* l'exemple d'une méthode qui se fait fréquemment qui prouve que l'approbation de la suggestion n'est pas en contradiction avec l'objectif de *Y*.

Précondition : l'acteur *X* est dans l'obligation de chercher un autre agent qui possède un objectif identique à celui de *Y*, qui a été déjà accepté une suggestion similaire et qui a constaté que le consentement n'a pas compromis ses objectifs.

- « **Menace** » : le négociateur fait pression pour ne pas accepter à effectuer/fournir quelque chose qui est en contradiction avec les vœux de *Y*.

Préconditions : l'acteur *X* a l'obligation de chercher un des vœux de *Y* dont la satisfaction peut être directe par un objet de négociation que l'agent *X* peut fournir ou *X* doit chercher une action qui est en contradiction avec ce qu'il pense être un des vœux de *Y*.

### 3.4. Les différents systèmes de négociation automatisée

Plusieurs travaux liés à la négociation automatisée ont été réalisés et différents systèmes de négociation ont apparus. La plupart de ces travaux sont inspirés du protocole Contract Net et dont les principaux sont :

#### 3.4.1. Le Kasbah

Kasbah [Verrons 2004] a été proposé en 1996 par Pattie Maes. Ce système permet aux différents utilisateurs de créer sur Internet des agents permettant la négociation de l'achat et la vente des objets. Ces objets sont ordonnés par type.

Afin de créer un agent qui effectue l'achat ou bien la vente, l'utilisateur doit spécifier certains paramètres tel que : le type d'objet, la date de l'opération, le tarif souhaité, le plus petit tarif admissible ainsi que la méthode de négociation sélectionnée (exponentielle, quadratique, linéaire) déterminant le (calcul) de la progression du tarif dans le temps. L'utilisateur doit spécifier aussi si l'agent négociateur demande ou non son accord avant de clôturer la vente, de

plus il doit déterminer s'il aime être informé par *e-mail* lorsqu'il y aura un accord. L'opération physique (acteur humain) se déclenche lorsque l'acceptation aboutie.

Pour ce qui est de l'exécution, le Kasbah regroupe les agents qui ont des objectifs communs ensemble et leurs permet d'effectuer des échanges entre eux ; ces échanges s'effectuent 1 vers 1. Ce système affecte aux agents une période de temps (*time slice*) durant laquelle l'agent doit déterminer le tarif courant souhaité et l'agent avec qui il veut communiquer. Une fois ses choix sont précisés, il commence la communication. Les agents effectuent des échanges en utilisant un langage propre à Kasbah.

### 3.4.2. L'AuctionBot

L'AuctionBot représente un serveur d'enchères proposé en 1998 par Peter et Wellman [Verrons 2004]. Il permet donc à n'importe quel utilisateur d'Internet de participer à une session d'enchère. Ce système représente un framework qui propose à l'utilisateur de choisir parmi une grande variété d'enchère, l'enchère souhaitée. Egalement, il lui offre une interface web à travers laquelle l'internaute crée ses agents qui vont participer aux enchères.

L'AuctionBot permet de stocker les différentes enchères dans une base de données où chaque enchère est répertoriée dans un dossier structuré d'une façon hiérarchique et il gère à la fois plusieurs enchères. Pour la participation aux enchères, les usagers doivent s'enregistrer en créant un compte sur le serveur d'AuctionBot. Donc, afin d'apercevoir l'évolution de l'enchère l'utilisateur consulte son compte sur Internet ou bien il mentionne qu'il soit avertis par *e-mail*. Comme nous avons noté précédemment l'AuctionBot répertorie chaque enchère dans son catalogue approprié ce qui permet au vendeur de mettre son enchère dans le dossier existant ou allonger celui ci. Aussi, Il peut sélectionner s'il veut placer l'enchère dans une catégorie privée ou bien dans un dossier public

En ce qui concerne l'exécution, l'AuctionBot permet de collecter les enchères placées dans la base de données par les différents agents, puis déterminer l'emplacement de celle ci en utilisant des règles bien définies et enfin avertir les utilisateurs participants. Néanmoins, l'agent gestionnaire (initiateur) ne prend pas en charge l'exécution des opérations, la vérification de la fiabilité des utilisateurs, et l'imposition des communications. C'est un programme du système AuctionBot qui les effectue.

### 3.4.3. Le Fishmarket

Fishmarket [Noriega 1998] a été proposé par Pablo Noriega, il représente une « antenne d'enchères électroniques » qui permet la vente du poisson dans laquelle les acteurs participants peuvent être des acteurs humains ou bien des acteurs (agent) virtuels. Le marché de vente du poisson est un endroit dans lequel se passent diverses scènes où plusieurs participants s'impliquent en suivant certaines conduites bien déterminées. Parmi ces scènes, on trouve la scène enchère dans laquelle un « commissaire-priseur » déclare les tarifs des différentes caisses de poisson dans un ordre décroissant (enchère descendante) et l'ensemble d'acheteurs effectuent des offres. Mais, avant la vente du poisson, les vendeurs procèdent à la livraison au marché (la scène d'admission des vendeurs) où les acheteurs sont obligés de s'enregistrer (la

scène d'admission des acheteurs). De la même manière, lorsque la caisse de poisson est vendue l'acheteur la retire et passe donc par la scène des règlements des acheteurs alors que les vendeurs procèdent au retrait de leur paiement dans la scène des règlements des vendeurs après la vente de leur marchandise. Le système Fishmarket a été développé pour témoigner de la difficulté de ces interférences. Alors chacun des agents logiciels de ce système désigne un acheteur ou un vendeur ou bien le représentant du marché. Dans ce cas les différentes enchères sont traitées d'une façon séquentielle.

#### **3.4.4. Le système Adept**

Ce projet [Alty et al. 1994] a été réalisé afin de développer un modèle de négociation orienté service. Cette application est basée sur les travaux de « l'opérateur de télécommunications britannique » afin de déterminer le prix de l'installation d'un service demandé par les utilisateurs. Alors, dans le système Adept, les usagers seront représentés par des agents où ils emploient un protocole de négociation « réciproque » à travers lequel ils expriment leurs offres, ils évaluent et formulent leurs suggestions adverses. Chaque offre est caractérisée par un ensemble de caractéristiques (paramètres). La stratégie de chaque agent définit sa réaction en fonction de ces paramètres. Ce système est composé d'un module qui négocie les différents services. Chaque service est représenté par ensemble de caractéristiques par exemple: le service a un nom unique, un ensemble de données lui permettant d'être lancé, un ensemble d'informations qui décrit la manière dont il pourra être exécuté.

#### **3.4.5. Magnet (Multi Agent Testbed)**

Ce système [Collins et al. 1998] a été proposé par Collins et al. en 1998. Le but des auteurs est de concevoir, implémenter et analyser une architecture de marché Multi-agents généralisée, qui peut fournir un support explicite et intégré pour les interactions complexes entre agents, comme dans la prise de contrats automatisée. Ce système est basé sur le protocole Contract Net, alors la négociation se déroule en par trois phases :

- Annoncer l'offre
- Collecter les différentes enchères
- Notifier les résultats.

Les fournisseurs qui ont reçu l'annonce des offres, vont l'étudier puis décider selon leurs connaissances et leurs ressources s'ils vont effectuer une offre ou non. Chacun des fournisseurs a la possibilité de transmettre une multitude d'offres diverses pour une même annonce d'offres. Le client fixe une échéance durant laquelle les fournisseurs doivent envoyer leurs offres. Si le client ne reçoit pas une réponse cela veut dire que le fournisseur n'a pas fait d'offres. Lorsque toutes les offres sont perçues, le client utilise ses connaissances, ses préférences et ses contraintes temporelles afin de d'approuver ou refuser ces offres.

### **3.5. Les différents systèmes de négociation automatisée proposés en planification urbaine**

Dans le cadre de la planification urbaine, diverses recherches relatives à la négociation automatisée ont été effectuées. Nous citons :

Huang, Zhang and Wei [Huang et al. 2011] ont conçu un système multi-agent basé sur l'utilisation du protocole le plus connu dans le domaine de la négociation Contract Net afin d'optimiser le contrôle du réservoir des inondations dans une ville. Les auteurs ont amélioré le protocole de Contract Net en terme de l'efficacité de la communication interactive ce qui a amélioré l'efficacité du contrôle du réservoir.

Di Lecce, Amato, Soldo et Giove [Di Lecce et al. 2010] ont développé un système décisionnel basé sur la négociation coopérative entre les agents pour la planification des routes urbaines. Le but du système est d'être évolutif, flexible et capable de trouver l'itinéraire pour n'importe quel moyen de transport en considérant un grand nombre de contraintes (le risque hydrogéologique, le trafic urbain, le temps de transport, ...). Les auteurs ont proposé leur propre protocole de négociation afin d'atteindre ces buts.

Di Lecce and Amato [Di Lecce et Amato 2011] ont proposé une solution pour le problème de transport des matières dangereuses qui ont des effets négatif sur la santé des personnes et la qualité de l'environnement. Les auteurs ont développé leur propre processus de négociation entre différents agents intelligents qui négocient l'itinéraire le plus adéquat à survivre en prenant en compte différents paramètres tel que : la vitesse du véhicule, la position, l'équilibre de charge et les accélérations.

Takahashi, kanamori and Ito [Takahashi et al. 13] ont décrit un système de négociation décisionnelle pour le changement de route en cas de trafic urbain. Ce système est basé sur l'utilisation de la stigmergie anticipative pour estimer la position du véhicule et la négociation automatisée pour changer la route.

D'après les travaux cités précédemment, nous remarquons que les auteurs ont employé le mécanisme de la négociation pour résoudre différentes problématiques dans divers domaines de la planification urbaine ; nous constatons donc que la négociation constitue un outil efficace pour la planification urbaine. Cependant, à ce jour là, il n'existe aucun système ou protocole de négociation destiné à la résolution de la problématique d'évaluation et de sélection des projets urbains.

Cette problématique est considérée comme une problématique multi-acteurs où les décisions sont collectives, et elles peuvent prendre des formes variées où il est indispensable d'offrir aux différents acteurs un système de négociation automatisée afin de trouver un accord entre eux [Laurini 2002]. Selon Laurin [Laurini 2002] un tel système permet à un acteur principal de proposer un certain consensus aux acteurs concernés. Chacun d'eux doit évaluer cette proposition et donner son accord. Lorsque tous les acteurs ne sont pas d'accord, dans ce cas l'acteur principal peut imposer une solution.

C'est ce qui nous a inspiré et motivé à introduire le paradigme de la négociation automatisée dans notre processus décisionnel dédié à l'évaluation des projets urbains.

### 3.6. Conclusion

La négociation joue un rôle fondamental dans les cas où les personnes doivent solutionner les divergences pouvant menacer l'esprit de coopération. De ce fait nous avons essayé tout au long de cette partie de donner la plus part des principes qui permettent de bien comprendre la négociation entre les acteurs et sans oublier les concepts sur lesquels elle est basée. Nous

---

pouvons dire qu'avec l'introduction de la négociation dans notre processus décisionnel, nous pouvons résoudre les conflits entre les différents décideurs.

Dans le but d'améliorer notre système décisionnel (DSS) qui sera basé sur l'analyse multicritère et la négociation et d'assurer un bon fonctionnement en terme de fiabilité, accessibilité, disponibilité et évolutivité, nous proposerons d'offrir notre DSS comme une plateforme de service en le déployant dans un Cloud Computing. Dans le chapitre 6 nous développerons l'architecture Cloud proposée pour intégrer notre système décisionnel.

Nous détaillerons dans le chapitre qui suit les principes de base du Cloud Computing et ses diverses solutions.

# Chapitre 4

## Cloud Computing pour l'intégration du DSS

*«L'objectif de ce chapitre est de clarifier les concepts du Cloud Computing en abordant ses notions fondamentales et de comparer ses différentes solutions afin de choisir la plus adéquate. »*

---

### Sommaire

4.1. Introduction.....	78
4.2. Présentation du Cloud Computing.....	78
4.3. Les services Cloud.....	79
4.4. Les modèles de déploiement.....	81
4.5. La virtualisation dans le Cloud Computing .....	81
4.6. Les solutions Cloud open source .....	82
4.6.1. Solutions IaaS.....	83
4.6.2. Solutions PaaS.....	88
4.7. Les solutions Cloud propriétaires .....	89
4.7.1. Google.....	89
4.7.2. Salesforce.com .....	90
4.7.3. Amazon Web Services .....	92
4.8. Conclusion.....	94

## 4.1. Introduction

Incontestablement, Internet évolue de façon vélocité dès son invention. A présent, une "tendance" nouvelle vient d'apparaître dans le domaine de l'IT, c'est le Cloud Computing. Il s'agit d'une technique basée sur le « WEB 2.0 » qui fournit aux organisations des opportunités afin de diminuer les dépenses dues à l'exploitation des logiciels en les utilisant en ligne directement [Kherbache 2010]. Le Cloud Computing permet d'interconnecter et coopérer les différentes ressources informatiques qui sont localisées dans diverses organisations externes, internes ou bien mixtes et dont la manière d'y accéder repose sur des règles (protocoles) et des normes d'Internet. Il est devenu, l'objet le plus discuté actuellement dans le domaine des ITs parce que son rôle devient essentiel dans les différentes opérations informatiques des sociétés et des organismes.

Les systèmes décisionnels (DSS) représentent l'une des applications qui peut être déployée dans un Cloud. Dans le contexte de notre thèse qui est la planification territoriale urbaine quelques systèmes décisionnels basés Cloud ont été développés ; On trouve : dans le domaine du transport urbain Miah et Ahamed [Miah et Ahamed 2011] qui ont développé un DSS déployé dans un Cloud public qui aide les autorités à obtenir on-demand des informations sur le comportement des conducteurs sur les routes australiennes. Dans le domaine de la gestion d'eau, les auteurs dans [Montalvo Arango et al. 2014] ont suggéré d'intégrer un système décisionnel dédié à la distribution d'eau dans une ville urbaine dans un Cloud privé. L'utilisation du Cloud permet d'accélérer le processus de distribution d'eau. Dans le domaine du transport maritime et environnemental, PROMATECH [PROMATECH 2014] a proposé un système décisionnel basé Cloud qui aide les entreprises pétrolières et gazières à se décider sur l'allocation des navires dans le but d'optimiser l'impact environnemental et les émissions de carburant.

Dans notre thèse, comme indiqué précédemment nous nous intéressons au domaine de l'évaluation des projets urbains, et notre but est de proposer un DSS basé Cloud pour aider les différents décideurs d'une ville urbaine à choisir le meilleur projet urbain. Alors dans cette partie, nous allons présenter dans un premier temps les notions fondamentales du Cloud Computing, dans un second temps nous étudierons les trois services principaux, sur lesquels le Cloud Computing repose ainsi que les quatre types du Cloud. Et la dernière partie sera consacrée à la présentation des solutions Cloud open source et propriétaires afin de déterminer les solutions que nous utiliserons pour la création d'une architecture Cloud privé dédié à l'intégration de notre DSS.

## 4.2. Présentation du Cloud Computing

Le Cloud Computing se traduit le plus souvent en français par "informatique dans les nuages", faisant référence aux technologies d'Internet lequel Cloud est souvent représenté schématiquement par un nuage comme le montre la Figure 4.1. Le Cloud Computing est une notion abstraite permettant de rassembler les différentes technologies qui délivrent les services. Selon le NIST [Mell et Grance 2011] le Cloud Computing peut être défini comme étant un modèle qui permet à la demande (on demand) d'accéder facilement aux ressources partagées (tel que les mémoires, les serveurs, les applications...) via un réseau.



Figure 4.1. Cloud Computing.

Le Cloud Computing donc, correspond à la création et à la manipulation des applications et des services atteignables seulement par Internet. Alors, les usagers dépendent seulement d'Internet pour accéder à leurs services. Ils ont besoin uniquement d'un navigateur Internet c'est à dire ils peuvent atteindre leurs applications sans procéder à l'installation de quoi que ce soit. En utilisant le Cloud Computing, les données seront stockées d'une manière permanente dans des serveurs nombreux et extrêmement puissants.

Selon le NIST [Mell et Grance 2011], le Cloud Computing possède six caractéristiques :

- Self-service à la demande : un client peut s'approvisionner d'un service de façon unilatérale et sans avoir besoin d'interactions avec un tiers humain du côté fournisseur.
- Large accès au réseau haut débit : c'est la disponibilité des services via le réseau et leurs accessibilités grâce aux mécanismes standards de communication.
- Mise en commun des ressources : les ressources du fournisseur sont regroupées afin de servir plusieurs clients locataires en leur offrant un ensemble de ressources virtuelles et physiques. Ces ressources sont affectées ou réaffectées d'une manière dynamique selon les besoins du client. Il y a un sentiment d'indépendance de la localisation dans le sens que le client ne connaît pas la localisation des ressources. Par contre, il est capable de déterminer à un niveau d'abstraction supérieure la localisation telle que le « Datacenter ».
- Elasticité rapide : cela veut dire la capacité d'étendre ou de réduire de façon rapide et efficace les ressources du Cloud pour répondre aux exigences des demandes.
- Service mesuré : pour la facturation des services utilisés par les clients, le concept pay-per-use est utilisé, ce qui veut dire que l'utilisateur ne paye que ce qu'il a utilisé comme ressource.

### 4.3. Les services Cloud

Le Cloud Computing se décline sous la forme de trois offres : SaaS, PaaS et IaaS : [Yassa 2014]

- **SaaS (Software as a Service):** le SaaS est le plus célèbre dans le domaine du Cloud Computing, Il représente un modèle qui déploie un ensemble d'applications. Ce modèle permet

au fournisseur de louer une application en tant que service « on demand » à ses clients au lieu qu'ils achètent des licences. De cette manière, les usagers finaux n'installent plus les logiciels existants sur leurs machines ; ce qui permet de réduire la tâche de la mise à jour. Ce modèle de Cloud a l'avantage alors de transformer les dépenses logicielles fixes en variables, de plus on a plus besoin d'acheter un logiciel pour chacun des employés de l'entreprise.

- **PaaS (Platform as a Service)** : le PaaS représente un modèle qui comprend tous les outils et les éléments essentiels pour la création, le déploiement, la gestion du cycle de vie et la livraison des applications accessibles par Internet sans télécharger ou installer des logiciels. Cette architecture est utilisée par les responsables informatiques, les développeurs des applications, ou bien les usagers finaux.

- **IaaS (Infrastructure as a Service)**: l'IaaS offre à la demande un ensemble de ressources matérielles (tel que les clusters, le stockage ou la mémoire des informations) qui peuvent être louées par le client avec un tarif relié au taux d'utilisation (occupation). Ces dernières sont placées à distance dans des différents « Datacenters ». L'IaaS permet également aux administrateurs des entreprises d'accéder aux serveurs ainsi que leurs configurations.

Les trois acronymes SaaS, PaaS et IaaS sont alors, les trois services importants du Cloud Computing. Chaque service constitue une base pour l'autre. La majorité des experts s'entendent pour modéliser le Cloud Computing sous une architecture pyramidale sur laquelle les divers services sont décrits comme des couches consécutives.

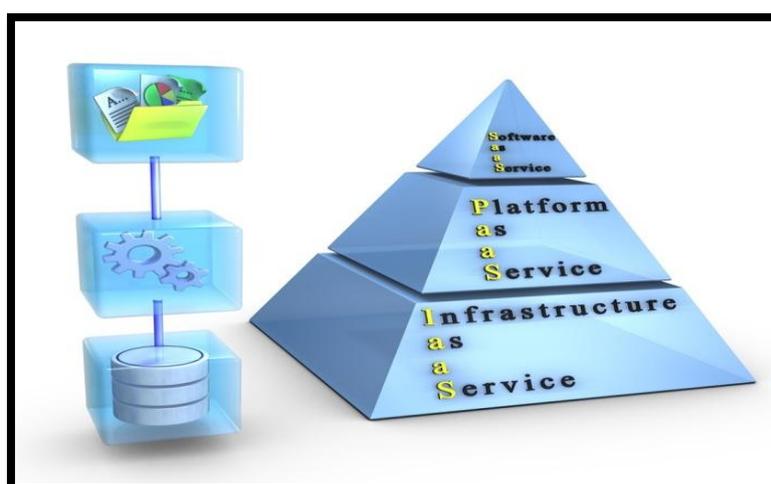


Figure 4.2. L'architecture pyramidale IaaS, PaaS, SaaS [7].

La base de l'architecture Cloud est accordée au service IaaS, vu qu'il permet aux architectes réseaux de manipuler les équipements d'un réseau par le biais des serveurs virtuels (machines virtuelles). Sur cette couche, on trouve le service PaaS. Celui-ci repose sur la flexibilité du modèle IaaS afin d'offrir aux développeurs des applications une plateforme de développement qui mémorise les ressources et les informations, et distribue les calculs en toute crédibilité et souplesse. Au sommet de l'architecture, se trouve le service SaaS qui sera employé par le grand public. A cause de sa puissance et sa souplesse, ce modèle assure une meilleure utilisation des capacités de la machine par la virtualisation des informations mémorisées [12].

## 4.4. Les modèles de déploiement

Selon la définition du Cloud Computing donnée par le NIST [Mell et Grance 2011], il existe quatre modèles de déploiement des services de Cloud, à savoir : le Cloud privé, public, hybride et communautaire.

- **Le Cloud privé** : est exploité par une seule société ou organisation. Son fonctionnement est assuré par cette même organisation ou bien par une tierce partie. L'infrastructure est géographiquement installée dans les locaux de la société ou ailleurs.
- **Le Cloud public** : ce Cloud est destiné au grand public. Les usagers donc peuvent atteindre leurs applications et services par Internet sans qu'ils sachent où exactement sont stockées leurs informations ni où sont opérés leurs traitements. L'hébergement et le stockage des ressources et des bases de données de l'utilisateur se fait dans n'importe quel Datacenter du fournisseur et même ils peuvent circuler d'un « Datacenter » à un autre dans le but de l'amélioration des capacités du fournisseur.
- **Le Cloud communauté** : les Clouds communautés offrent à plusieurs structures la possibilité d'assurer le partage des ressources informatiques en mode Cloud. Ces ressources vont être exclusivement utilisées par ces structures. Ce type de Cloud est géré par les structures membres ou bien par un fournisseur externe.
- **Le Cloud hybride** : le Cloud hybride est l'utilisation de plusieurs Clouds, publics ou privés. On peut ainsi déporter nos applications vers un Cloud public qui consommera des données stockées et exposées dans un Cloud privé, ou bien faire communiquer deux applications hébergées dans deux Clouds privés distincts, ou encore consommer plusieurs services hébergés dans des Cloud publics différents. Dans tous ces cas de figure, nous avons affaire à la notion de Cloud hybride.

## 4.5. La virtualisation dans le Cloud Computing

La virtualisation est le fait de faire cohabiter une multitude d'applications ou de systèmes d'exploitation dans la même machine ou hardware. Ce type de système virtuel s'appelle, serveur privé virtuel (VPS) ou encore environnement virtuel (VE). La virtualisation a révolutionné la technologie des datacenters grâce à un ensemble de techniques et d'outils qui facilitent la fourniture et la gestion de l'infrastructure dynamique des datacenters. Elle est devenue une technologie essentielle des environnements Cloud Computing. La virtualisation peut être définie comme l'abstraction des quatre ressources informatiques ; stockage, puissance de traitement, mémoire et réseau ou I/O. En d'autres termes, c'est une vue abstraite et logique des ressources physiques. La Figure 4.3 représente les différentes couches d'un système de virtualisation.

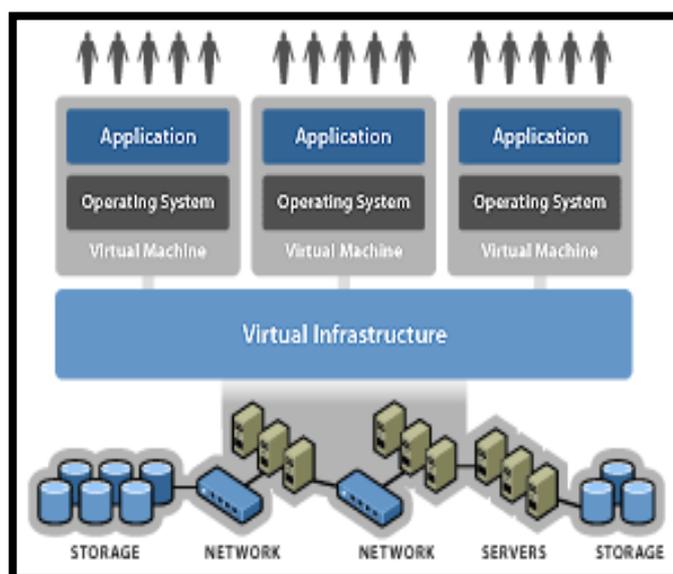


Figure 4.3. Différentes couches de la virtualisation.

Les systèmes de virtualisation peuvent être divisés en deux types, voire la paravirtualisation et la full virtualisation ou la virtualisation complète.

- La paravirtualisation : représente une technique de la VM (Virtual Machine). Dans ce type de virtualisation le système d'exploitation « esclave » ou « invité » interagit et collabore avec le système d'exploitation « maître » ou « hôte » dans le but d'assurer une haute performance. Dans ce cas le SE « invité » est "conscient" de la virtualisation.
- La virtualisation complète : représente une technique de la VM (Virtual Machine). Dans ce type de virtualisation, le système d'exploitation (SE) « invité » n'effectue pas une interaction ou une collaboration avec le système d'exploitation (SE) « hôte ». Donc le SE « invité » n'est pas "conscient" de la virtualisation. Alors, pour assurer la virtualisation dans ce cas, il est nécessaire d'utiliser des recommandations matérielles spécifiques.

La virtualisation représente une variété de concepts et de technologies différentes, qui diffèrent par leurs mises en œuvre, la pertinence pour une utilisation pratique, la fréquence d'utilisation et les outils virtualisés. On trouve : la virtualisation des systèmes d'exploitation, la virtualisation des plateformes, la virtualisation de stockage, la virtualisation d'outils réseaux et la virtualisation des applications.

Dans cette thèse, notre intérêt porte sur la virtualisation des applications, des plateformes et des systèmes d'exploitation. Dans la partie ci-après nous allons présenter et comparer les diverses solutions Cloud open source (offrant les niveaux de virtualisation désirés) afin de déterminer les solutions les plus adéquates pour le développement de notre propre architecture Cloud qui sera dédiée à l'intégration de notre système décisionnel (DSS).

## 4.6. Les solutions Cloud open source

Dans cette partie, nous allons présenter les solutions Cloud open source de type infrastructure (IaaS) et plateforme (PaaS).

### 4.6.1. Solutions IaaS

De nombreuses solutions sont considérées comme étant des solutions d'infrastructures qui permettent d'assurer une gestion simplifiée des architectures matérielles, les plus complexes. Dans ce contexte l'open source intervient de façon significative apportant des solutions dans le domaine du Cloud Computing. Voici quelques outils d'infrastructure open source :

#### ➤ Eucalyptus

C'est un Cloud open source proposé par l'Université de Californie [6]. Cet outil est le plus célèbre parce qu'il est incorporé dans les distributions Debian et Ubuntu Server. Il est écrit en langages « Java », « C » et « Python » et il permet la création des Clouds de type infrastructure (IaaS) hybride ou privé. L'Eucalyptus supporte les plateformes de virtualisation (hyper-viseurs) « KVM » et « Xen » et les VM Linux. Comme le montre la Figure 4.4, il comprend cinq composants :

- « Cloud Controller » : il représente un point d'entrée à travers lequel les usagers ainsi que les administrateurs accèdent au système. Son rôle principal est la gestion du système.
- « Cluster Controller » : il permet de conserver et de stocker toutes les données liées aux ressources physiques (matérielles comportant des informations).
- « Node Controller » : il coopère avec l'hyperviseur afin d'assurer la gestion des VMs. Lorsque Cloud Controller demande une attribution des ressources physiques, le « Cluster Controller » lui accorde ces ressources en requérant les « Nodes Controllers ».
- « Walrus » : il permet la gestion de l'accès au service de stockage.
- « Storage Controller » : il travaille avec le « Walrus » et il assure le stockage des informations des usagers ainsi que les images (systèmes d'exploitation) des machines virtuelles.

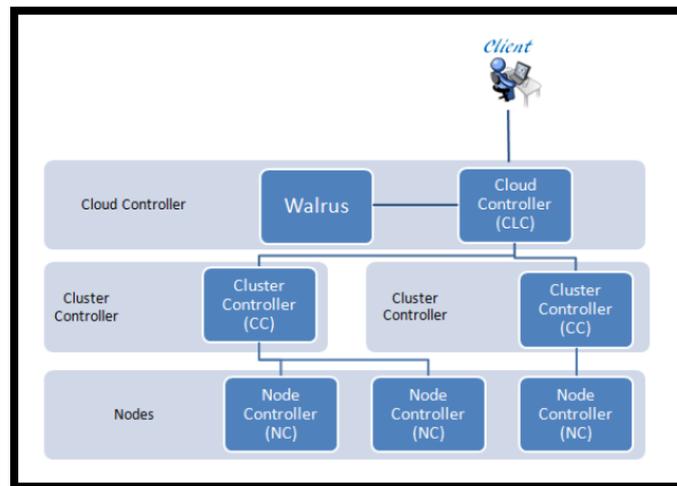


Figure 4.4. Architecture d'Eucalyptus [Mahjoub 2011].

#### ➤ OpenNebula

C'est un Cloud IaaS open source, il permet le déploiement des Clouds hybrides, publics et privés [8]. L'OpenNebula représente une solution écrite en langage « Ruby », « C++ » et « Shell » et qui supporte les hyper-viseurs « KVM » et « Xen ». Il permet de gérer les VMs et les ressources physiques d'une manière centralisée, de répartir les charges, et d'entendre les capacités en ajoutant des serveurs physiques. OpenNebula est composé de trois couches principales comme le représente la Figure 4.5 :

- « Tools » : il représente les outils permettant de gérer OpenNebula ;
- « Core » : il comprend un ensemble d'éléments permettant le contrôle du stockage, du réseau virtuel et des VMs.
- « Drivers » : ce sont des pilotes spécifiques qui permettent d'assurer la coopération entre l'infrastructure de Cloud et OpenNebula.

Les machines « Front end » et « Node » sont interconnectées entre elles via un réseau privé.

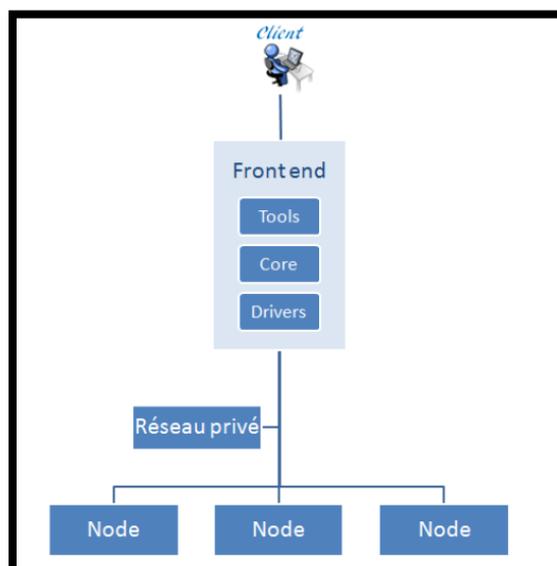


Figure 4.5. Vu interne de l'architecture d'OpenNebula [Mahjoub 2011].

### ➤ Nimbus

C'est un outil qui déploie un Cloud IaaS [8]. Il supporte les hyper-viseurs « KVM » et « Xen ». Il peut assurer le déploiement des services de stockage en Cloud. Ce Cloud est composé d'ensemble d'éléments comme le représente la Figure 4.6. Ces éléments sont interconnectés entre eux via un réseau privé.

- « Client-supported modules » : il est employé pour la gestion des clients du Cloud. Il contient le « Cloud client module », le « context client module », le « reference client module » et l'« EC2 client module ».
- « Service-supported modules » : il offre tous les services Cloud. Il contient le « Web service resource Framework module », le « context broker module ».
- « Background resource management modules » : il permet la gestion des ressources physiques du Cloud. Il contient divers modules tel que: l'« IaaS gateway module », le « workspace pilot module », le « workspace resource management module », le « EC2 module », le « workspace controller », et le « workspace service manager module ».

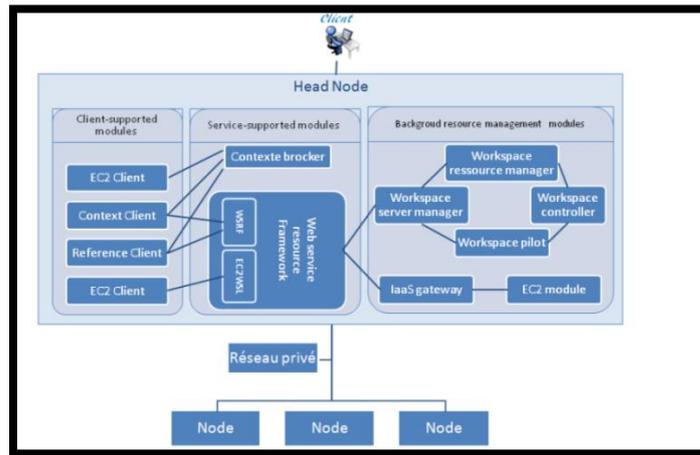


Figure 4.6. Vu interne de l'architecture de Nimbus [Mahjoub 2011].

➤ **Xen Cloud Platform (XCP)**

C'est un Cloud de type IaaS open source proposé par la compagnie « Xen » [11]. Il offre une plateforme gratuite et open source permettant la construction des services du Cloud et la communication entre eux. Xen Cloud Platform est composé d'un ensemble d'éléments, nous citons : (Figure 4.7)

- « XCP Host » : il représente le système d'exploitation (SE) Xen.
- « Master XCP Host » : il permet la gestion des « XCP hosts ».
- « Shared storage » : est un constituant optionnel dans lequel les VMs sont stockées. Il offre la possibilité aux administrateurs de faire le déplacement des VMs d'un « XCP host » à un autre.

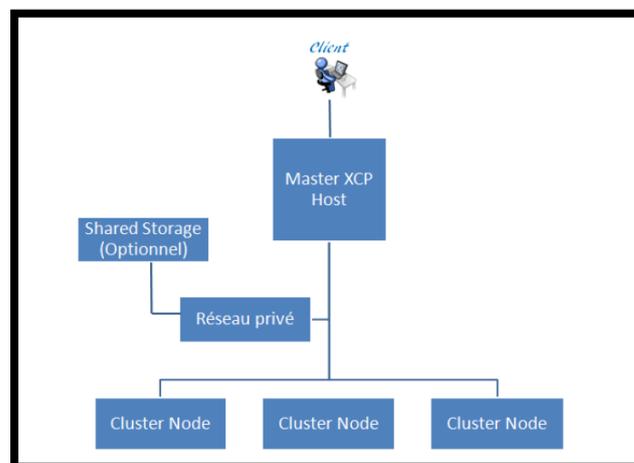


Figure 4.7. Architecture de XCP [Mahjoub 2011].

➤ **AbiCloud**

AbiCloud [1] a été proposé par la compagnie Abiquo. Il représente une plateforme qui permet la création et la gestion des Cloud hybrides, publics et privés. AbiCloud est constitué de trois composants élémentaires comme le représente la Figure 4.8:

- « AbiCloud server » : il représente le système d'exploitation(SE) Xen.
- « AbiCloud WS (AWS) » : il gère les VMs.
- « Virtual System Monitor (VSM) WS » : il suit la totalité de l'infrastructure virtuelle du Cloud.

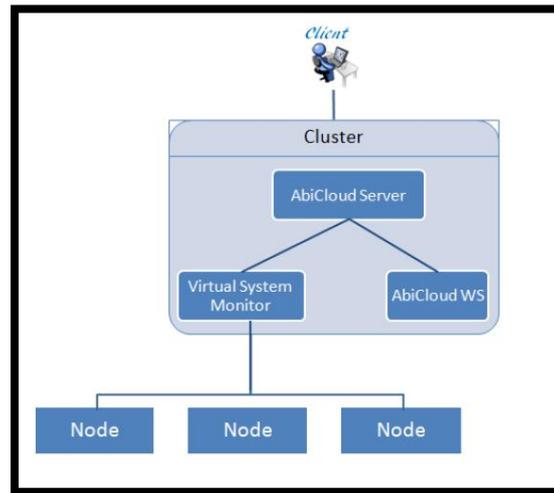


Figure 4.8. Architecture d'Abi Cloud [Mahjoub 2011].

### ➤ OpenStack

C'est le Cloud open source de type IaaS [9], son rôle est de construire des Clouds public et privé. L'objectif principal d'OpenStack est d'aider les différentes structures à mettre en place un système de stockage et de serveur virtuel. Son installation se fait sur un système d'exploitation opensource tel que Debian ou Ubuntu et sa configuration se fait en ligne de commande. Il est constitué de trois éléments essentiels qui sont reliés entre eux comme le montre la figure 4.9.

- « OpenStack Compute (Nova) » : il est employé pour le démarrage des machines virtuelles (VM) et la configuration du réseau pour chaque VM.
- « OpenStack Object Storage (Swift) » : il représente un système dédié au stockage des différents objets qui ont des capacités massivement évolutives.
- « OpenStack Imaging Service (Glance) » : il permet la création des images (SE) des VMs.

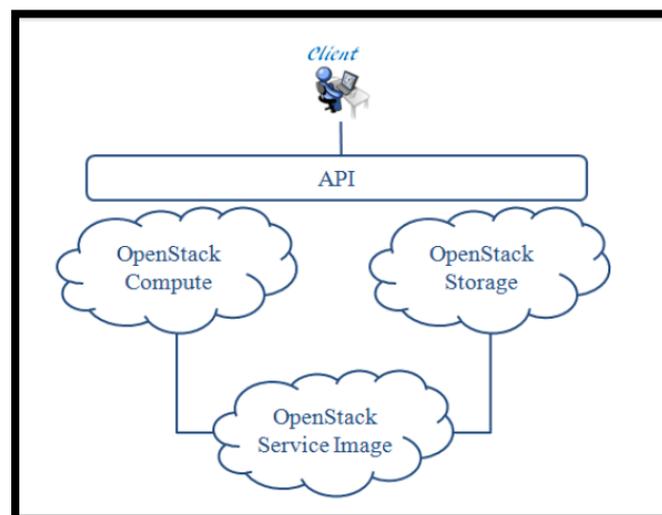


Figure 4.9. Architecture d'OpenStack [Mahjoub 2011].

### Synthèse

Une comparaison entre les différentes solutions IaaS est représentée par le Tableau 4.1.

	Eucalyptus	OpenNebula	Nimbus	Xen Cloud Platform	AbiCloud	Openstack
<b>But</b>	Une réponse opensource pour le Cloud commerciale EC2	Un Cloud privé pure	Solution scientifique du Cloud Computing	Configuration automatique et maintenance des Cloud Computing	La gestion des Clouds publics et privés dans des environnements hétérogènes	Créer et offrir des fonctionnalités de Cloud Computing
<b>Architecture</b>	- Hiérarchique - Cinq composants - Supporte multiple cluster	- Centralisé - Trois composants - Minimum deux serveur	- Centralisé - Trois composants - Minimum deux serveurs	- Centralisé - Trois composants - Minimum deux serveurs	- Centralisé - Trois composants - Minimum deux serveurs	- Intégration des deux composants - OpenStackobjet et OpenStackcompute
<b>Systèmes d'exploitation supportés</b>	Linux (Ubuntu, Fedora, CentOS, OpenSUSE et Debian)	Linux (Ubuntu, RedHat Enterprise Linux, Fedora et SUSE Linux Enterprise Server)	La plupart des distributions Linux	Linux (Fedora, RedHat, CentOS et Suse Linux Enterprise Server) - Windows 7	Linux (Ubuntu et CentOS) - Mac OS - Windows XP	- Linux et récemment Windows - Edge x86 Processor
<b>Langage de Programmation</b>	Java, C, et Python	Java, Ruby, C++	Python, Java	Caml	Java, Ruby, C++, Python	Python
<b>Interface utilisateur</b>	EC2 WS API Outliste que : Hybrid - Fox, ElasticFox	EC2 WS API OCCL API	EC2 WS API Nimbus WSRF	XenCenter et Versiera (application commerciale pour Windows)	Interface Web implémentée avec Adobe Flex	Interface Web
<b>Emplacement des VMs</b>	Node controller	Cluster node	Physical nodes	XCP Host	AbiServer	OpenStack Compute

Tableau 4.1. Solution IaaS open source de Cloud [Akbi et Zehri 2013]

Dans la section précédente, nous avons donné une idée sur les logiciels qui permettent la création des solutions de Cloud infrastructure (IaaS). La mise en œuvre d'un environnement de ce type de Cloud pour des objectifs de recherche implique principalement le choix d'un Cloud qui soit libre (open source), sécurisé, jeune, innovant, modulaire, facile à déployer et à installer, représente une plateforme de développement (sur laquelle nous pouvons déployer un Cloud PaaS), adapté à n'importe quel type d'infrastructures déjà existantes, destiné à toutes les tailles d'organisation et enfin bien documenté.

Alors le Cloud qui répond et correspond le mieux à nos besoins, c'est la solution OpenStack. Dans la Figure 4.10 nous présentons la proportion d'utilisation du Cloud OpenStack en comparaison aux autres solutions Clouds d'après « Zenoss.com ».

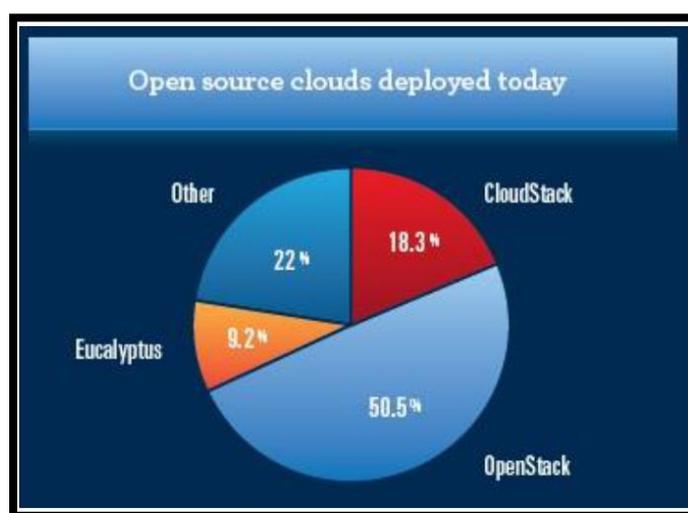


Figure 4.10. Pourcentage d'utilisation d'OpenStack [13].

#### 4.6.2. Solutions PaaS

D'autres solutions PaaS open source ont été proposées, nous citons :

➤ **CloudFoundry**

Cloud Foundry [5] est une plateforme ouverte en tant que service ; elle permet d'offrir un choix de Cloud, de développer des frameworks et des services d'applications. Cloud Foundry rend plus rapide et plus facile à construire, tester et déployer des applications à grande échelle.

➤ **Cloudify**

Cloudify [4] est un outil libre proposé par la compagnie « GigaSpaces Technologies », il permet la fabrication d'une plateforme personnalisée. Son rôle principal est de déployer une application dans un Cloud et gérer son cycle de vie.

➤ **CloudBees**

CloudBees [2] fournit une plate-forme intégrée et évolutive basée sur des normes pour les développeurs Java qui souhaitent développer et déployer des applications Web dans un environnement de Cloud Computing.

## Synthèse

Une comparaison entre les différentes solutions PaaS est représentée par le Tableau 4.2.

	CloudBees	Cloudify	Cloudfoundry
Technologies Supportées	Java	Java, .Net, Groovy, Ruby, C++, Node.JS, Spring, chef	Java, Spring, Groovy/Grails, Sinatra, Node.JS, Netvia extention.
Hebergement en Cloud	OpenStack	Supporte le déploiement sur n'importe quel Cloud (privé/public)	AWS, Azure, Rackspace
Repositories	Maven, Git, SVN	<a href="https://github.com/cloudifyso/urce">https://github.com/cloudifyso/urce</a>	<a href="https://github.com/cloudfoundry">https://github.com/cloudfoundry</a>
Base de données	MySQL	Cassandra, mongoDB, MySQL, HSQL	mongoDB, MySQL, REDIS
Plugins	300	Non	Non
API	REST API et CLI	CLI, REST, Web	CLI

Tableau 4.2. Solution PaaS open source de Cloud [10].

Dans la section précédente, nous avons donné un aperçu sur les logiciels qui permettent la création des solutions de Cloud plateforme (PaaS). La mise en œuvre d'un environnement de ce type de Cloud pour des objectifs de recherche implique principalement le choix d'un Cloud qui soit libre (open source), jeune, facile à déployer et à installer, peut être hébergé sur le Cloud OpenStack, et enfin très bien documenté

Alors le Cloud qui répond et correspond le mieux à nos besoins, c'est la solution CloudBees.

## 4.7. Les solutions Cloud propriétaires

Dans cette section nous présenterons les solutions Cloud propriétaires mais nous n'employons aucune de ces solutions dans notre approche vu qu'elles sont propriétaires, payantes et non open source. Afin de faire le tour de tous ces modèles de livraison de services, nous allons nous concentrer le plus sur un fournisseur pour chacun des types de livraison

### 4.7.1. Google

Google est l'un des plus grands fournisseurs de services Cloud, il assure les trois modèles de fourniture en service, voire IaaS, PaaS grâce à Google App Engine et SaaS grâce à Google Apps. Google offre une très grande variété de service SaaS tel que Google Search Engine, Google Maps, la messagerie Gmail, Google Calendar, Google docs, Google traduction, Google Cloud Connect pour Microsoft ...

Dans cette partie, nous allons nous concentrer plus sur la partie PaaS, ce qui veut dire Google App Engine et un peu sur la partie IaaS, car App Engine est l'un des principaux fournisseurs, pour ce type de service.

### ➤ **Google App Engine**

Google App Engine offre des services PaaS, en fournissant un environnement de programmation, de développement et d'exécution, avec un support d'outils, dans le but de développer, de tester et d'exécuter des applications Web Java ou Python, sur une infrastructure élastique, évolutive et qui s'adapte parfaitement aux besoins de ces dites applications. Autre avantage est la libération des développeurs de toute tâche d'administration. Du coup, ils peuvent se concentrer uniquement au développement des applications. En plus ces dernières bénéficient de la fiabilité et des technologies d'évolutivité utilisées par Google, pour ses propres applications.

En plus des bibliothèques Java et Python et les API standards qu'il a l'habitude de fournir, Google App Engine offre tout un autre ensemble de services et d'API, type Cloud, généralement accessibles via une interface Java, pour faciliter la portabilité et l'utilisation des applications, comme Memcache, Datastore et bien d'autres services. L'appel à certaines de ces API est fait à la demande ; dès lors il pourrait être automatique, dans le cas où il est fait par les mécanismes et les technologies de l'infrastructure elle-même. Un exemple de service est celui de l'authentification et la gestion des droits d'utilisateur : l'authentification des utilisateurs des applications Web est réalisée grâce à des comptes Google, pour lesquels une interface spécifique est valable. Les utilisateurs sont redirigés vers la page de login Google où ils pourront s'identifier. Un système de gestion basique distingue les utilisateurs normaux des administrateurs et contrôle ainsi l'accès aux fonctions et données privées.

Le problème majeur de ce type de service (PaaS), est l'utilisation des technologies propres au fournisseur, ce qui limite la portabilité et la réutilisation des services. En revanche, pour Google App Engine, une plateforme compatible et open source appelé AppScale, qui consiste en un projet de l'Université de Californie Santa Barbara mais pris en charge par Google après, a émergée [Baun et al. 2011]. Cette dernière offre la possibilité aux applications Web de migrer vers EC-2.

Un autre moyen de ré-implémentation de Google App Engine est TyphoonAE 1. Il permet l'utilisation des applications d'App Engine sur n'importe quel environnement Linux ainsi que sur Mac OS.

### ➤ **Google Storage**

Google Storage est un service payant de type IaaS, car il permet la location "à la demande" de l'espace mémoire pour le stockage des données et objets Web dans les Datacenters de Google. L'invite de commande GSUtil offert par Google permet aux utilisateurs, également ceux d'Amazon Web Services, le contrôle de leurs données : création, suppression, transaction et toute manipulation.

## **4.7.2. Salesforce.com**

C'est un fournisseur de services Cloud concernant la gestion des relations clients Customer Relationship Management (CRM). Salesforce.com est donc un fournisseur SaaS, mais il a également des services PaaS, Plateforme de développement, pour que les tierces parties (autres fournisseurs ou individus) peuvent développer et déployer leurs propres applications sur cette dite plateforme. Salesforce.com consiste en trois parties majeures :

- Les offres SaaS, principalement l'application Salesforce qui est une application CRM.
- Forces.com est le nom de la plateforme responsable des offres PaaS.

– AppExchange qui représente le dépôt des applications développées pour Salesforce.com.

### ➤ **Salesforce.com CRM**

Ce « .com » représente le principal initiateur des applications CRM, ses prestations peuvent être divisées en cinq applications de base :

- Sales : c'est l'application la plus populaire dans l'ensemble des offres Cloud du fournisseur Salesforces.com, appelé aussi Salesforce. Elle est utilisée par plus de 1,1 million de personnes à travers le monde. Les raisons de cette célébrité sont la grande simplicité de l'utilisation ainsi que celle de la personnalisation. Elle a l'avantage de permettre aux compagnies, de mieux faire la gestion des clients et de ce fait libérer leurs représentants des tâches administratives, pour mieux se concentrer sur la vente.
- Marketing : avec Salesforce.com CRM Marketing, les développeurs de solutions marketing peuvent mettre et déployer les dernières technologies Web de façon à collaborer avec les applications Salesforce.
- Service en raison d'être sur le Web, le Service Cloud permet aux compagnies une collaboration instantanée, en temps réel, en plus du partage des données, des tâches et des contacts entre partenaires.
- Collaboration : Salesforce.com CRM aide les organisations à réaliser un travail plus efficace avec les clients, les partenaires et les employés, en créant une atmosphère de collaboration et d'échange sur le Cloud.
- Analytique : Forces.com offre un rapport en temps réel, un calcul et des outils, ce qui permet de réaliser une bonne optimisation, prise de décisions et l'allocation de ressources.
- Personnalisation des applications : la personnalisation des applications peut être faite rapidement ; en s'appuyant sur un modèle de données, un modèle de partage et une interface utilisateur.

Salesforce.com CRM est une offre SaaS, elle fournit une solution Web pour ventes, Marketing, gestion des partenaires et autres et elle est facturée généralement par mois ou par an.

### ➤ **Forces.com**

Forces.com est le nom de l'offre PaaS, il permet aux clients, développeurs ou vendeurs de software indépendants (ISVs), de développer leurs propres applications et les déployer et exécuter sur l'infrastructure du Salesforce.com. Considéré comme le premier PaaS au monde [Velte et al. 2010].

La plateforme Forces.com contient un ensemble d'outils de développement et un environnement de programmation adapté aux besoins appelé Apex. D'autres caractéristiques sont le support de développement de l'interface utilisateur avec Visualforce, l'intégration des procédures de test logiciel et la connexion des services Web externes via une API spécialisée. Les développeurs qui veulent apprendre comment programmer sur Forces.com, peuvent avoir de l'aide sous formes de tutoriels, manuels, documents référentiels et exemples de code sur les portails Web de la communauté.

### ➤ **AppExchange**

AppExchange est un dépôt pour les applications développées pour Salesforce.com donc grâce à AppExchange les organisations sont capables d'ajouter facilement de nouvelles applications à leur déploiement Salesforce.com. Une option (outil) appelée "Get It Now" permet à cette application d'être instantanément disponible à tous les adhérents d'un compte Salesforce.

Une fois installée, cette application pourrait être personnalisée pour fournir à l'utilisateur, uniquement les applications qui l'intéressent, donc qui sont dans le domaine applicatif du client, selon ses besoins.

### 4.7.3. Amazon Web Services

Amazon peut-être le fournisseur de services Cloud le plus connu, car il est considéré comme étant l'un des principaux fondateurs du Cloud Computing. Il offre un grand nombre de services Cloud et sur tous les niveaux, IaaS, PaaS et SaaS.

#### ➤ Amazon EC2 (Amazon Elastic Compute) Cloud

Amazon EC2 représente un service Cloud accessible via Internet qui offre une capacité de calcul variable selon les besoins. Et ce par le déploiement des VMs chez le prestataire Amazon pour profiter de leurs puissances de traitement offertes. La technologie de Virtualisation utilisée est issue de la technologie Open Source Xen. Le Cloud Amazon EC2 a l'avantage de permettre aux développeurs d'accéder aux ressources informatiques via le grand Web.

EC2 permet d'avoir et de configurer la capacité (puissance) via une interface. Cette interface fournit également un contrôle complet sur les ressources informatiques du côté utilisateur. Elle permet aussi, l'exécution des applications sur la structure informatique d'Amazon et profiter des capacités de cette dernière. Le Cloud EC2 sert à réduire le temps nécessaire pour l'obtention et le démarrage de nouvelles VMs, ce qui permet à la puissance de calcul d'être élevée ou abaissée rapidement selon les besoins au fur et à mesure des modifications. En outre, ce Cloud offre aux développeurs des outils et des éléments nécessaires pour la construction des applications souples.

#### ➤ Amazon S3 (Simple Storage Service)

Amazon S3 est un Cloud qui offre le stockage via Internet, il utilise une structure de stockage très simple. Il a l'avantage de permettre aux développeurs d'accéder aux ressources informatiques via le grand Web.

Comme EC2, le Cloud S3 fournit une interface simple pour le stockage et la récupération des données quelque soit leur taille ou leur quantité via le Web sans contraintes géographiques ou temporelles.

Amazon offre également des services de base de données, grâce à Amazon SimpleDB. C'est un stockage non relationnel de données. Il travaille en étroite collaboration avec Amazon S3 et Amazon EC2. Il fournit les fonctions de base pour une base de données, voire l'indexation et l'interrogation des données.

Le développeur ne fait que stocker et interroger les objets de données via des requêtes de services Web, et Amazon SimpleDB s'occupe du reste, ce qui représente un avantage par rapport aux bases de données relationnelles traditionnelles, car ces dernières nécessitent des dépenses importantes au départ. Elles sont également complexes à concevoir et nécessitent souvent l'emploi d'un administrateur de base de données.

➤ **Amazon CloudFront**

Amazon CloudFront est un service de diffusion du contenu, il permet aux développeurs et aux sociétés une distribution facile du contenu, vers les utilisateurs finaux, avec des vitesses de transfert de données élevées, de faibles latences et sans engagement.

➤ **Amazon SQS (Simple Queue Service)**

Il représente une file d'attente, offerte comme service, hautement évolutive, où l'utilisateur peut stocker des messages qui pourront, par la suite, être lus et traités par un utilisateur (personne ou application). En d'autres termes cela permet le déplacement des données entre les composants distribués des applications qui effectuent différentes tâches.

### **Synthèse**

Une comparaison entre les fournisseurs Cloud Computing les plus utilisés (Google Apps, Google App Engine, Force.com, EC2 et S3) est représentée par le Tableau 4.3.

	<b>Amazon Web Services</b>		<b>Force.com</b>	<b>Google App Engine</b> PaaS	<b>Google Apps</b> SaaS
<b>Service</b>	<b>EC2 (IaaS)</b>	<b>S3 (PaaS)</b>	Applications pour les RH, applications de stock, applications iPhone, iPad, Android et BlackBerry	Générer des applications Web conçues pour un trafic élevé sans avoir à gérer l'infrastructure correspondante	- Gmail - Google Documents - Google Groupes - Google Agenda - Google Sites - Google Vidéos - Google Talk
		<b>Grande Instance</b> RAM 15 GO Stockage 850 GO Plateforme 64 GO	Très grande Instance RAM 15 GO Stockage 1690 GO Plateforme 64 GO	Stoçlage illimité de données	Gratuit si pas de dépassement de pourcentage prédéfini (stockage < 1 GO, taille du code d'une application : 150 MO)
<b>Coût</b>	<b>Petite Instance (par défaut)</b> RAM 1,7 GO Stockage 160 GO Plateforme 32 GO	Windows 0,12\$ par heure Linux 0,085\$ par heure	Édition gratuite Gratuit		
	<b>Grande Instance</b> RAM 15 GO Stockage 850 GO Plateforme 64 GO	Windows 0,48\$ par heure Linux 0,34\$ par heure	Édition d'entreprise 75,0438\$ par mois		
<b>Fournisseur</b>	Amazon		Salesforce.com	Google	Google
<b>Sécurité</b>	Chaque fichier stocké sera identifié par une clé		- Certification ISO 27001 - Hébergement redondant et répliqués des données	Authentification	Authentification
	Java, Python		Langage Apex (Supporte C#, .NET, Java, C++)	Java, Python	supporte Java, Python

Tableau 4.3. Solution IaaS propriétaire de Cloud [Mahjoub 2011].

### 4.8. Conclusion

De l'informatique pratique utilisée dans les années 60, puis à l'emploi de la bureautique dans les années 80, ainsi que l'apparition d'Internet et des progrès de la virtualisation, le Cloud Computing et les chiffres le prouvent, a un bel avenir devant lui. [Prunie 2011].

---

Dans cette partie, nous avons donné un aperçu général sur le Cloud Computing, par la suite nous avons effectué une étude comparative entre les diverses solutions Cloud ce qui nous a permis de sélectionner et de choisir les Clouds appropriés avec lesquels nous créons notre architecture Cloud.

Dans le prochain chapitre, nous présenterons nos contributions décisionnelles dédiées à l'évaluation des projets urbains et nous détaillerons l'architecture Cloud suggérée dans laquelle nous intégrons notre système d'aide à la décision.

# Chapitre 5

## Contributions décisionnelles pour l'évaluation des projets urbains

*« Ce chapitre est dédié à nos contributions qui consistent en l'élaboration d'un modèle décisionnel pour résoudre la problématique de l'évaluation des projets urbains dans le cadre de la planification urbaine. »*

---

### Sommaire

5.1. Introduction.....	97
5.2. Contribution 1: un système d'aide à la décision basé Cloud pour l'évaluation des projets urbains ...	97
5.2.1. Approche proposée [Benatia et al. 2015].....	97
5.2.1.1. Modèle décisionnel [Benatia et al. 2015].....	98
5.2.1.2. Intégration du modèle décisionnel dans un Cloud privé .....	101
5.2.2. Synthèse .....	108
5.3. Contribution 2 : Système de négociation décisionnelle pour l'évaluation des projets urbains .....	108
5.3.1. Présentation du protocole Contract Net .....	108
5.3.2. Adaptation du protocole Contract Net pour le problème de la planification urbaine [Benatia et al. a 2016].....	110
5.4. Conclusion.....	112

## 5.1. Introduction

La planification territoriale urbaine peut être considérée comme une série de concepts, procédures et outils qui permettent aux décideurs l'identification des actions à mettre en œuvre afin de favoriser un développement durable. La planification urbaine constitue un champ d'étude riche pour l'analyse des conflits et la conception des outils adaptés pour leur gestion. Nous constatons que le domaine de la planification urbaine est de nature multidimensionnelle, interdisciplinaire et mal définie. En effet nous observons une grande diversité de contextes et de types d'acteurs, cependant les évolutions actuelles des modalités de la décision justifient un besoin accru de développement d'outils permettant la gestion des conflits qui se multiplient au nombre d'acteurs. La résolution de ce problème consiste alors à trouver une solution commune qui permet de satisfaire tous ces acteurs.

Dans ce contexte, les systèmes décisionnels permettent la détermination d'une multitude de caractéristiques (critères) qui sont souvent contradictoires (conflictuels), et n'ont pas toujours la même importance concernant diverses personnes et organisations qui ont le plus souvent des préférences et des buts différents ce qui signifie que leurs opinions sont également différentes et doivent être prises en considération pour la décision finale.

Notre objectif, par la présente étude est de présenter une approche originale pour le développement et la conception des outils décisionnels permettant de traiter la problématique de l'évaluation des projets urbains dans le cadre de la planification urbaine. Cette problématique consiste à choisir et sélectionner le meilleur projet urbain parmi un ensemble de projets.

Ce chapitre est consacré à la présentation de nos contributions pour traiter la problématique de la planification urbaine. Dans ce contexte nous avons proposé une première contribution qui est un système décisionnel basé Cloud et une deuxième contribution qui est un protocole de négociation pour l'évaluation des projets urbains.

## 5.2. Contribution 1 : un système d'aide à la décision basé Cloud pour l'évaluation des projets urbains

Dans cette section nous allons présenter notre première contribution à savoir la proposition d'un système décisionnel basé sur l'analyse multicritère et la négociation ainsi que l'architecture Cloud dans lequel elle est intégrée.

### 5.2.1. Approche proposée [Benatia et al. 2015]

L'originalité de notre approche tient à l'utilisation simultanée d'un système d'aide à la décision et du Cloud Computing. La littérature offre peu d'exemples de couplage de ces deux types de représentation de la réalité. Notre approche consiste à intégrer un système décisionnel dédié à la résolution d'un problème de planification urbaine dans un Cloud privé. Notre modèle décisionnel est composé de deux phases : la première consiste à effectuer une analyse multicritère pour élaborer les rangements des projets urbains et la deuxième consiste à exécuter un processus de négociation afin de déterminer le choix final qui est le meilleur projet urbain. Notre modèle décisionnel est représenté par la Figure 5.1.

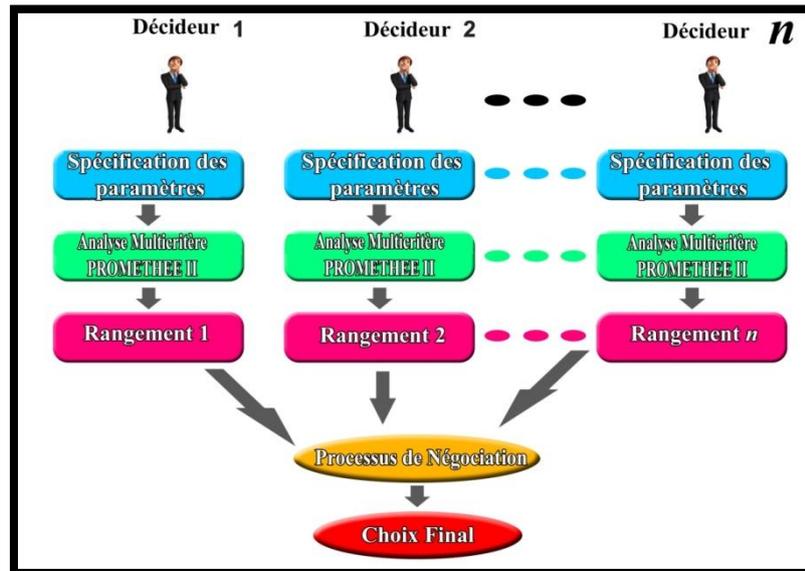


Figure 5.1. Modèle décisionnel proposé [Benatia et al. 2015].

Chaque décideur spécifie ses paramètres subjectifs cela veut dire que chaque décideur précise les poids des critères des projets urbains selon ses préférences, ses intérêts et ses connaissances. Une fois que les poids sont fixés, ils vont être envoyés avec toutes les informations des projets urbains sous forme d'un tableau de performance vers une analyse multicritère ; cette dernière sert à ranger les projets urbains du meilleur vers le moins bon, vu que nous avons  $n$  décideurs alors nous obtiendrons  $n$  rangements où chaque rangement correspond à un décideur. Par la suite tous ces rangements vont passer vers le processus de négociation pour déterminer le meilleur projet urbain parmi ces classements. L'analyse multicritère et le processus de négociation sont détaillés ci-après dans la section 1.

Nous constatons que plus le nombre de décideurs est grand, plus il y'aura une augmentation dans le temps de développement et de déploiement de notre système d'aide à la décision, des difficultés pour la gestion des différents décideurs et leurs choix des projets urbains, une insuffisance de l'espace mémoire, une diminution de la vitesse d'accessibilité et de communication de différents décideurs. Dans le but de pallier tous ces problèmes, nous décidons d'intégrer notre système d'aide à la décision dans un Cloud privé dont l'architecture est représentée dans la section 2.

### 5.2.1.1. Modèle décisionnel [Benatia et al. 2015]

Notre modèle décisionnel passe par deux phases essentielles : l'analyse multicritère et la négociation. Le fonctionnement de ces étapes est présenté ci-dessous.

#### • Analyse Multicritère

La première phase de notre modèle décisionnel est l'analyse multicritère. Comme dit précédemment l'analyse multicritère ou l'aide à la décision multicritère est employée dans divers domaines afin d'aider le décideur à choisir la solution optimale parmi un ensemble de solutions. Ainsi selon Vincke [Vincke 1989] l'aide multicritère à la décision vise à fournir aux décideurs un ensemble d'outils qui leurs permettent de résoudre un problème décisionnel dans lequel plusieurs points de vue conflictuels sont pris en considération. Comme présenté au chapitre 2, l'analyse d'aide à la décision multicritère s'articule autour de quatre

problématiques définies par Roy [Roy 1985]: la « problématique de choix » consiste à élaborer une procédure de sélection, la « problématique de tri » permet d'effectuer une procédure d'affectation, la « problématique de rangement » permet de classer les diverses alternatives dans un ordre décroissant (de la meilleure à la moins bonne) et la « problématique de description » décrit les alternatives ainsi que leurs conséquences.

La problématique de planification urbaine étudiée est considérée comme une problématique de rangement où il existe plusieurs projets urbains à caractéristiques conflictuelles qui doivent être classés par un ensemble de décideurs du meilleur vers le moins bon. Pour résoudre ce genre de problème, il existe différentes méthodes de rangement, pour notre cas nous avons pris la méthode PROMETHEE II comme méthode de rangement, vu que cette méthode donne un rangement total des alternatives, cela veut dire que cette méthode range les différentes alternatives dans un ordre décroissant. Toutes ces alternatives sont comparables, donc, on peut les distinguer contrairement aux autres méthodes de rangement. Cette méthode est basée sur l'utilisation des relations de sur-classement entre deux alternatives de la meilleure à la moins bonne. Dans ce cas, pour chacune des alternatives, un poids sera affecté à chaque attribut (critère), pour déterminer un repère de préférence d'une alternative sur l'autre. Ce repère est employé par la suite pour le calcul de l'influence d'une alternative sur l'autre (représente la domination et la soumission d'une alternative par rapport à toutes les autres). L'ensemble des étapes de cette méthode sont représentées ci-dessous :

**Phase 1:** La première étape calcule pour chaque paire de projets et pour chaque critère, la valeur du degré de préférence. Soit  $g_j(pk)$  la valeur d'un critère  $j$  pour un projet  $pk$ . On note  $d_j(pi, pk)$ , la différence de valeur du critère  $j$  pour deux projets  $pi$  et  $pk$ .

$$d_j(pi, pk) = g_j(pi) - g_j(pk) \quad (1)$$

$P_j(pi, pk)$  est la valeur de degré de préférence du critère  $j$  pour deux projets  $pi$  et  $pk$ . Les fonctions de préférence utilisées pour calculer ces degrés de préférences sont définies comme suit:

$$\begin{cases} 0 : \text{if } d_j(pi, pk) \leq 0 \\ 1 : \text{if } d_j(pi, pk) > 0 \end{cases} \quad (2)$$

**Phase 2:** La deuxième étape consiste à agréger pour chaque paire de projets les degrés de tous les critères de préférence. Pour chaque paire de projets, nous calculons un indice global de préférence. Soit  $C$  l'ensemble des critères pris en compte et  $w_j$  le poids associé au critère  $j$  (spécifié par le décideur). L'indice de préférence globale pour chaque paire de projet  $pi$  et  $pk$  est calculé comme suit:

$$\pi(pi, pk) = \sum_{j \in C} w_j \times P_j(pi, pk) \quad (3)$$

**Phase 3:** La troisième étape, qui est la première qui concerne le classement des projets, elle consiste à calculer les flux de sur-classement. Pour chaque projet  $pi$ , nous calculons le flux de

sur-classement positif  $\phi^+(pi)$  et le flux de sur-classement négatif  $\phi^-(pi)$ . Soient  $A$  l'ensemble des projets et  $n$  le nombre de projets. Le flux de sur-classement positif d'un projet  $pi$  est calculé par la formule suivante:

$$\phi^+(pi) = \frac{1}{n-1} \sum_{x \in A} \pi(pi, x) \quad (4)$$

Le flux de sur-classement négatif d'un projet  $pi$  est calculé par la formule suivante:

$$\phi^-(pi) = \frac{1}{n-1} \sum_{x \in A} \pi(x, pi) \quad (5)$$

**Phase 4:** La dernière étape consiste à utiliser les flux de sur-classement afin d'établir un classement complet entre les projets. Le classement est basé sur les flux de sur-classement. Celles-ci sont calculées pour chaque projet à partir des flux de sur-classement positif  $\phi^+(pi)$  et négatif  $\phi^-(pi)$ . Le flux de sur-classement net d'un projet est calculé comme suit:

$$\phi(pi) = \phi^+(pi) - \phi^-(pi) \quad (6)$$

Dans le contexte de notre étude, il existe plusieurs décideurs où chaque décideur exprime son point de vue personnel en spécifiant les paramètres subjectifs (c'est à dire fixer le poids des critères pour chaque projet urbain), pour cela nous avons utilisé la méthode PROMETHEE II pour chaque décideur; on va donc obtenir plusieurs rangements où chaque rangement ou classement correspond à un décideur comme le montre la Figure 5.1; dans ce cas il y aura un problème de conflit entre les décideurs pour choisir le meilleur projet parmi un ensemble de rangements. Pour résoudre ce problème, nous avons proposé d'utiliser un processus de négociation.

#### • Processus de négociation

La négociation représente une démarche à travers laquelle plusieurs acteurs arrivent à une solution commune. Dans le but de résoudre le problème du conflit entre les différents décideurs, nous utilisons le système de vote pour élire un projet urbain parmi un ensemble de rangement des projets urbains effectués par la méthode PROMETHEE II de l'analyse multicritère. Il existe différentes méthodes de vote qui ont été présentée dans le chapitre 3 ; pour notre système d'aide à la décision nous avons employé la méthode Hare. Cette dernière est basée sur la détermination du meilleur choix en retirant consécutivement les actions les moins souhaitées. Les étapes de cette méthode sont les suivantes :

- Dans le cas où, un projet urbain est classé premier selon au moins la moitié des séries de préférence (représentent les listes des projets), il existe un meilleur projet urbain et la méthode est achevée.
- Dans le cas contraire où il n'y a pas de projet qui est classé premier selon au moins la moitié des séries, on choisit alors le projet classé premier sur le moins de séries et on le supprime de

toutes les séries. Si une multitude de projets sont à égalité, on les supprime des séries. Les projets qui suivent celui qui a été enlevé sur les séries remportent donc une place.

- On recommence la méthode pour chercher si un projet est classé premier selon au moins la moitié des séries et on supprime le projet le moins souhaité dans le cas contraire.

- La méthode cesse lorsqu'un projet se présente en premier selon au moins la moitié des séries ou bien dans le cas où tous les projets restants se présentent en premier sur précisément le même nombre de séries.

### 5.2.1.2. Intégration du modèle décisionnel dans un Cloud privé

Dans cette section, nous allons présenter l'architecture du Cloud privé que nous avons suggérée afin d'intégrer notre système d'aide à la décision (DSS) proposé pour l'évaluation des projets urbains. Pour notre architecture nous avons employé deux types de Cloud, un Cloud de type infrastructure qui est le Cloud OpenStack et un Cloud de type plateforme qui est le CloudBess. Avant d'introduire notre architecture suggérée nous allons présenter ses deux types de Cloud.

#### 5.2.1.2.1. Présentation d'OpenStack

Le Cloud OpenStack est un projet libre qui permet à toute structure la création et l'offre des services Cloud. Il est développé en grande partie par la société Rackspace qui s'est associée à la NASA pour développer un outil libre et standardisé sous licence Apache 2.0 (licence de logiciel libre et open source).

La NASA a été un utilisateur du projet open source d'Eucalyptus avant les problèmes d'évolutivité dans leur projet Nebula ce qui signifiait qu'ils avaient besoin de développer leurs propres technologies dans cet espace. Une contribution de leurs fichiers Cloud par Rackspace, combinée avec le logiciel de calcul Nebula de la NASA a mené à la naissance initiale d'OpenStack. En même temps, depuis sa création, l'OpenStack consortium a réussi à ramener plus de 100 membres, y compris les hauts noms de l'industrie tels que Citrix profil, Canonical et Dell.

Puisque l'AWS d'Amazon était le premier service principal qui a utilisé le Cloud, OpenStack rend également ses services disponibles par l'intermédiaire des API EC2 et S3 compatibles d'Amazon. Cela garantit que tous les outils existants qui fonctionnent avec les offres du Cloud d'Amazon, peuvent travailler avec des déploiements d'OpenStack.

##### 5.2.1.2.1.1. Les composants d'OpenStack

Le projet OpenStack est une combinaison de trois éléments [Sehgal 2012] principaux:

#### 1. OpenStack Compute (Nova)

Il est utilisé pour orchestrer, gérer et offrir des machines virtuelles sur plusieurs hyperviseurs, y compris QEMU et KVM. Ceci est analogue à Amazon Elastic Compute Cloud (EC2).

Nova reprend le rôle de fournisseur de services informatiques dans le Cloud OpenStack. En tant que telle, n'importe quelle activité a dû soutenir le cycle de vie d'une instance de la machine virtuelle dans le Cloud est manipulée par Nova. Ceci inclut la gestion du stockage de bloc, le réseau, l'ordonnancement, les ressources informatiques, l'autorisation et les hyperviseurs.

Cependant, Nova ne fournit pas des fonctionnalités de virtualisation par lui-même. Il est conçu pour utiliser l'API libvirt afin d'interagir avec les hyperviseurs pris en charge.

Cela signifie que Nova est un hyperviseur agnostique, il fournit un support pour « Xen », « KVM », « UML », « XenServer / XCP », « VMware vSphere » et « Hyper - V ».

Tous les services offerts par le Nova sont accessibles à travers une API qui est compatible avec AWS EC2.

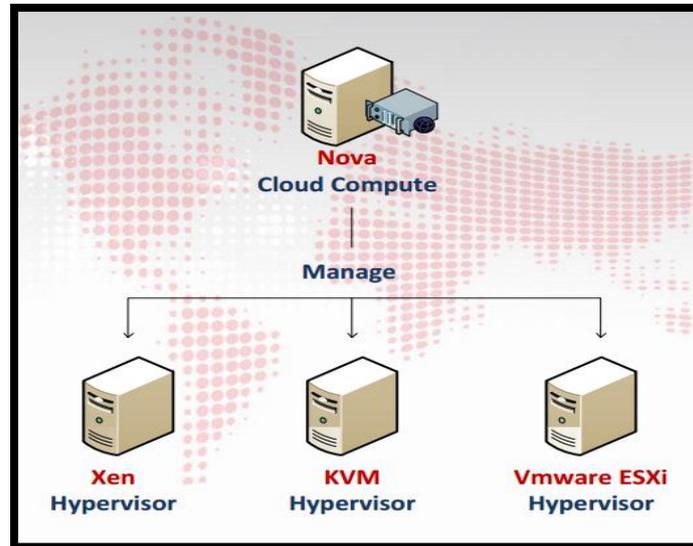


Figure 5.2. Nova [Sehgal 2012].

Les principales composantes de la Nouvelle-nova-api [2] sont : « nova-api », « rabbitmq-server », « nova-compute », « nova-network », « nova-volume » et « nova-scheduler ».

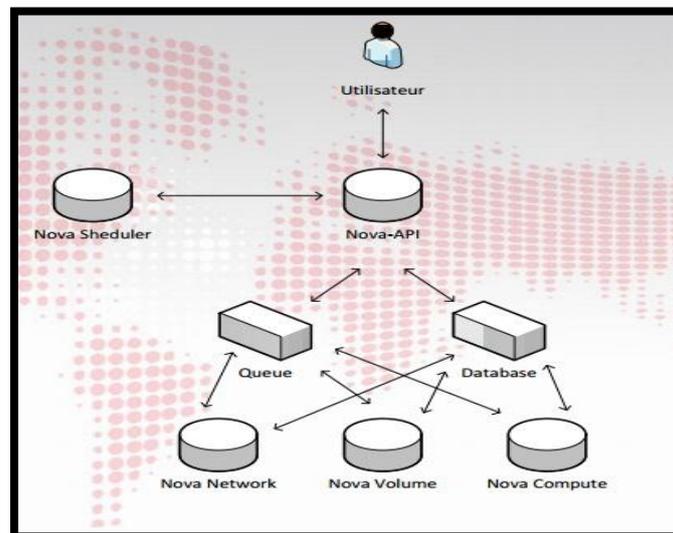


Figure 5.3. Architecture logique de Nova [Sehgal 2012].

**API Services (nova-api) :** le service Nova API fournit une interface avec le monde extérieur pour interagir avec l'infrastructure Cloud. Le serveur API est le seul élément que le monde extérieur utilise pour gérer l'infrastructure. Le serveur API alors, à son tour, communique avec les composantes pertinentes de l'infrastructure Cloud en utilisant le message Queue.

**Message Queuing (rabbitmq-server)** : le contrôleur de Cloud OpenStack communique avec les composants Nova tel que le contrôleur, l'ordonnanceur, le contrôleur réseau, et le contrôleur de volume à l'aide du protocole « AMQP » (Advanced Message Queue). Le Cloud OpenStack utilise le « rabbitmq-server » à cet effet. Nova utilise des appels asynchrones de la demande- réponse, avec un call- back qui se déclenche une fois qu'une réponse est reçue. Depuis que la communication asynchrone est utilisée, aucune des actions de l'utilisateur ne reste bloquée longtemps dans un état d'attente. Cela est particulièrement vrai puisque de nombreuses actions prévues par les appels d'API tels que le lancement d'une instance ou le téléchargement d'une image sont longues.

**Computing Services (nova-compute)** : les serveurs fournissant des services de calcul par l'intermédiaire du « nova-compute », traitent le cycle de vie de gestion d'instance. Ils reçoivent des demandes de gestion de cycle de vie par l'intermédiaire des messages Queue et effectuent des opérations appropriées. Il existe plusieurs serveurs fournissant des services informatiques dans un déploiement typique de nuage de production. Une instance est déployée sur n'importe quel travailleur de calcul disponible basé sur l'algorithme d'ordonnement utilisé par Nova.

**Network Services (nova-api)** : les services de réseau fournis par nova-api sont traités avec la configuration du réseau des machines hôtes. Ils effectuent des opérations telles que l'allocation des adresses IP, la configuration VLAN pour les projets, la mise en place des groupes de sécurité et la configuration de réseaux pour les nœuds de calcul.

**Block Storage Services (nova-volume)** : les services de stockage de bloc assurés par le nova-volume incluent la création, la suppression, l'attachement d'un volume à une instance, et le détachement d'un volume d'une instance. Les volumes permettent de fournir le stockage persistant à l'usage des instances, car le disque principal attaché à une instance est non-persistant et toutes les modifications qui lui sont apportées sont perdues quand le volume est isolé ou l'instance est terminée. Quand un volume est isolé d'une instance ou quand une instance, à laquelle le volume est attaché, est terminée, il maintient les données stockées au cours d'un attachement à une instance antérieure. Ces données sont accessibles par le rattachement du volume à la même instance ou l'attachement à d'autres instances. En tant que tel, n'importe quelle donnée précieuse accumulée pendant le cycle de vie de l'instance doit être écrite dans un volume, de sorte qu'elle soit accessible par la suite.

**Scheduling Services (nova-scheduler)** : le « nova-scheduler » trace des appels API pour les composants d'OpenStack appropriés. Il choisit un serveur à partir d'un réservoir en fonction de l'algorithme d'ordonnement en place. Un ordonnanceur peut fonder ses décisions sur divers facteurs tels que la charge, la mémoire, la distance physique de la zone de disponibilité, CPU architecture, etc.

## 2. OpenStack Image Service (Glance)

Le Glance est un outil qui récupère et recherche les images de la machine virtuelle. Les informations concernant les images enregistrées sont stockées dans une base de données SQL, qui peut être soit « MySQL », « PostgreSQL », « SQLite » ou de nombreuses autres variétés.

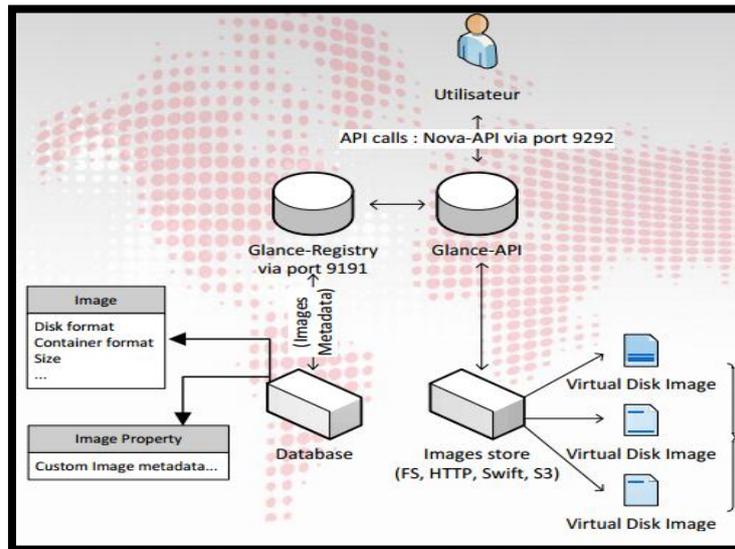


Figure 5.4. Architecture globale de Glance [Sehgal 2012].

### 3. OpenStack Object Store(Swift)

Il fournit le stockage redondant pour les objets statiques. Ce service est extensible à des données de tailles massives et théoriquement peut fournir un stockage infini. Il est analogue à Amazon Simple Storage Service (S3).

#### 5.2.1.2.1.2. Architecture d'OpenStack

Un aperçu de l'architecture d'OpenStack peut être vu sur la Figure 5.5. Le client interagit avec le serveur Nova API. Dans le cas où des demandes relatives à l'enregistrement ou l'interrogation d'images sont envoyées, l'API transmet ces demandes à l'API Glance, qui peut effectuer des requêtes dans la base de registres Glance (stockées dans une base de données SQL).

Toutefois, si la demande porte sur la gestion d'une instance, alors celle-ci est transmise à la file d'attente du serveur, qui va à son tour distribuer les demandes de composants appropriés.

Les demandes d'allocation d'adresses réseau, d'association et de désallocation sont traitées par Nova network. D'autre part, la création de stockage de bloc, la suppression et les demandes d'association sont traitées par le Nova Volume. De même, Les requêtes liées à l'instance de machine virtuelle sont traitées par Nova Compute.

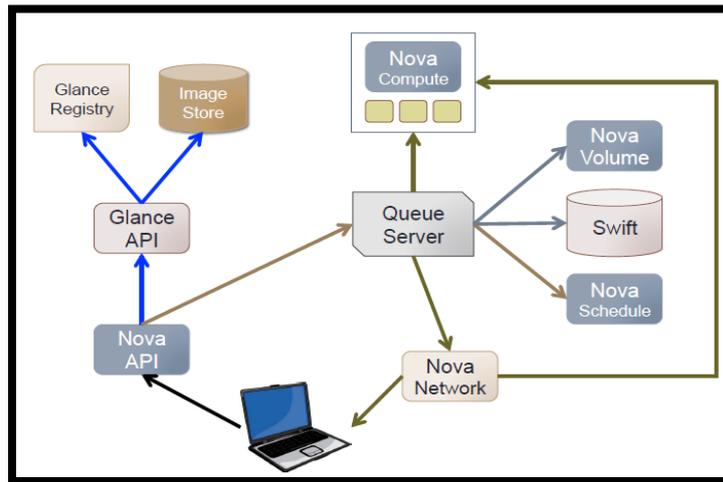


Figure 5.5. Une vue d'ensemble de l'architecture OpenStack [Sehgal 2012].

Le client interagit avec les services Nova API. Ce dernier utilise Glance API pour l'enregistrement et la récupération d'images. Toutes les autres demandes sont envoyées à la file d'attente du serveur, qui les fait passer aux nœuds de calcul, de volume, de programme ou de réseau selon le besoin.

#### 5.2.1.2.2. Présentation de CloudBees

CloudBees propose une plate-forme PaaS pour environnements Java qui englobe la totalité du cycle de vie des applications, de leur développement à leur déploiement et à leur utilisation. De l'extérieur, la solution CloudBees se présente comme ceci (Figure 5.6) :

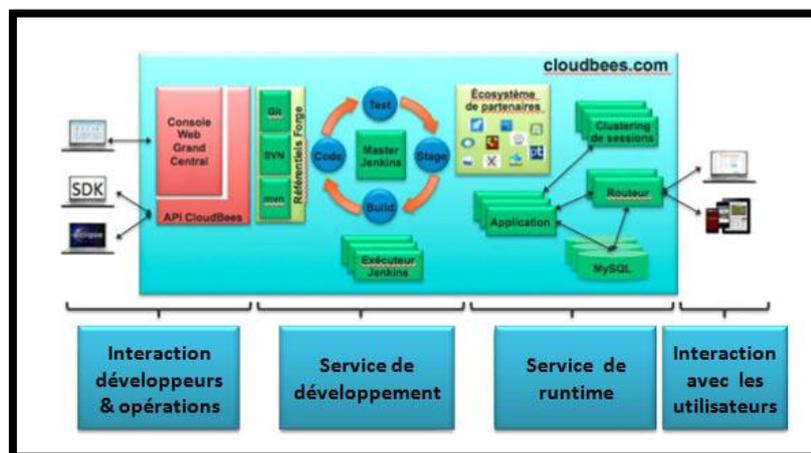


Figure 5.6. Le PaaS CloudBees vu de l'extérieur [3].

Les développeurs ou responsables de l'exploitation d'applications de production, disposent d'une vue unifiée par l'intermédiaire de la console Web CloudBees « GrandCentral ». Cette console est une simple interface utilisateur qui s'ajoute à la riche interface API de CloudBees, à laquelle les développeurs accèdent également par l'intermédiaire du kit de développement logiciel « SDK » ou le module additionnel Eclipse. À l'attention des développeurs, CloudBees héberge une panoplie de services de développement, parmi lesquels le serveur d'intégration « continue Jenkins (Continuous Integration Server) », qui est utilisé pour réaliser les « *builds* »,

tester et promouvoir ou déployer l'application. Le service d'intégration continue « CI » de CloudBees dispose d'une fonction unique qui utilise les ressources de Cloud sur demande pour accélérer la réalisation de nos « builds » et gérer à un très faible coût le caractère irrégulier qui caractérise l'intégration continue.

CloudBees propose également un jeu de services de « runtime » qui permettent de déployer une application sur la machine virtuelle Java « JVM » de notre choix, ainsi qu'un service de base de données MySQL hébergé. Le service de « clustering de sessions » facilite la gestion de l'état des sessions Java entre instances d'applications, garantit l'intégrité des transactions en cas de panne, et assure une scalabilité horizontale « scale-out » totalement transparente. On peut accéder instantanément à l'application que nous avons déployée aux fins de tests ou d'évaluation interne, ou mapper nos propres noms de domaines et adresses (*URL*), ce qui permet aux utilisateurs d'interagir avec l'application hébergée conformément à nos attentes, ainsi qu'aux leurs.

Les services « Development » et « Runtime » sont tous les deux intégrés dans une large gamme de services hébergés par des partenaires. Nous disposons ainsi d'un accès intégré aux meilleurs outils de surveillance, de journalisation et de test.

### 5.2.1.2.3. Notre architecture Cloud Proposée

L'architecture Cloud proposée est illustrée par la Figure 5.7. Nous avons choisi d'intégrer notre DSS (Système d'aide à la décision) dans un Cloud privé parce que le modèle décisionnel n'est pas destiné au grand public; il est utilisé par un groupe de décideurs qui sont les décideurs de la ville, de plus en déployant notre système d'aide à la décision dans un Cloud privé, les données seront plus protégées et plus sécurisées. Notre architecture Cloud est fondée sur trois couches essentielles: une couche infrastructure, une couche plateforme et une couche utilisateur.

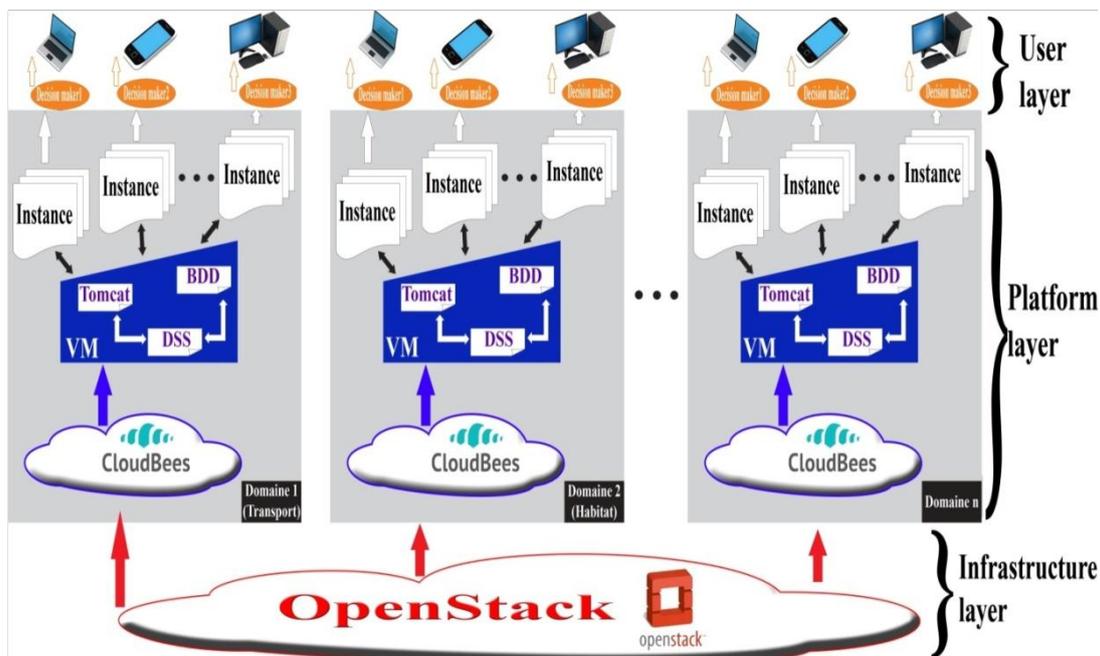


Figure 5.7. Système décisionnel basé Cloud pour l'évaluation des projets urbains [Benatia et al. 2015].

## A. Couche Infrastructure

Cette couche permet de mettre à disposition des ressources virtuelles en utilisant un matériel standard à savoir : des unités dotées d'une grande puissance de calcul, de traitement et de stockage.

Dans le cadre de notre étude, la ville est composée de plusieurs secteurs tels que le transport, l'habitat, le commerce; l'économie... etc. Afin d'aider l'ensemble des décideurs à évaluer les projets urbains proposés dans les différents secteurs nous avons mis à leur disposition un système décisionnel. Ce dernier fonctionne de la même façon pour tous les secteurs, la seule différence ce sont les projets urbains et l'ensemble des décideurs. Pour la mise en place de notre système d'aide à la décision il faut avoir un serveur pour chaque secteur de la ville. Pour cela nous avons utilisé le Cloud OpenStack qui permet de fournir une base de l'infrastructure virtualisée, distribuée et automatisée. La fonction de base d'OpenStack est la création de machines virtuelles donc nous utilisons ce type de Cloud pour créer un serveur virtuel pour chaque secteur de la ville.

## B. Couche Plateforme

La plateforme de développement repose sur l'infrastructure. C'est la couche centrale de notre architecture. Elle utilise toutes les ressources fournies par la couche Infrastructure telles que : la RAM virtuelle, le Disque dur virtuel, le réseau, le CPU virtuel...

Cette couche permet d'offrir tous les éléments et les outils nécessaires afin de construire, déployer et assurer le cycle de vie de n'importe quelle application.

A ce niveau nous avons déployé un Cloud de type PaaS qui est le CloudBees au sein de chaque serveur virtuel ou machine virtuelle créée par le Cloud OpenStack. Le CloudBees nous permet de fabriquer une plateforme de développement personnalisée et de réaliser des tâches liées au déploiement et la gestion de cycle de vie d'une application dans un Cloud ; donc nous employons ce Cloud pour la création de machine virtuelle logicielle (VM) dans laquelle on déploie notre application qui est composée des trois services : Tomcat, BDD (Base de Données) et DSS (Système d'aide à la décision). Pour chaque connexion d'un décideur qui veut participer au processus d'évaluation des projets urbains, le CloudBess lui affecte une instance composée des trois services de l'application.

-Service Tomcat : il représente l'interface web qui va faire l'interaction entre les décideurs et le système d'aide à la décision.

-Service Base de Données (BDD) : elle permet aux chefs des projets urbains de stocker tous les projets urbains proposés avec leurs caractéristiques pour résoudre un problème dans un domaine de la ville dans une base de données SQL. Le système d'aide à la décision communique avec cette BDD afin de collecter les informations relatives aux projets urbains.

-Service Système d'aide à la décision (DSS) : c'est le système d'aide à la décision que nous avons présenté précédemment dans la section A de ce chapitre. Le DSS interagit avec la base de données pour collecter toutes les informations concernant les projets urbains afin d'effectuer une aide à la décision multicritère qui range les projets urbains proposés. Un processus de négociation sera lancé par la suite pour négocier la solution parmi l'ensemble des rangements

fournis par l'analyse multicritère et de déterminer le meilleur projet urbain à l'ensemble des décideurs via l'interface Tomcat.

### **C. Couche utilisateur**

La couche utilisateur permet aux utilisateurs finaux (les décideurs) d'accéder à notre système d'aide à la décision pour la participation au processus de choix et d'évaluation des projets urbains via un navigateur web. Donc chaque décideur peut accéder à la demande pour au processus de décision via n'importe quel appareil (ordinateur, Smartphone, tablette,...) doté d'un navigateur web et une connexion Internet.

#### **5.2.2. Synthèse**

La méthode d'aide à la décision multicritère et celle de la négociation employées dans notre processus décisionnel sont des méthodes automatiques cela veut dire que le décideur ne participe pas réellement au processus décisionnel et il ne donne pas son choix concernant les projets urbains, de ce fait les résultats (décisions) peuvent ne pas le convaincre et le satisfaire, alors il y aura un conflit entre les différents décideurs. Cependant nous proposerons dans la section suivante une deuxième contribution qui prend en considération l'opinion de chaque décideur sur les projets urbains. Dans cette contribution-là nous proposerons un protocole de négociation inspiré du protocole le plus célèbre dans la négociation à savoir le protocole Contract Net où l'avis et le choix des différents décideurs sont inclus.

### **5.3. Contribution 2 : Système de négociation décisionnelle pour l'évaluation des projets urbains**

Dans cette section nous allons présenter notre deuxième contribution qui est la proposition d'un nouveau protocole de négociation dédié à l'évaluation des projets urbains et basé sur le principe du protocole Contract Net. Avant d'introduire notre protocole suggéré nous allons présenter les concepts de ce protocole (Contract Net).

#### **5.3.1. Présentation du protocole Contract Net**

L'algorithme représenté par la Figure 5.8 montre les sept phases détaillées du protocole Contract Net. Ces phases regroupent l'ensemble des traitements effectués par le « gestionnaire » et les « contractants », ainsi que les différentes communications entre eux.

- Etant donné une tâche, un gestionnaire, un groupe de  $(n-1)$  soumissionnaires :
1. le gestionnaire envoie un message de type « annonce- tâche » à un groupe d'agents (ou fait un 'broadcast').
  2. Chaque agent évalue la tâche annoncée à l'aide d'une fonction locale 'évalue- annonce'.
  3. L'évaluation précédente permet à certains agents de soumettre une proposition à l'aide d'une 'soumission- tâche' au gestionnaire.
  4. Si une proposition est jugée satisfaisante (l'aide de la fonction 'évalue- soumission'), alors le gestionnaire envoie un message de type 'acceptation' à celui dont la proposition est retenue. Il envoie également un message de type 'refus' aux autres agents dont les propositions n'ont pas été retenues.
  5. Le gestionnaire peut mettre fin à la période d'acceptation de proposition si le temps d'expiration est dépassé.
  6. Si aucune proposition n'a été retenue, alors le gestionnaire fait parvenir à tous les agents non retenus un message de type 'refus' pour indiquer le rejet de chacune des propositions.
  7. Il peut alors se retirer de la négociation, retenir la proposition la plus acceptable redémarrer un nouvel appel d'offre (nouveau 'annonce- tâche') ou prolonger le temps d'expiration de la période d'acceptation de proposition.
  8. L'agent ayant obtenu le contrat, remet un rapport d'exécution lorsque la tâche est complétée

Figure 5.8. Algorithme du protocole Contract Net [Sahli 2010].

Les étapes du protocole Contract Net (CNP) sont résumées par le diagramme de séquence suivant :

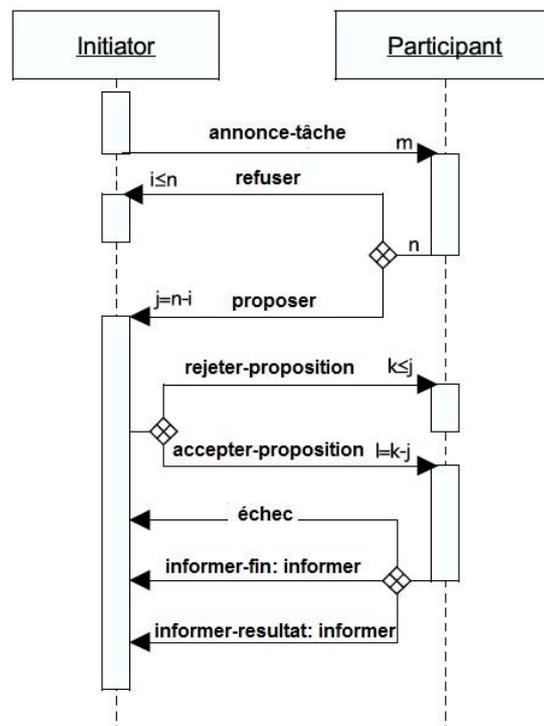


Figure 5.9. Le diagramme de séquence du protocole Contract Net [Alam et al. 2005].

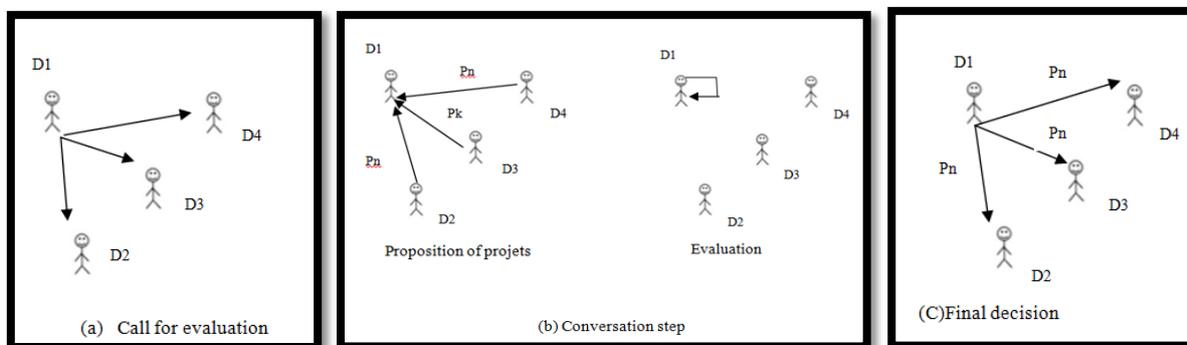
### 5.3.2. Adaptation du protocole Contract Net pour le problème de la planification urbaine [Benatia et al. a 2016]

Basé sur le principe et les étapes du protocole Contract Net (CNP), un protocole de négociation est proposé pour aider les décideurs à évaluer et choisir le meilleur projet urbain. Le protocole de négociation proposé comprend deux types de participants: le décideur principal qui est le gestionnaire du processus décisionnel et les décideurs participants qui sont les décideurs intervenants, ils communiquent et discutent directement avec le décideur principal. Notre protocole proposé est composé de trois phases essentielles : la phase d'authentification, la phase de conversation et la phase de décision finale.

- La phase d'authentification: cette étape est la première phase de notre protocole proposé, elle introduit la négociation. Dans cette phase, le décideur principal annonce à tous les décideurs participants le début du processus d'évaluation des projets urbains. Chaque participant confirme ou refuse sa participation au processus décisionnel.

- La phase de conversation: c'est la phase principale de notre protocole, dans cette étape une conversation se déroule entre le décideur principal et les décideurs participants où ces derniers proposent des projets urbains selon leur choix au décideur principal. Le décideur principal évalue les propositions des participants, soit il met fin au processus de prise de décision, ou il demande de refaire un nouveau choix.

-La Phase finale de décision: dans cette phase, le décideur principal notifie tous les décideurs participants par la fin de la négociation avec le meilleur projet urbain élu.



Pour le décideur principal, les types de messages sont les suivants:

- **start\_evaluation** : ce message indique le début du processus de prise de décision.
- **ask\_proposition** : ce message indique aux décideurs participants d'envoyer leurs propositions (projet urbain) au décideur principal.
- **ask\_vote (list of projects)** : ce type de message est utilisé uniquement quand il ya une égalité entre les votes des projets urbains. Ce message est envoyé par le décideur principal afin de demander aux participants de choisir un projet parmi une liste de projets en égalité.
- **end\_success (project)** : ce message est envoyé par le décideur principal pour indiquer aux participants la fin de l'évaluation ainsi que le meilleur projet élu.

Pour les décideurs participants, les types de messages sont les suivants:

- **confirm** : ce message indique au décideur principal la confirmation de la participation au processus de l'évaluation des projets urbains.
- **refuse** : ce message indique au décideur principal le refus de la participation au processus de l'évaluation des projets urbains.
- **propose (project)** : ce message est utilisé par chaque décideur participant pour envoyer sa proposition (projet urbain) au décideur principal.
- **vote\_established (project)** : ce message répond à la demande de vote du décideur principal. A travers ce message, le décideur participant indique au décideur principal, le choix du projet urbain réalisé.

L'algorithme ci-après présente les étapes de notre protocole de négociation proposé. Ces phases regroupent l'ensemble des traitements effectués par le décideur principal et les décideurs participants ainsi que les différentes communications entre eux.

1. Le décideur principal envoie à tous les décideurs participants un message de type "start\_evaluation".
2. Chaque décideur participant recevant ce type de message de la part du décideur principal répond par le message "confirme" ou "refuse".
3. Le décideur principal demande à tous les participants d'envoyer leurs propositions (projet urbain) par le biais du message "ask\_proposition".
4. Chaque décideur participant envoie sa proposition (projet urbain selon son choix) au décideur principal en utilisant le message "proposition (project)".
5. Le décideur principal évalue toutes les propositions reçues. Il calcule le nombre de votes pour chaque projet urbain et détermine le projet majoritaire s'il existe et envoie le message "end\_success (project)". Dans le cas d'égalité dans le vote, les projets élus vont être proposés aux décideurs qui ont voté pour les projets non élus afin de refaire un vote restreint par l'envoi du message "ask\_vote (list of projets in equality)".
6. Dans ce cas les décideurs participants recevant ce message vont choisir un projet parmi la liste des projets proposés et envoyer au décideur principal un message "vote\_established (selected project)".
7. Le décideur principal recevant le vote de chaque participant évalue et détermine le projet majoritaire et envoie un message "end\_succs (project)" à tous les décideurs participants. Sinon, dans le cas d'égalité entre les projets élus, le décideur effectue lui-même un vote et envoie aux décideurs participants un message "end\_succes (project)".

Figure 5.11. Algorithme de notre protocole de négociation [Benatia et al. a 2016].

La modélisation de notre protocole de négociation par le diagramme de séquence est représentée par la Figure 5.12.

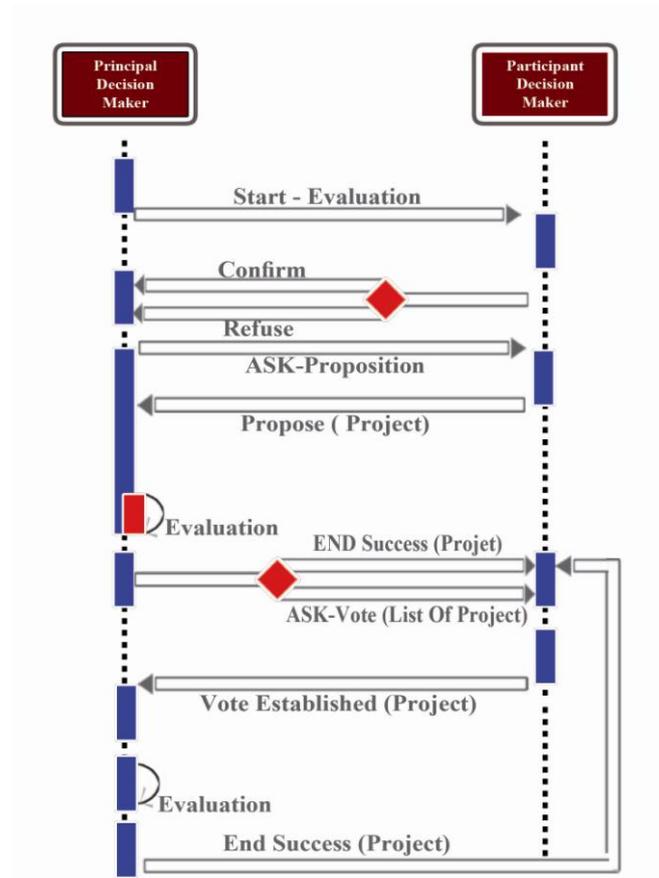


Figure 5.12. Diagramme de séquence pour notre protocole de négociation [Benatia et al. a 2016].

## 5.4. Conclusion

Au terme de ce chapitre, nous avons présenté deux contributions pour résoudre les problématiques relatives à la planification urbaine.

Notre première contribution consiste à proposer une approche qui permet de répondre à la problématique d'évaluation des projets urbains. Cette approche est basée sur l'intégration d'un système décisionnel à la technologie Cloud Computing, afin de profiter de ses avantages. Le modèle décisionnel proposé permet : de déterminer les poids des critères de projets urbains par chaque décideur, d'effectuer une analyse multicritère pour chaque décideur en utilisant la méthode PROMETHEE II qui range les projets urbains du meilleur au moins bon et enfin d'exécuter un processus de négociation en utilisant la méthode Hare qui permet d'élire entre plusieurs rangements effectués par la méthode PROMETHEE II et de fournir la décision finale à l'ensemble des décideurs.

Notre deuxième contribution consiste à suggérer un système de négociation décisionnel toujours pour résoudre le problème de la planification urbaine. Nous avons proposé un nouveau protocole de négociation inspiré du protocole de négociation le plus célèbre qui est le protocole Contract Net (CNP). Notre protocole est composé de trois étapes essentielles : l'étape d'authentification où le décideur principal annonce le début du processus négociation, l'étape de conversation dans laquelle une conversation se déroule entre les décideurs et l'étape de fin de

décision où le décideur principal annonce la fin du processus de négociation et notifie les décideurs par la décision finale.

Le chapitre suivant sera consacré à une étude de cas réel accompagnée d'une discussion des résultats obtenus.

# Chapitre 6

## Implémentation et mise en œuvre

*« Le but de ce chapitre est d'expérimenter nos modèles décisionnels suggérés et de présenter une étude de cas accompagnée d'une discussion des résultats. »*

---

### Sommaire

6.1. Introduction.....	115
6.2. Etude de cas.....	115
6.3. Validation du système d'aide à la décision basé Cloud .....	117
6.3.1. Application de la méthode PROMETHEE II [Benatia et al. 2015].....	117
6.3.2. Application de la méthode Hare [Benatia et al. 2015] .....	121
6.3.3. Intégration de notre DSS dans un Cloud privé.....	123
6.3.3.1. Installation d'openstack (quick start).....	123
6.3.3.2. Installation de Cloudbees (quick start).....	124
6.3.3.3. Déploiement du DSS dans le Cloud [Benatia et al. b 2016].....	125
6.4. Validation du système décisionnel basé sur la négociation [Benatia et al. a 2016].....	129
6.5. Conclusion.....	134

## 6.1. Introduction

Selon Gambino [Gambino 2010] la planification du territoire désigne l'action publique s'efforçant de redistribuer la population ainsi que les différentes activités économiques sur le plan géographique, dans le but soit pour rendre homogène le territoire, ou bien pour activer ou contrôler le développement, soit pour l'amélioration des situations du pays dans l'exercice des compétitions sur le plan international. La planification urbaine représente une problématique qui implique plusieurs décideurs souhaitant faire face à des problèmes avec le maximum d'éléments objectifs et prenant leurs décisions en tenant compte des intérêts de toutes les parties concernées.

L'objectif de notre étude est de concevoir des modèles décisionnels pour la planification urbaine et plus précisément la problématique d'évaluation et de sélection des projets urbains.

Afin de valider nos modèles décisionnels suggérés, une étude de cas a été réalisée. Les différents résultats obtenus seront détaillés et discutés dans ce chapitre.

## 6.2. Etude de cas

Dans l'objectif de la validation de nos contributions décisionnelles, la ville test sélectionnée afin d'effectuer notre étude de cas est la ville d'Annaba. Cette dernière est composée d'un ensemble de sites. Nous avons choisi cette région à cause de la pluralité des projets de planification dans celle-ci et plus particulièrement dans le secteur de l'habitat. Donc, afin de réaliser notre étude de cas, nous avons contacté un bureau d'étude qui est le GART (General ART & Technique) dans la Wilaya d'Annaba qui nous a donné toutes les informations sur les projets d'habitat proposés et leurs caractéristiques qui sont représentés ci-dessous :

1/ Projet 2000 Logements Publics Locatifs : le nouveau projet de 2000 logements sera situé dans la commune Draa Erich de la ville d'Annaba, ses caractéristiques sont les suivantes :

- Nombre d'appartements : 2000.
- Surface appartement : 72 m<sup>2</sup>.
- Montant : 4 721 426 750.00 DA.
- Délais : 27 mois.
- Répartis en : 58 blocs.

2/ Projet 300 Logements Publics Promotionnels : le nouveau projet 300 logements sera localisé dans la commune de Draa Erich de la Wilaya d'Annaba, ses caractéristiques sont les suivantes :

- Nombre d'appartements : 300.
- Surface appartement : 60 m<sup>2</sup>.
- Montant : 1 424 214 610.00 DA.
- Délais : 10 mois.
- Répartis en : 13 blocs.

3/ Projet 954 Logements Publics Locatifs : le nouveau projet 954 logements sera situé dans la commune Kalitoussa daïra de Berrahal de la ville d'Annaba, ses caractéristiques sont les suivantes :

- Nombre d'appartements : 954.

- Surface appartement : 67 m<sup>2</sup>.
- Montant : 2 322 273 600.00 DA.
- Délais : 24 mois.
- Répartis en : 72 blocs.

4/ Projet 500 Logements Publics Promotionnels : le nouveau projet 500 logements sera localisé dans la commune d'El Bouni de la ville d'Annaba, ses caractéristiques sont les suivantes :

- Nombre d'appartements : 500.
- Surface appartement : 70 m<sup>2</sup>.
- Montant : 9481350000.00 DA.
- Délais : 18 mois.
- Répartis en : 28 blocs.

5/ Projet 260 Logements Publics Promotionnels : ce nouveau projet 260 logements sera situé dans la commune Berrahal de la Wilaya d'Annaba, ses caractéristiques sont les suivantes :

- Nombre d'appartements : 260.
- Surface appartement : 80 m<sup>2</sup>.
- Montant : 4471313380.00 DA.
- Délais : 12 mois.
- Répartis en : 16 blocs.

En s'inspirant du concept du développement durable, nous classons les caractéristiques des projets d'habitat en trois catégories (sociale, économique et écologique) afin d'aider les décideurs à évaluer les projets urbains.

- Caractéristiques sociales: C1 (nombre de blocs), C2 (Nombre d'appartements), C3 (Surface d'appartement).
- Caractéristiques économiques: C4 (délais), C5 (Montant).
- Caractéristiques écologiques: C6 (effet écologique : prend les valeurs 1, 2, 3).

Les caractéristiques des projets d'habitat collectés sont regroupées et présentées par le Tableau 6.1.

Liste des Projets	Caractéristiques des projets					
	Caractéristiques Sociales			Caractéristiques économiques		Caractéristiques écologiques
	C1	C2	C3	C4	C5	C6
Projet1	58	2000	72	27	4721426750	1
Projet2	13	300	60	10	1424214610	3
Projet3	72	954	67	24	2322273600	2
Projet4	28	500	70	18	948135000	1
Projet5	16	260	80	12	447131380	2

Tableau 6.1. Caractéristiques des projets d'habitat [Benatia et al. 2015] [Benatia et al. a 2016].

### 6.3. Validation du système d'aide à la décision basé Cloud

Au cours de cette section, nous allons présenter les étapes d'application des deux méthodes de l'analyse multicritère et négociation, de validation de notre système décisionnel, d'installation des Clouds et d'intégration de notre système d'aide à la décision dans le Cloud.

#### 6.3.1. Application de la méthode PROMETHEE II [Benatia et al. 2015]

Dans un premier temps, le décideur spécifie les poids des critères des projets d'habitat selon ses préférences, Nous avons choisi une « échelle de préférence » commune. Le choix en conservant une « échelle de préférence » commune va faciliter considérablement l'attribution des poids aux différents attributs (critères). La Figure ci-dessous montre l'affectation des poids des critères dans le cas d'un seul décideur.

	<input checked="" type="checkbox"/>					
<b>Décideur 1</b>	Nb blocs	Nb logts	superficie ap...	delai travaux	Montant	ecologie
Unité	nombre	nombre	m <sup>2</sup>	mois	DA	unit
Cluster/Groupe	◆	◆	◆	◆	◆	◆
<b>Préférences</b>						
Min/Max	max	max	max	min	min	max
Poids	3,00	3,00	2,00	4,00	5,00	1,00
Fn. de préférence	Usuel	Usuel	Usuel	Usuel	Usuel	Usuel
Seuils	absolu	absolu	absolu	absolu	absolu	absolu
- Q: Indifférence	n/d	n/d	n/d	n/d	n/d	n/d
- P: Préférence	n/d	n/d	n/d	n/d	n/d	n/d
- S: Gaussien	n/d	n/d	n/d	n/d	n/d	n/d
<b>Statistiques</b>						
<b>Evaluations</b>						
<input checked="" type="checkbox"/> Projet 1	58,00	2000,00	72,00	27,00	4721426752,0	1,00
<input checked="" type="checkbox"/> Projet 2	13,00	300,00	60,00	10,00	1424214610,0	3,00
<input checked="" type="checkbox"/> Projet 3	72,00	954,00	67,00	24,00	2322273600,0	2,00
<input checked="" type="checkbox"/> Projet 4	28,00	500,00	70,00	18,00	9481350000,0	1,00
<input checked="" type="checkbox"/> projet 5	16,00	260,00	80,00	12,00	45176830,00	2,00

Figure 6.1. Affectation des poids.

Les résultats du calcul des flux de sur-classement positifs et négatifs pour chaque projet sont représentés la Figure 6.2.

Rang	action	Phi	Phi+	Phi-
1	Project1	0,4677	0,7097	0,2419
2	Project 3	0,3065	0,6290	0,3226
3	Project 5	-0,1613	0,3952	0,5565
4	Project 4	-0,2258	0,3629	0,5887
5	Project 2	-0,3871	0,3065	0,6935

Figure 6.2. Calcul des flux pour chaque projet [Benatia et al. 2015].

Le rangement des projets repose sur le score des PHI. Alors, à ce stade, nous classons les projets par ordre décroissant en utilisant les scores PHI. Le classement des projets du meilleur au moins bon selon un seul décideur est représenté par la Figure 6.3.

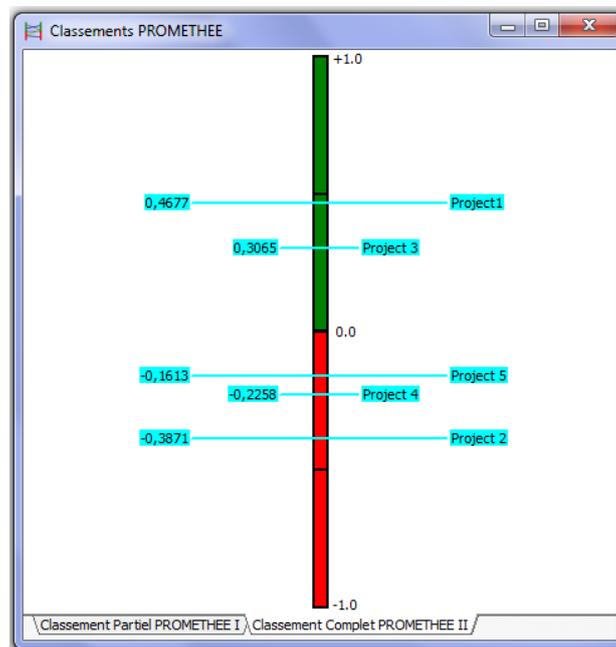


Figure 6.3. Rangement des projets [Benatia et al. 2015].

On suppose qu'on a cinq décideurs, chacun d'entre eux doit spécifier les poids des critères des projets d'habitat selon ses préférences et ses connaissances comme le montre les figures suivantes :

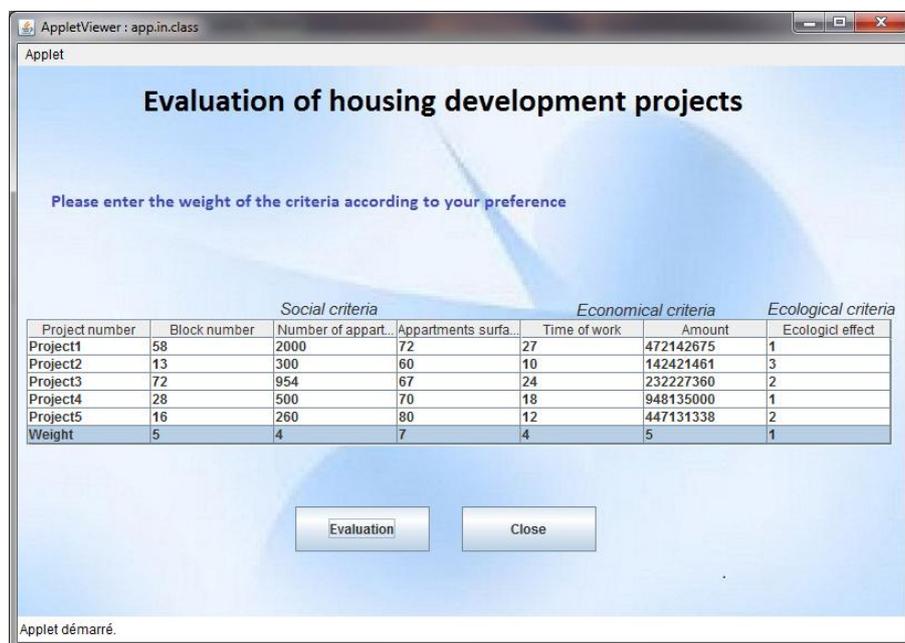


Figure 6.4. Spécification des poids des critères par le décideur 1 [Benatia et al. 2015].

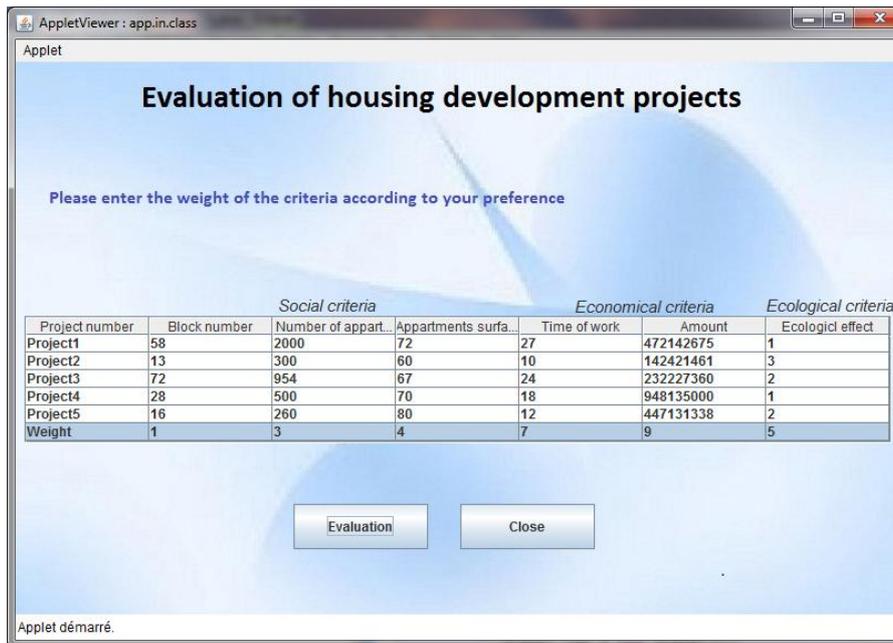


Figure 6.5. Spécification des poids des critères par le décideur 2.

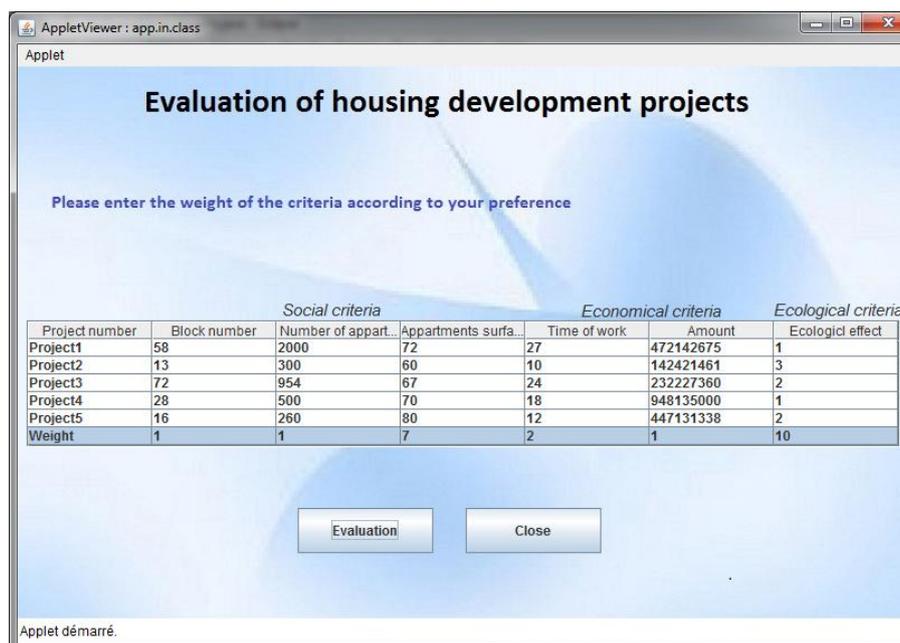


Figure 6.6. Spécification des poids des critères par le décideur 3.

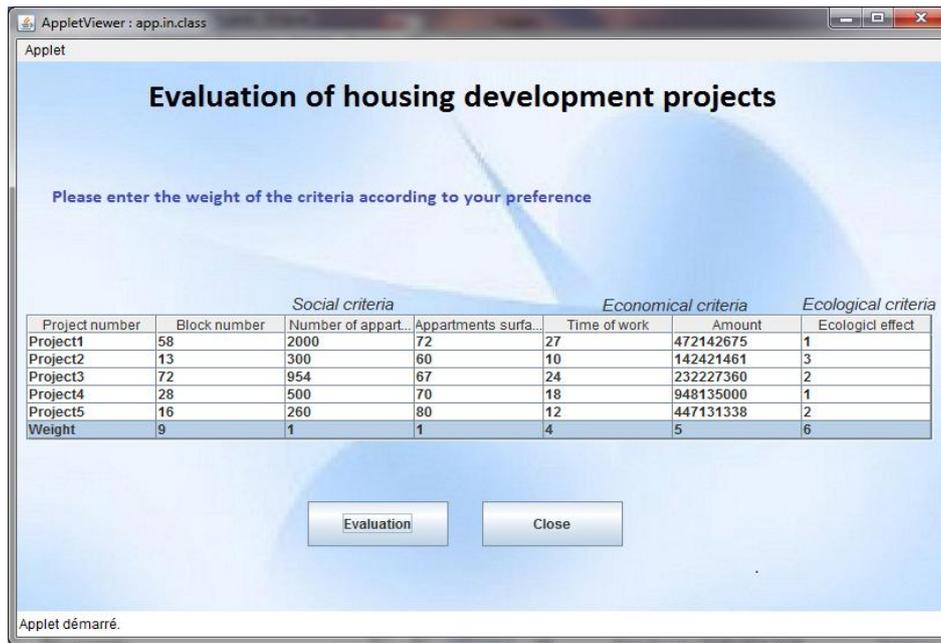


Figure 6.7. Spécification des poids des critères par le décideur 4.

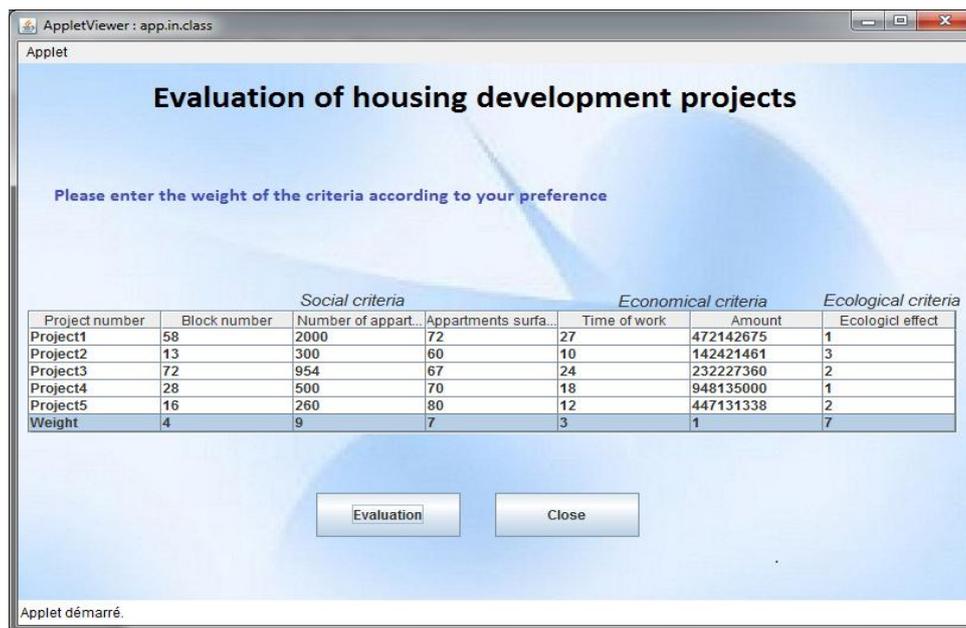


Figure 6.8. Spécification des poids des critères par le décideur 5.

Une fois que tous les décideurs ont saisi et validé les poids des caractéristiques, la procédure de détermination du meilleur projet d'habitat sera déclenchée. Dans un premier temps, la méthode PROMETHEE II sera appliquée afin de donner les différents rangements des décideurs comme le montre la Figure 6.9.

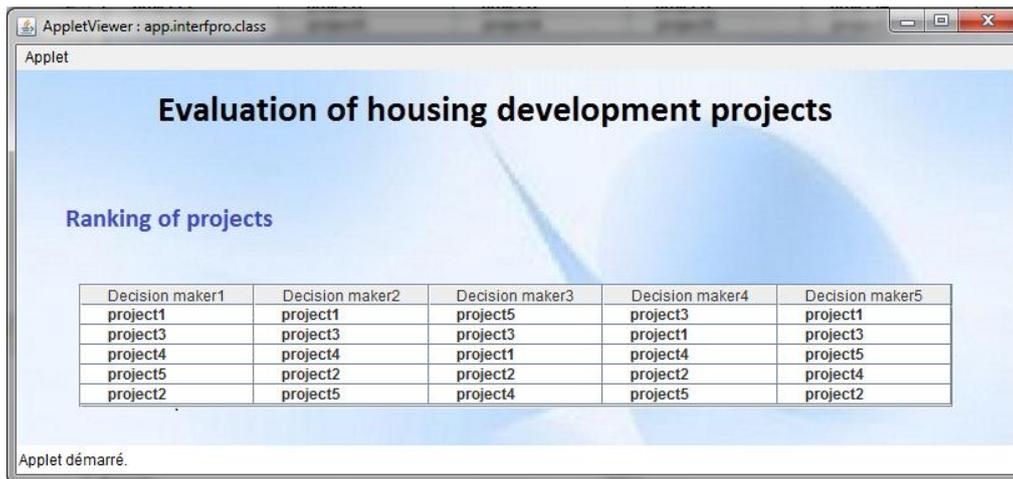


Figure 6.9. Rangements des projets d'habitat en utilisant PROMETHEE II [Benatia et al. 2015].

Une fois tous les rangements des projets effectués, la méthode Hare du processus de négociation sera déclenchée pour déterminer le meilleur projet d'habitat. Le déroulement de la méthode Hare sera représenté ci-après.

### 6.3.2. Application de la méthode Hare [Benatia et al. 2015]

La méthode Hare s'applique lorsque les différents rangements des projets sont effectués après l'exécution de la méthode PROMETHEE II. On suppose qu'on a cinq décideurs, alors après l'application de la méthode PROMETHEE II nous obtiendrons cinq rangements où chaque rangement correspond à un décideur comme le représente le tableau 6.2.

Décideur 1 (Rangement 1)	Décideur 2 (Rangement 2)	Décideur 3 (Rangement 3)	Décideur 4 (Rangement 4)	Décideur 5 (Rangement 5)
Project1	Project1	Project5	Project3	Project1
Project3	Project3	Project3	Project1	Project3
Project4	Project4	Project1	Project4	Project5
Project5	Project2	Project2	Project2	Project4
Project2	Project5	Project4	Project5	Project2

Tableau 6.2. Rangements des projets selon chaque décideur [Benatia et al. 2015].

La méthode Hare utilise ces résultats pour déterminer le meilleur projet d'habitat. Le système Hare est basé sur l'idée d'arriver à un choix social par les suppressions successives des alternatives moins souhaitables. Plus précisément, la procédure est la suivante. Nous commençons par la suppression du projet(s) qui se produit au-dessus des séries (listes) les plus faibles. Dans notre cas, il est nécessaire qu'un projet soit classé premier selon au moins les trois séries pour qu'il soit le « choix social ». Aucun projet ne satisfait cette condition (car project1 gagne uniquement 2 votes), on doit alors supprimer le projet le moins souhaité. Le projet le moins classé premier dans les séries des préférences est project2 parce qu'il n'est classé premier sur aucune des séries. On supprime project2 de toutes les séries et nous obtiendrons alors les séries suivantes :

project 1 project 1 project 5 project 3 project 1  
 project 3 project 3 project 3 project 1 project 3  
 project 4 project 4 project 1 project 4 project 5  
 project 5 project 5 project 4 project 5 project 4

On n'a pas toujours eu un projet classé premier selon au moins trois séries, il est nécessaire donc de supprimer le projet le moins souhaité à nouveau. Project4 est le seul projet qui n'apparaît pas au sommet de la série, on le supprime alors de toutes les séries de préférences et nous obtiendrons les rangements suivants:

project 1 project 1 project 5 project 3 project 1  
 project 3 project 3 project 3 project 1 project 3  
 project 5 project 5 project 1 project 5 project 5

Maintenant, project3 est le projet qui possède un seul vote ; donc il sera supprimé.

project 1 project 1 project 5 project 1 project 1  
 project 5 project 5 project 1 project 5 project 5

Project5 est le projet le moins désiré, donc il sera supprimé des séries de préférences. Donc project1 sera classé premier sur les trois séries, il représente alors le « choix social » par la procédure Hare.

project 1 project 1 project 1 project 1 project 1

La Figure 6.10 montre le déroulement du processus de négociation par la méthode Hare. Le projet 1 est élu par notre système décisionnel comme étant le meilleur projet d'habitat.

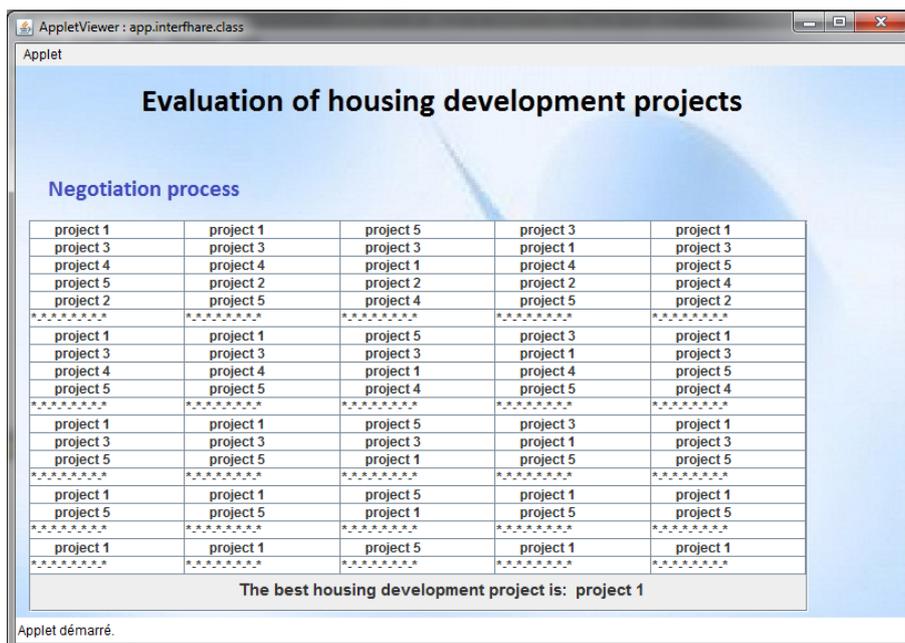


Figure 6.10. Processus de négociation en utilisant la méthode Hare [Benatia et al. 2015].

### 6.3.3. Intégration de notre DSS dans un Cloud privé

D'après notre contribution discutée dans le Chapitre 5, nous avons proposé une architecture d'un Cloud privé afin d'intégrer notre système décisionnel (DSS). Cette architecture repose sur deux types de Cloud, un Cloud IaaS qui est l'OpenStack et un Cloud PaaS qui est le CloudBees. Dans cette partie nous allons présenter les phases d'installation de ces deux types de Cloud ainsi que les étapes de leurs utilisations pour l'intégration de notre DSS.

#### 6.3.3.1. Installation d'openstack (quick start)

-La première étape de l'installation rapide d'OpenStack est de disposer d'un système d'exploitation Linux tel que : Ubuntu, Fedora, Debian...

-La deuxième étape consiste à télécharger le Devstack en utilisant la commande suivante :

```
git clone https://git.openstack.org/openstack-dev/devstack
```

Le devstack contient un script qui installe OpenStack et les modèles pour les fichiers de configuration.

-La troisième étape est la phase de la configuration, voici les valeurs qui doivent être le plus souvent réglées :

- no logging
- pre-set the passwords to prevent interactive prompts
- move network ranges away from the local network (FIXED\_RANGE and FLOATING\_RANGE, commented out below)
- set the host IP if detection is unreliable (HOST\_IP, commented out below)

```
[[local|localrc]]
ADMIN_PASSWORD=secrete
DATABASE_PASSWORD=$ADMIN_PASSWORD
RABBIT_PASSWORD=$ADMIN_PASSWORD
SERVICE_PASSWORD=$ADMIN_PASSWORD
SERVICE_TOKEN=a682f596-76f3-11e3-b3b2-e716f9080d50
#FIXED_RANGE=172.31.1.0/24
#FLOATING_RANGE=192.168.20.0/25
#HOST_IP=10.3.4.5
```

-La dernière phase est la phase du début d'installation en utilisant la commande ci-dessous :

```
cd devstack; ./stack.sh
```

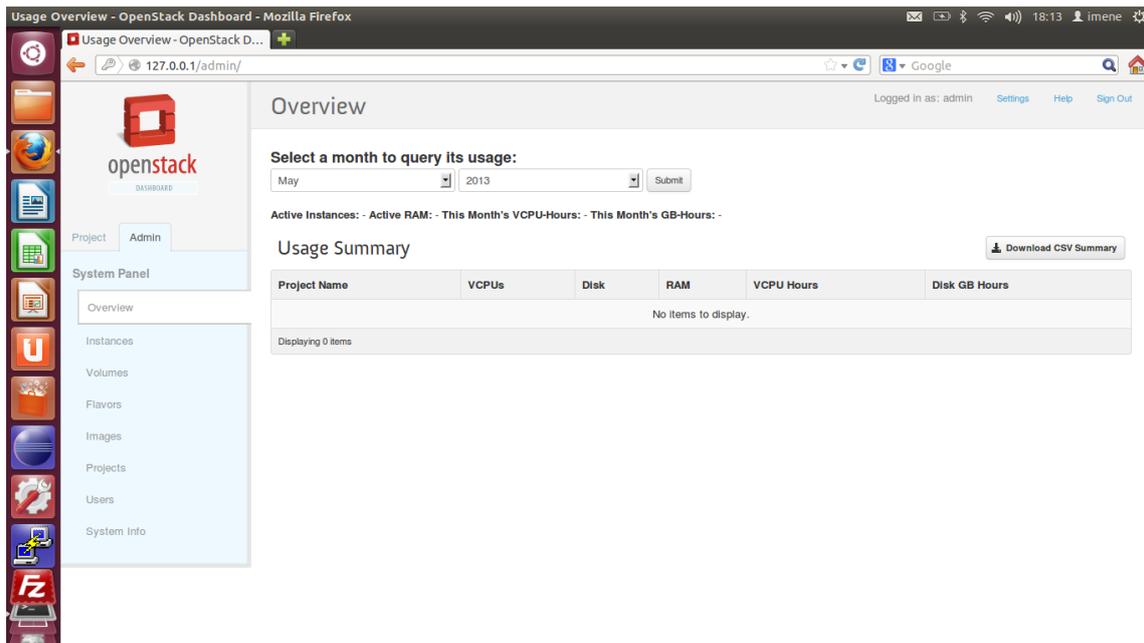


Figure 6.11. L'interface principale d'OpenStack [Benatia et al. b 2016].

### 6.3.3.2. Installation de Cloudbees (quick start)

Avant de procéder aux étapes d'installation de CloudBees, il faut que Java 6 ou une version supérieure à celle-ci soit installée, et la commande Java doit être disponible sur le PATH (on peut la vérifier en exécutant la commande : `java -version`). Les étapes de déploiement du CloudBees sont les suivantes :

- Téléchargement et configuration du SDK CloudBees : la première étape de l'installation consiste à télécharger le SDK CloudBees et le décompresser dans un répertoire sur notre machine, ce fichier doit être nommé `BEES_HOME`. Afin d'exécuter les outils de ligne de commande de CloudBees, nous devons définir la variable d'environnement `BEES_HOME`, et d'ajouter `BEES_HOME` à notre variable d'environnement `PATH`.

- Installation sur linux : la deuxième phase concerne l'installation ou le déploiement sur le système d'exploitation linux en utilisant les commandes suivantes :

```
curl -L cloudbees-downloads.s3.amazonaws.com/sdk/cloudbees-sdk-1.5.2-bin.zip > bees_sdk.zip
unzip bees_sdk.zip
rm bees_sdk.zip
cd cloudbees-sdk-1.5.2
```

- Ajout des variables Bees à notre environnement linux : on rajoute au fichier `./bash_profile` les lignes de commandes suivantes :

```
export BEES_HOME=~/.cloudbees-sdk-1.5.2
export PATH=$PATH:$BEES_HOME
```

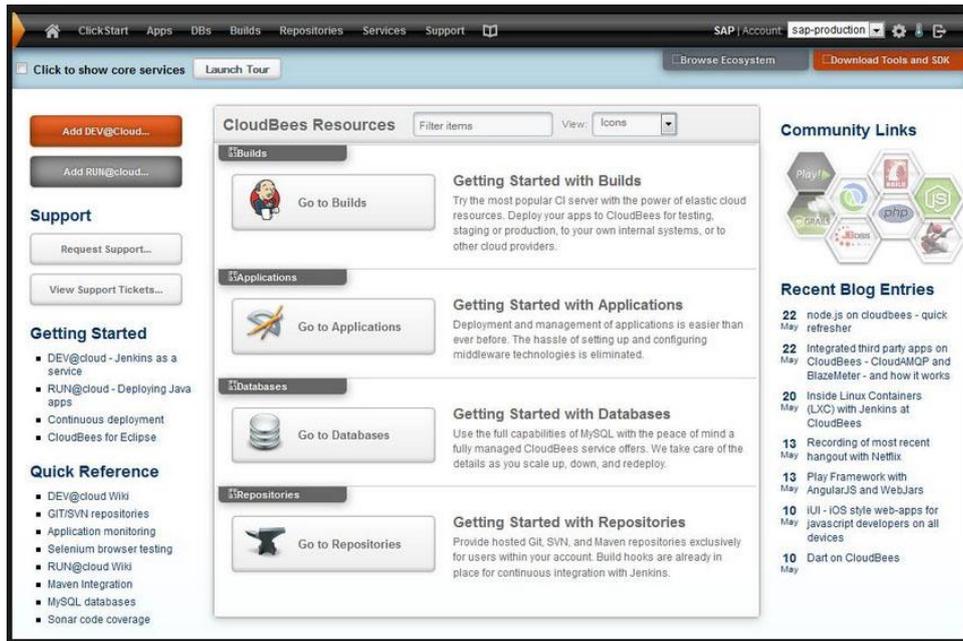


Figure 6.12. L'interface principale du CloudBees.

### 6.3.3.3. Déploiement du DSS dans le Cloud [Benatia et al. b 2016]

Nous constatons que si le nombre de décideurs augmente, le temps de développement et d'exécution augmente, l'espace mémoire sera insuffisant et la vitesse d'accessibilité et de communication entre les différents décideurs sera lente. Pour ces raisons, nous avons pris la décision d'intégrer notre système décisionnel multicritère dans un Cloud de type plateforme (PaaS) qui est CloudBees et de déployer ce dernier dans un Cloud de type infrastructure (IaaS) qui est le Cloud Openstack. En utilisant OpenStack, nous construisons une infrastructure sur laquelle repose CloudBees et nous créons des machines virtuelles pour chaque domaine (secteur) de la ville. Les étapes de la création et le lancement d'une machine virtuelle sont les suivantes:

1. Création d'images: l'image représente un système d'exploitation; c'est là que nous pouvons choisir des images spécifiques pour Openstack. On peut choisir soit l'image par défaut qui est cirros 0.3.0 x86\_64 ou bien télécharger d'autres images tel que : ubuntu, linux, windows, fedora...

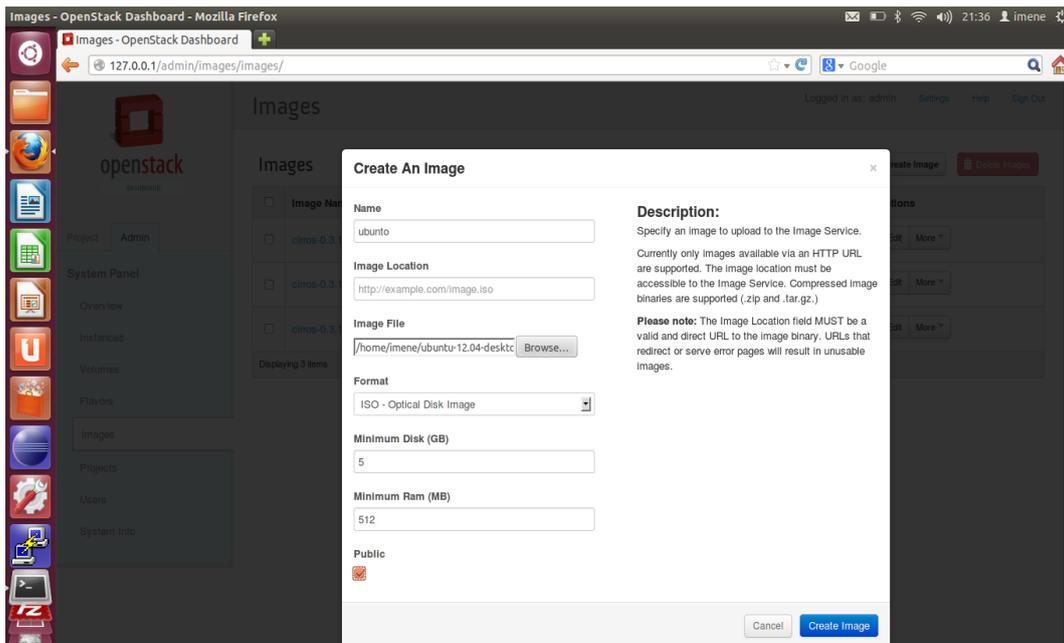


Figure 6.13. Création d'une image [Benatia et al. b 2016].

2. Création des flavors : les Flavors sont les paramètres personnalisables avec lesquels on crée chaque machine virtuelle. Les options configurables pour les Flavors sont : VCPU, RAM, disque racine et disque éphémère. Cela rend plus facile de personnaliser chaque machine virtuelle (VM) pour répondre aux besoins de performance.

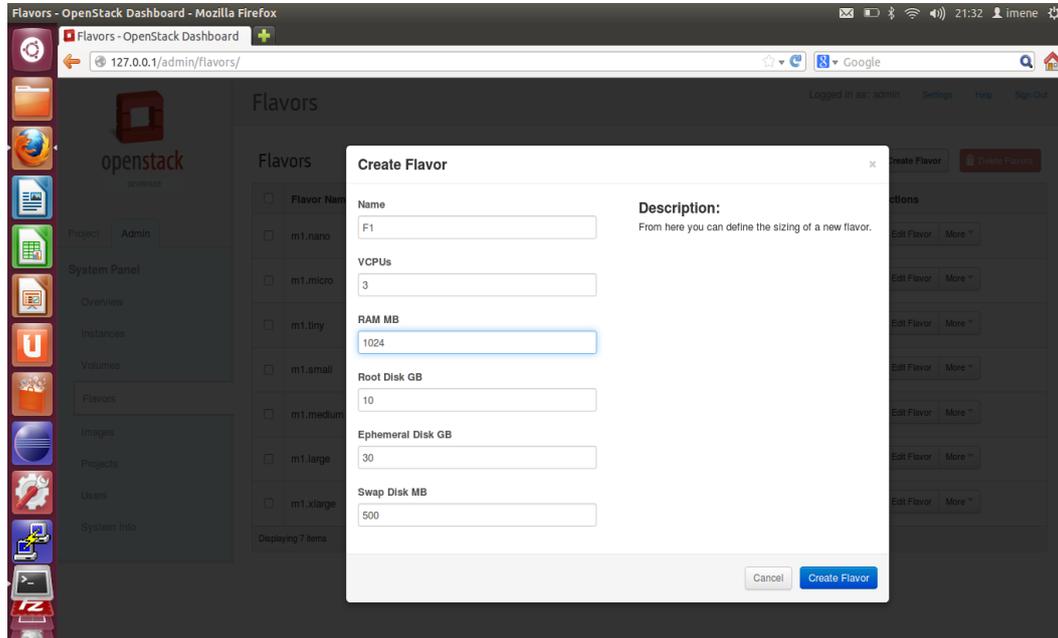


Figure 6.14. Création d'un flavor [Benatia et al. b 2016].

3. Création de volumes : les volumes OpenStack sont des dispositifs persistants qui peuvent être attachés ou détachés des virtuelles machines (VM), mais ils peuvent être attachés uniquement à une seule VM à la fois. Pour ajouter de nouveaux volumes, nous avons besoin de seulement du nom, du type et de la taille du volume.

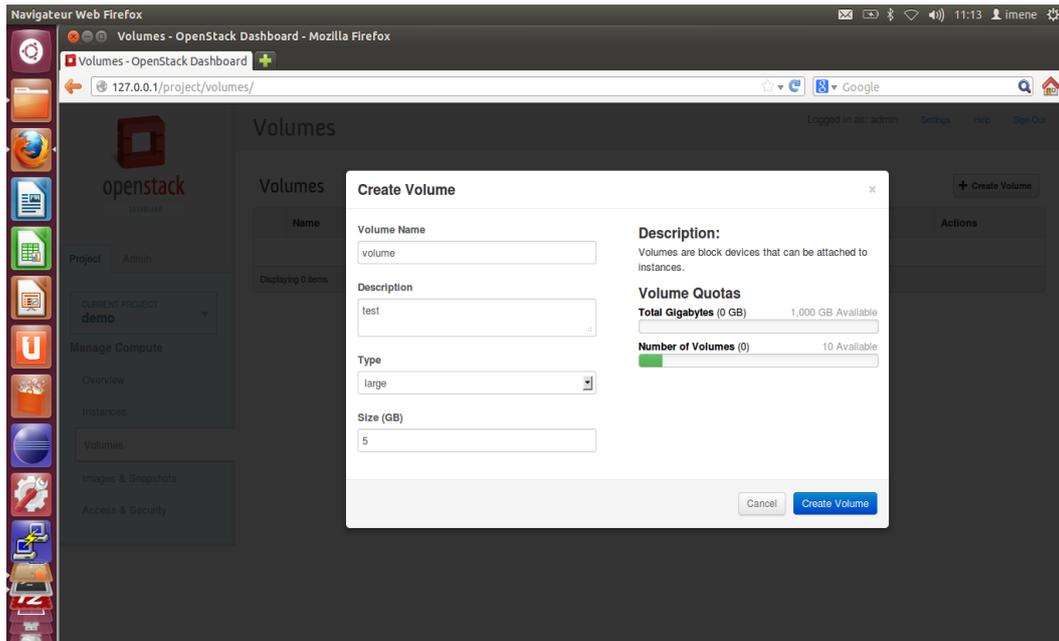


Figure 6.15. Création d'un volume [Benatia et al. b 2016].

4. Création et lancement d'une instance : les instances sont des machines virtuelles en cours d'exécution au sein du Cloud OpenStack. Pour lancer une instance, nous avons besoin de choisir un nom, une image, un flavor, le nombre d'instances (ou VM) et un volume (en option). Nous débutons une instance à partir du dashboard en cliquant sur le bouton de lancement instance sur la page Instances.

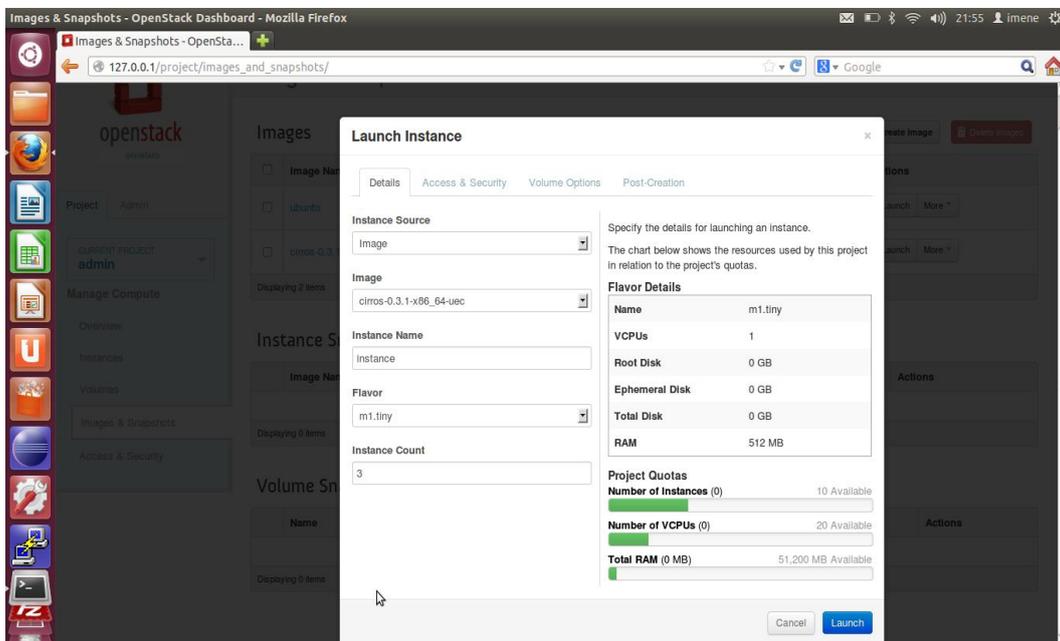


Figure 6.16. Lancement d'une machine virtuelle [Benatia et al. b 2016].

Selon notre problème, la ville est composée de plusieurs domaines ou secteurs tels que l'habitat, le commerce, le transport ... au lieu d'acheter pour chaque domaine de la ville un serveur pour déployer notre système d'aide à la décision, nous utilisons OpenStack pour créer un serveur

virtuel pour chacun d'entre eux. La création des machines virtuelles nous permettent d'économiser de l'argent consacré à l'achat de serveurs et de réduire le temps du déploiement de notre DSS pour chacun des domaines. Comme représenté dans la Figure 6.17, nous créons pour chaque domaine de la ville une machine virtuelle. La Figure 6.18 montre une machine virtuelle du domaine d'habitat lancée dans laquelle nous déployons le Cloud de type plateforme CloudBees.

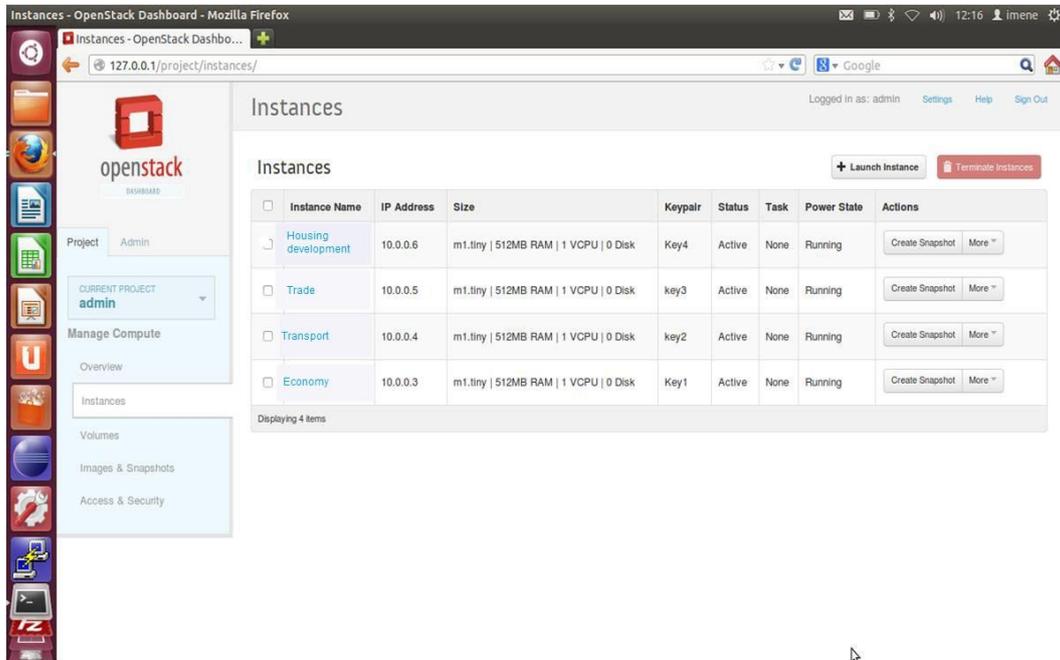


Figure 6.17. Création des machines virtuelles en utilisant OpenStack [Benatia et al. b 2016].

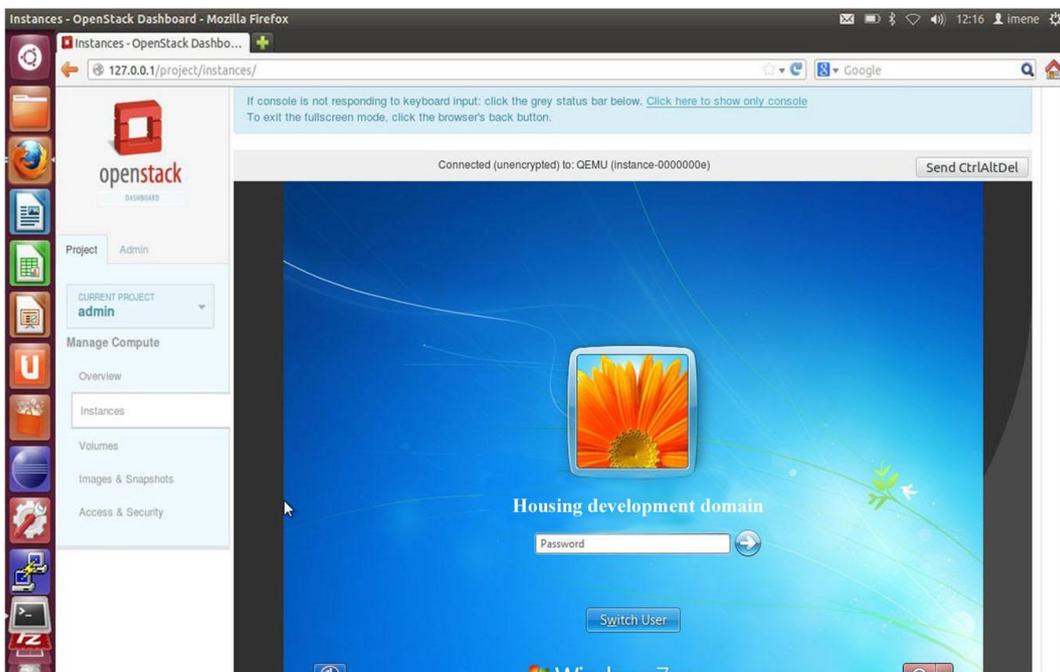


Figure 6.18. Machine virtuelle lancée.

Nous prenons CloudBees comme une plate-forme de développement pour le déploiement de notre DSS. Nous pouvons facilement déployer notre système d'aide à la décision (DSS) et bénéficier des options offertes par CloudBees selon nos besoins. Nous utilisons CloudBees pour affecter une instance à chaque décideur où chaque instance est composée de ressources matérielles et logicielles nécessaires pour le fonctionnement de notre Système d'aide à la décision (DSS). Chaque décideur participe au processus d'évaluation des projets d'habitat sans se rendre compte qu'il travaille sur une instance matérielle et logicielle (OpenStack et CloudBees).

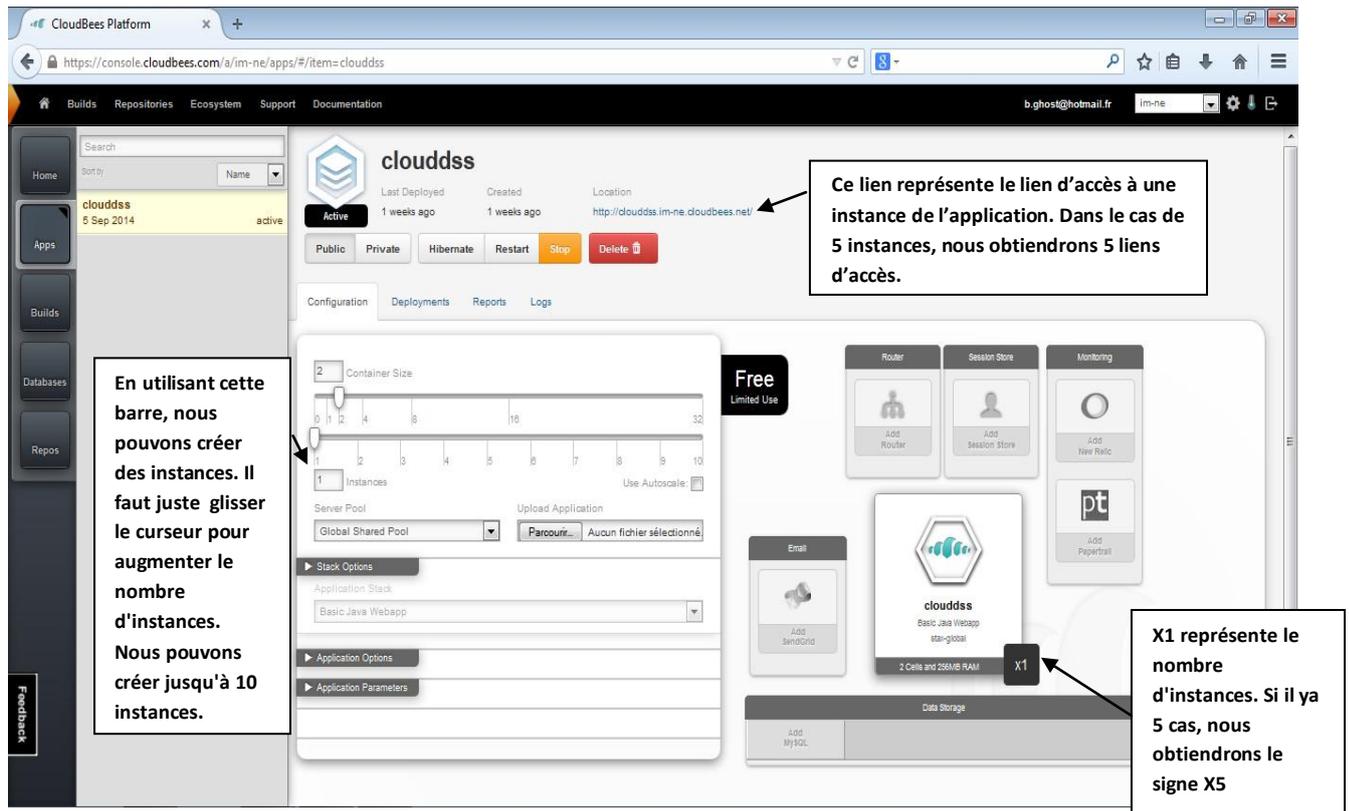


Figure 6.19. Intégration du DSS dans CloudBees [Benatia et al. b 2016].

#### 6.4. Validation du système décisionnel basé sur la négociation [Benatia et al. a 2016]

Nous avons développé et mis en place un système décisionnel basé sur notre protocole proposé qui permet d'aider à décider et à choisir le meilleur projet d'habitat. Notre système de négociation comporte trois étapes. Dans la première étape le décideur principal (initiateur) invite tous les décideurs à participer à une session d'évaluation des projets d'habitat. Nous supposons qu'il existe neuf décideurs participants où le décideur principal les invite comme représenté dans la Figure 6.20.

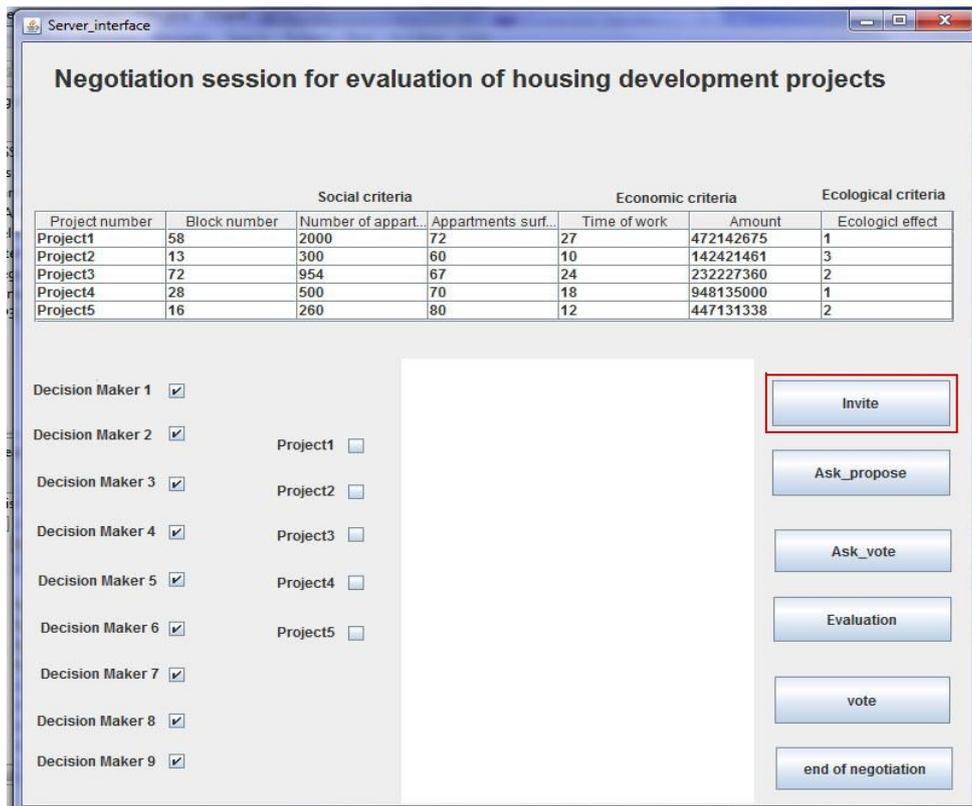


Figure 6.20. Invitation des décideurs pour l'évaluation des projets urbains [Benatia et al. a 2016].

Chaque décideur recevant l'invitation qui est l'annonce du début de la session de négociation, va accepter ou refuser l'invitation, comme l'indique la Figure 6.21.



Figure 6.21. Réponse des décideurs [Benatia et al. a 2016].

La deuxième étape de notre système de négociation est la conversation entre le décideur principal et l'ensemble des décideurs participants. La Figure 6.22 représente l'interface du décideur principal à travers laquelle il communique avec les autres décideurs. Dans ce stade, le décideur principal demande aux décideurs participants de donner le projet d'habitat de leur choix.

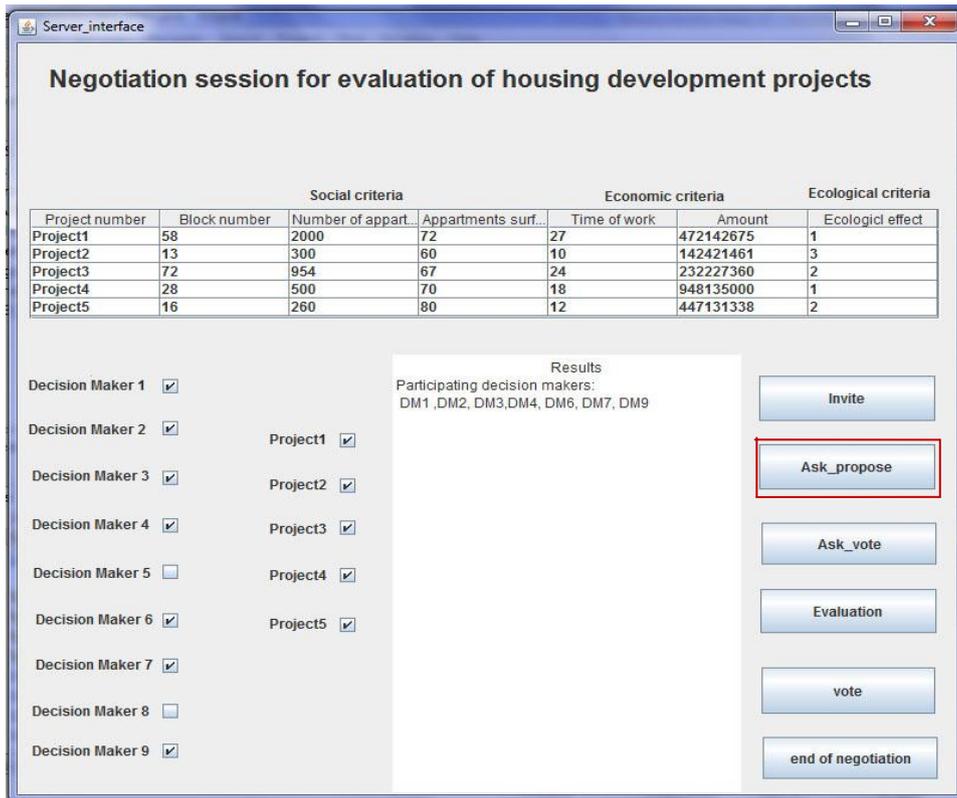


Figure 6.22. Demande de proposition [Benatia et al. a 2016].

Après cela, chaque décideur participant recevant ce message, choisit le projet d'habitat en fonction de ses préférences et ses connaissances.

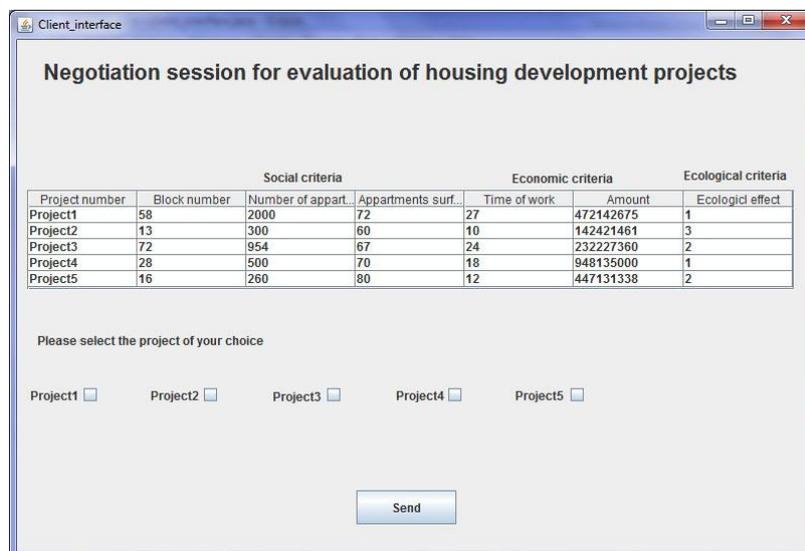


Figure 6.23. Choix des projets urbains par les décideurs [Benatia et al. a 2016].

Une fois que tous les décideurs participants ont donné leurs choix, le décideur principal demande d'effectuer une évaluation automatique des votes réalisés par les décideurs comme représenté sur la Figure 6.24.

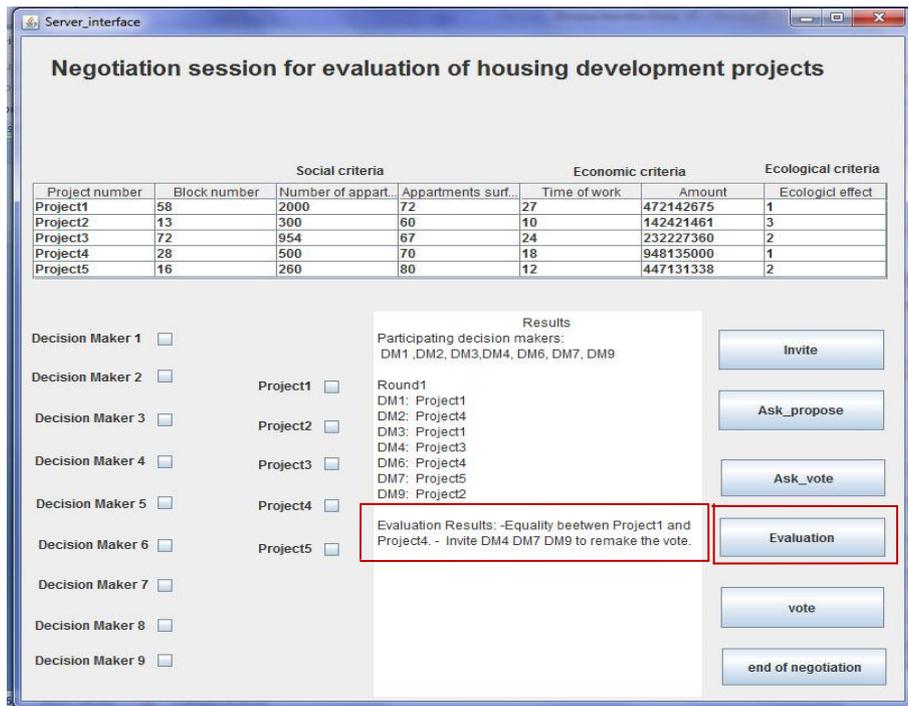


Figure 6.24. Evaluation des votes.

Les résultats de l'évaluation montrent qu'il y a une égalité entre deux projets (project1 et project4) ; dans ce cas, le décideur principal demande aux décideurs participants (qui n'ont pas choisi les projets à égalité c'est-à-dire project1 et projec4) de refaire un autre vote avec la liste des projets en égalité.

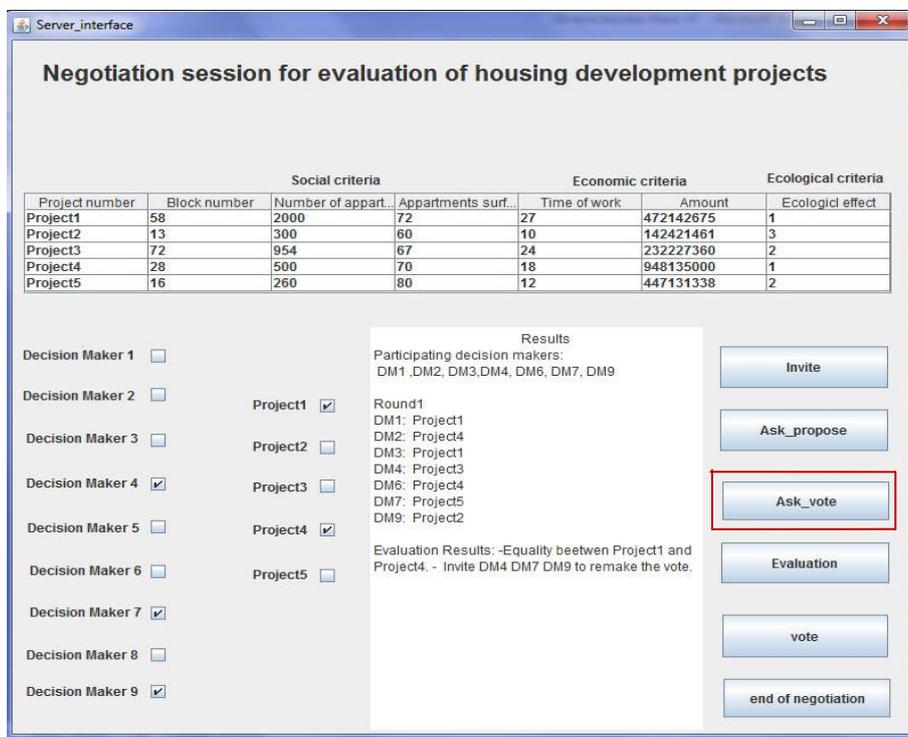


Figure 6.25. Demande de vote pour les projets en égalité [Benatia et al. a 2016].

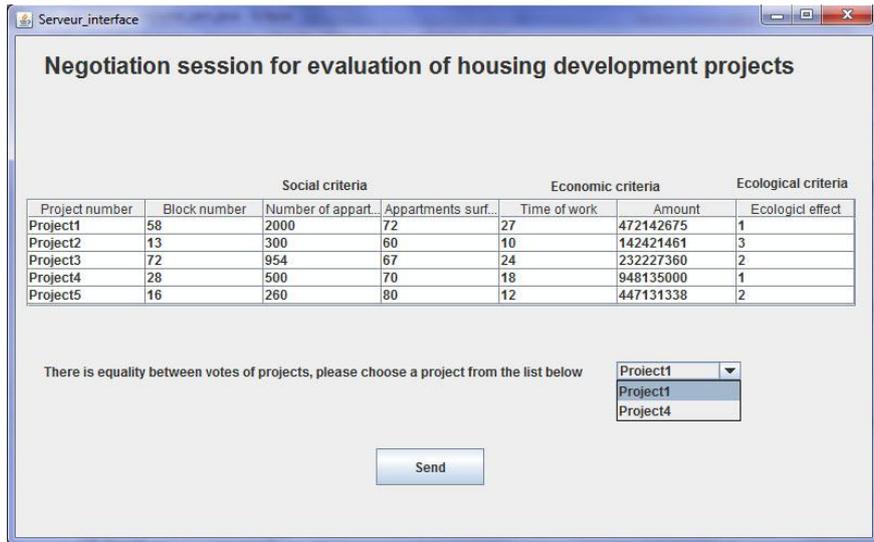


Figure 6.26. Choix entre les projets en égalité [Benatia et al. a 2016].

Lorsque le deuxième tour de vote est terminé, le décideur principal effectue à nouveau une évaluation des résultats. La Figure 6.27 montre le résultat de l'évaluation des projets d'habitat. Si le résultat de l'évaluation est l'égalité entre les votes alors, le principal effectue lui-même un vote en cliquant sur "vote" sinon il notifie tous les décideurs participants par la fin de la négociation comme représenté à la Figure 6.28.

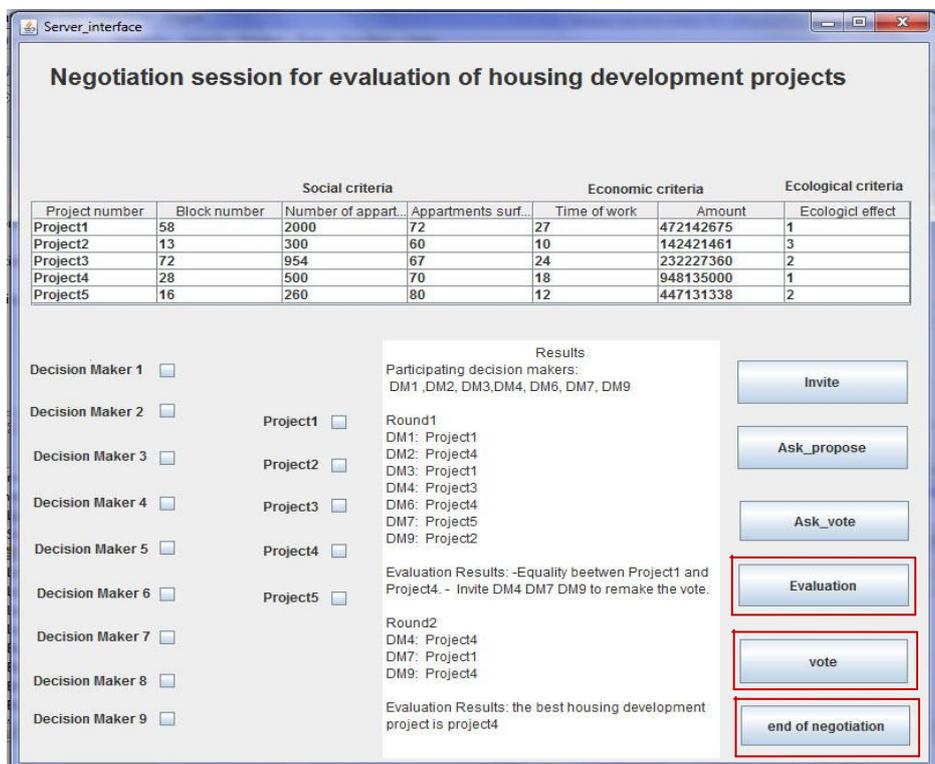


Figure 6.27. Evaluation du deuxième tour de vote [Benatia et al. a 2016].



Figure 6.28. Notification de la fin de la négociation [Benatia et al. a 2016].

## 6.5. Conclusion

Tout au long de ce chapitre, nous avons exposé et détaillé les étapes d'implémentation et de mise en place de nos systèmes d'aide à la décision qui permettent d'apporter une aide considérable aux décideurs de la ville. Nos outils décisionnels permettent de traiter la problématique d'évaluation des projets urbains en profitant des avantages du Cloud Computing, des méthodes de l'analyse multicritère et des protocoles de négociation.

# Conclusion Générale

De nos jours, on note la complexité et l'évolution rapide de l'environnement du décideur qui s'explique par le développement des ordinateurs et de la technologie de l'information (IT), et aussi à la complexité des décisions, l'ouverture au marché international, la forte compétition... Cependant, les problèmes décisionnels ont atteint le domaine géographique, et plus particulièrement le domaine de la planification territoriale urbaine. Ainsi, les villes et les zones urbaines jouent un rôle primordial en matière de développement durable dans la mesure où ces territoires sont soumis à des évolutions qui peuvent compromettre sérieusement l'équilibre économique, écologique et social. Pour permettre à une ville de devenir intelligente et d'assurer le développement durable du territoire urbain, les TIC (Technologie de l'information et de la communication) peuvent être considérées comme l'un des outils permettant l'évolution de la ville.

Le thème de notre étude est multidisciplinaire et s'articule autour de différents domaines : la planification urbaine, l'aide à la décision multicritère, la négociation et le Cloud Computing.

Les problèmes de la planification urbaine sont généralement mal définis du point de vue mathématique et de nature multidimensionnelle et qui nécessitent la détermination de plusieurs caractéristiques conflictuelles qui n'ont pas la même importance ; plusieurs acteurs sont impliqués avec intérêts, opinions et buts conflictuels. Tout cela peut engendrer des conflits entre les différents décideurs, ce qui peut affecter la décision finale.

Le rapport entre différents décideurs (acteurs) provenant de domaines différents et opposés disposant de leurs propres intérêts, objectifs et préférences qui sont souvent conflictuelles, ainsi que la multiplicité et la diversité des critères des projets urbains nous a poussée à chercher de nouvelles stratégies permettant d'aider les décideurs dans leurs processus de décision afin de minimiser le conflit et le débat entre eux.

Notre première contribution consistait à proposer une approche qui permet de répondre à la problématique d'évaluation des projets urbains et celle du conflit entre les décideurs. Cette approche est basée sur la combinaison d'un système d'aide à la décision et la nouvelle technologie Cloud Computing, afin de profiter de ses avantages à savoir : la facilité de communication et d'accessibilité sans contraintes temporelles ou géographiques, la réduction des coûts, la diminution du temps de traitement, de développement et de déploiement. Notre modèle décisionnel que nous avons proposé permet :

- de spécifier les paramètres subjectifs des projets urbains pour chaque décideur.
- d'effectuer une analyse multicritère pour chaque décideur en utilisant la méthode PROMETHEE II permettant de classer les projets urbains du meilleur au moins bon.
- d'exécuter un processus de négociation en employant la procédure Hare qui permet d'élire entre plusieurs rangements effectués par l'analyse multicritère et de fournir la décision finale à l'ensemble des décideurs (le meilleur projet urbain).

Nous avons constaté que la méthode d'analyse multicritère et celle de la négociation sont des méthodes qui fournissent des décisions automatiques où les décideurs ne participent pas au processus de décision ; cependant les résultats peuvent ne pas les convaincre, et donc il y aura un débat entre eux ; pour ces raisons nous avons proposé une deuxième contribution qui prend en considération l'opinion des décideurs. Nous avons suggéré un nouveau protocole de négociation inspiré du protocole le plus célèbre dans le domaine de négociation qui est le protocole Contract Net. Notre propre protocole est composé de trois phases principales : la phase d'authentification qui représente le début du processus de négociation, la phase de conversation dans laquelle une négociation se déroule entre les différents décideurs et la phase de fin de décision qui représente la fin du processus de négociation et notifie les décideurs par la décision finale (meilleur projet urbain).

Nos perspectives à court terme consistent à rendre notre système décisionnel un système participatif. Dans le but d'un développement durable, les organismes municipaux font face depuis un certain nombre d'années à la nécessité de consulter la population et de lui faire une place dans le débat sur la prise des décisions dans le domaine de l'aménagement du territoire et d'urbanisme [Gauthier et Gagnon 2013]. Afin d'assurer une efficacité de notre système d'aide à la décision dédié à la planification urbaine, l'introduction et la participation du citoyen au processus décisionnel est nécessaire.

Nous projetons d'autres perspectives à long terme, nous proposons de combiner notre première et deuxième contribution et d'intégrer les systèmes multi-agents en utilisant des agents virtuels qui seront dotés de capacités de raisonnement et de communication. La technologie Multi Agents a déjà fait ses preuves dans de nombreux domaines par leur capacité de modélisation, ils permettent de représenter les interactions entre différents acteurs pouvant coopérer, négocier et communiquer.

Nous envisageons que notre outil décisionnel qui répond à la problématique d'évaluation des projets urbains soit :

- doté d'une base de données comportant les différents projets urbains avec leurs caractéristiques.
- doté d'un module de système multi-agents composé de deux types de décideurs : coordinateur et participants où chaque décideur sera représenté par un agent.
- doté d'un module aide à la décision multicritère qui effectue une analyse multicritère en utilisant la méthode PROMETHEE II afin de ranger les projets urbains du meilleur au moins bon dans un vecteur de rangement.
- doté d'un module de négociation qui sera composé d'un protocole de négociation automatique permettant de trouver un compromis entre les différents agents.
- intégré dans un Cloud privé pour l'améliorer et profiter des avantages du Cloud Computing.

Le système décisionnel envisagé, permet à chaque agent participant d'exprimer ses points de vue et ses préférences par l'introduction des poids des critères des projets urbains, les poids de chaque agent vont passer vers l'analyse multicritère qui permet de ranger les projets urbains du meilleur au moins bon en utilisant la méthode PROMETHEE II, le résultat des rangements sera déposé sur un vecteur de rangement spécifique à chaque décideur. Par la suite l'agent coordinateur négocie en utilisant un protocole de négociation inspiré du protocole Contract Net ; la négociation va se dérouler en plusieurs tours jusqu'à ce qu'un compromis qui satisfasse la

majorité des agents soit trouvé. L'agent coordinateur propose aux agents participants un contrat (projet urbain) où chaque participant va soit l'accepter ou le refuser selon son vecteur de rangement. Dans le cas où l'agent coordinateur reçoit un grand nombre de refus, il modifie le contrat (projet urbain) pour le prochain tour.

Nous estimons que cette proposition peut, à la fois, rendre automatique la prise de décision et la négociation. Elle permet de trouver des solutions et les évaluer en employant des méthodes de raisonnement complexe. Pour cela, nous optons pour une modélisation des préférences à l'aide des techniques dérivées de l'aide à la décision multicritère permettant la construction d'outils adaptés et capable de remplacer l'acteur (décideur) sur des problèmes complexes.

# Publications et Communications

Les publications et les communications relatives à cette thèse sont les suivantes :

## Publications Internationales

1. Imene BENATIA, Mohamed Ridda LAOUAR, Hakim BENDJENNA and Sean B. EOM. Cloud-Based Multi-Criteria Decision Support System for Selecting Urban Housing Projects. International Journal Human Systems Management (HSM), Volume 35, Issue 2, 2015.
2. Imene BENATIA, Mohamed Ridda LAOUAR, Sean B. EOM and Hakim BENDJENNA. Incorporating the Negotiation Process in Urban Planning DSS. International journal of Information Systems in the Service Sector (IJISSS). Volume 8, Issue 2, 2016.
3. Imene BENATIA, Mohamed Ridda LAOUAR, Hakim BENDJENNA and Sean B. EOM. Implementing a Cloud-based Decision Support System in a Private Cloud: The Infrastructure and the Deployment Process, International Journal of Decision Support System Technology (IJDSST). Volume 8, Issue 1, 2016.

## Communications Internationales

1. Imene BENATIA, Mohamed Ridda LAOUAR, Hakim BENDJENNA. Cloud based Decision Support System for Urban Planning. International Conference on Information Technology for Organization Development (IT4OD'2014), Tebessa, Algeria. 19 -20 October 2014.
2. Imene BENATIA, Ridda LAOUAR, Hakim BENDJENNA and Sean EOM. Implementing a Cloud-based Decision Support System in a Private Cloud: The Infrastructure and the Deployment Process .1st EWG-DSS International Conference on Decision Support System Technology (ICDSST2015). Belgrade, Serbia. 27-29 May 2015.
3. Imene BENATIA, Mohamed Ridda LAOUAR, Sean B. EOM and Hakim BENDJENNA. New negotiation protocol for urban projects. International Conference on Pattern Analysis and Intelligent Systems (PAIS 2015), Tebessa, Algeria. 26-27 October 2015.
4. Imene BENATIA, Mohamed Ridda LAOUAR, Sean B. EOM and Hakim BENDJENNA. Prototype of decisional system based Cloud for evaluation of urban transport projects. ASD 2016 (Conférence sur les Avancées des Systèmes Décisionnels). Annaba, Algérie. 14-16 Mai 2016.

## Communications nationales

1. Imene BENATIA et Mohamed Ridda LAOUAR. Approche basée Cloud pour la planification territoriale urbaine. JOMI 2014 (Journées Ouvertes sur les Mathématiques et l'Informatique) Tébessa, Algérie. 14-15 Mai 2014.

2. Imene BENATIA, Mohamed Ridda LAOUAR, Hakim BENDJENNA and Sean B. EOM. Vers un nouvel système de négociation multi-agents pour la planification urbaine. JOMI 2015 (Journées Ouvertes sur les Mathématiques et l'Informatique) Tébessa, Algérie.25-26 Mai 2015.

# Bibliographie

- [Ahris et al. 2009] Ahris Y., Siti Zalina A., & Susilawati S., Decision Support System for Urban Sustainability Planning in Malaysia, *Malaysian Journal of Environmental Management*, 10(1), 2009.
- [Akbi et Zehri 2013] Akbi K., & Zehri M., Etude et mise en place d'une solution Cloud Computing privé au sein de l'université de Ouargla, *Mémoire de Master Académique de l'Université de Ouargla*, Algérie, 2013.
- [Alam et al. 2005] Alam S-J., Hillebrandt F., & Schillo M., Sociological Implications of Gift Exchange in Multiagent Systems, *Journal of Artificial Societies and Social Simulation*, 8(3), 2005.
- [Alty et al. 1994] Alty J-A., Griffiths D., Jennings N-R., Mamdani E-H., Struthers A., & Wiegand M-E., ADEPT: Advanced decision environment for process tasks: overview & architecture, *Conference on expert systems*, Cambridge, UK, pp. 359-371, 1994.
- [Arab 2004] Arab N., L'activité de projet dans l'aménagement urbain : processus d'élaboration et modes de pilotage Les cas de la ligne B du tramway strasbourgeois et d'Odysseum à Montpellier, *Thèse de doctorat de l'Ecole des Ponts et Chaussées*, Paris, France, 2004.
- [Ascher 1991] Ascher F., Projet public et réalisations privées. Le renouveau de la planification des villes, *Annales de la Recherche Urbaine*, n° 51, pp. 4-15, 1991.
- [Ascher 1995] Ascher F., Métapolis, ou l'avenir des villes, *Edition Odile Jacob*, Paris, France, 1995.
- [Aupaix 2010] Aupaix X., Analyse et conception d'un marketplace basé sur le logiciel D-Sight, *Mémoire d'ingénieur de l'Université Libre de Bruxelles*, Belgique, 2010.
- [Azzabi 2010] Azzabi D-A., Optimisation multicritère de la fiabilité : application du modèle de goal programming avec les fonctions de satisfactions dans l'industrie de traitement de gaz, *Thèse de doctorat de l'université de Sfax*, Tunisie, pp. 122, 2010.
- [Baun et al. 2011] Baun C., Kunze M., Nimis J., & Tai S., Cloud Computing: Web-Based Dynamic IT Services, *Springer Berlin Heidelberg*, 2011.
- [Belton et Stewart 2002] Belton V., & Stewart T., Multiple Criteria Decision Analysis: An Integrated Approach, *Kluwer Academic*, Dordrecht, Holland, 2002.
- [Benatia et al. 2015] BENATIA I., LAOUAR M-R., BENDJENNA H., & EOM S., Cloud-Based Multi-Criteria Decision Support System for Selecting Urban Housing Projects. *International Journal Human Systems Management (HSM)*, 35(2), 2015.
- [Benatia et al. a 2016] BENATIA I., LAOUAR M-R., EOM S., & BENDJENNA H., Incorporating the Negotiation Process in Urban Planning DSS. *International journal of Information Systems in the Service Sector (IJISSS)*, 8(2), 2016.
- [Benatia et al. b 2016] BENATIA I., LAOUAR M-R., BENDJENNA H., & EOM S., Implementing a Cloud-based Decision Support System in a Private Cloud: The Infrastructure and the Deployment Process, *International Journal of Decision Support System Technology (IJDSST)*, 8(1), 2016.

- [Bouimaza 2005]** Bouimaza N., Villes réelles, ville projetées : villes maghrébines en fabrication, *Edition Maisonneuve- Larose*, pp. 26, 2005.
- [Bouron 1992]** Bouron T., Structures de communication et d'organisation pour la coopération dans un univers multi-agents, *Thèse de doctorat de l'Université de Paris 6*, France, 1992.
- [Bouyssou et al. 2000]** Bouyssou D., Marchant T., Pirlot M., Perny P., Tsoukiàs A., & Vincke P., Evaluation and decision models: a critical perspective. *Kluwer Academic*, Dordrecht, Hollande, 2000.
- [Brans et Vincke 1985]** Brans J-P., & Vincke P., A Preference Ranking Organization Method, *Management Science*, 31(6), pp.647-656, 1985.
- [Brans 1993]** Brans J-P., Méthodes PROMETHEE-GAIA, *Support de cours*, mars 1993.
- [CEMAT 2003]** La Conférence Européenne des Ministères responsables de l'Aménagement du Territoire (CEMAT), 2003.
- [CFU 2001]** Code du Foncier et de l'Urbanisation (CFU), Complication de textes juridiques législatifs et réglementaires de la république algérienne, *Berti Editions*, 2001.
- [Collins et al. 1998]** Collins J., Youngdahl B., & Jamison S., Market Architecture for Multi-agent Contracting, *2nd International Conference Autonomous Agents*, 1998.
- [David et Smith 1980]** David R., & Smith R, Negotiation as a Metaphor for Distributed Problem Solving, *Artificial Intelligence*, 20(1), pp.63-109, 1980.
- [Dawood et Alshawi 2009]** Dawood I., & Alshawi M., Decision Support Systems (DSS) Model for the Housing Industry. *The 2<sup>nd</sup> International Conference on Developments in eSystems Engineering (DESE)*. 2009.
- [Debord 1987]** Debord B., Axiomatisation de procédures d'agrégation de préférences, *Thèse de doctorat de l'Université scientifique technologique et médicale de Grenoble*, France, 1987.
- [Derycke 1982]** Derycke H., Economie et planification urbaine, *presses universitaires de France*, pp. 298, 1982.
- [Destais 2011]** Destais G., Les théorisations économiques du développement durable Proposition de décryptage critique, *Colloque international francophone sur le développement durable : débats et controverses*, Clermont-Ferrand, France, 2011.
- [Di Lecce et al. 2010]** Di Lecce V., Amato A., Soldo D., & Giove A., A Multi Agent System Modelling an Intelligent Transport System, *Modeling Simulation and Optimization Modeling Simulation and Optimization - Focus on Applications*, pp.135-145, 2010.
- [Di Lecce et Amato 2011]** Di Lecce V., & Amato A., Route planning and user interface for an advanced intelligent transport system, *IET Intelligent Transport Systems*, 5(3), pp. 149-158, 2011.
- [Dind et Cunha 2011]** Dind J-P., & Da Cunha A., La gestion de projets urbains, Projets d'aménagement concertés dans des secteurs déjà bâtis : exemples en Suisse Romande, *Mémento à l'usage des responsables de projet*, *Université de Lausanne*, Suisse, 2011.
- [Edwards 1971]** Edwards W., Social utilities, *Engineering Economist, Summer Symposium Series* 6, pp. 119-129, 1971.

- [Eom 2007]** Eom S-B., The Development of Decision Support Systems Research: A Bibliometrical Approach. *The Edwin Mellen Press, Lewiston, New York, 2007.*
- [Fayech et al. 2002]** Fayech B., Maouche S., Hammad S., & Borne P., Multi-agent decision-support system for an urban transportation network. *Automation Congress, Proceedings of the 5th Biannual World .2002.*
- [Ferrand 1997]** Ferrand N., Modèles Multi-Agents pour l'Aide à la Décision et la Négociation en Aménagement du Territoire, *Thèse de Doctorat de l'Université Joseph-Fourier, Grenoble, France, 1997.*
- [Fishburn 1970]** Fishburn P-c., Utility Theory for Decision Making, *Wiley, pp.7. New York, 1970.*
- [Fouchet et Lopez 2000]** Fouchet R., & Lopez J.R., Processus de décision et aménagement territorial, *4ème rencontre ville et management, 2000.*
- [Gambino 2010]** Gambino M., Politiques d'aménagement du territoire, *Centre d'études et de prospective -service de la statistique et de la prospective, 2010.*
- [Gauthier et Gagnon 2013]** Gauthier M., & Gagnon L., La participation du public dans les démarches municipales de développement durable : principes, conditions de réussite, enjeux et dispositifs, *Les cahiers du CRGRNT Numéro 130, 2013.*
- [Genard & Pirlot 2002]** Genard J-L., & Pirlot M., Multiple criteria decision aid in a philosophical perspective, *International Series in Operations Research & Management Science, pp. 89-117, 2002.*
- [Ginting 2000]** Ginting R., Intégration du système d'aide a la décision multicritères et du système d'intelligence économique dans l'ère concurrentielles, *Thèse de doctorat de l'Université de droit et des sciences d'Aix-Marseille, France, 2000.*
- [Graillet et Waaub 2006]** Graillet D., & Waaub J-P., Aide à la décision pour l'aménagement du territoire : méthodes et outils, *science publications, pp. 21,2006.*
- [Guttman et al. 1998]** Guttman R-H., Moukas A-G., & Maes P., Agent Mediated electronic Commerce: a survey, *Knowledge Engineering Review, 1998.*
- [Guttouni et al. 1999]** Guitouni A, Martel J-M., & Vincke P., Un Cadre de Référence pour le Choix d'une Procédure d'Agrégation Multicritère, *Document de travail de l'Université Laval, Québec, Canada, 1999.*
- [Hammami 2003]** Hammami A., Modélisation technico-économique d'une chaine logistique dans une entreprise réseau, *Thèse de doctorat de l'université Laval, Québec, Canada, 2003.*
- [Hasan 2010]** Hasan M-K., A Framework for Intelligent Decision Support System for Traffic Congestion Management System. *Engineering, pp.270-289, 2010.*
- [Henderson 1997]** Henderson J-A., Community Planning And Development, *Lecture notes Spring Semester, Whittier College, California. 1997.*
- [Huang et al.]** Huang W., Zhang X., and Wei X., An Improved Contract Net Protocol with Multi-Agent for Reservoir Flood Control Dispatch, *Journal of Water Resource and Protection, pp.735-746, 2011.*
- [Humphrey et al. 1983]** Humphreys P-C., Svenson O., & Vári A., Analysis and aiding decision processes, *North-Holland, Amsterdam, 1983.*

- [Ineichen 2007]** Ineichen J., Recherche-Action Au sein de la communauté «Chico Mendes», *Ecole polytechnique fédérale de Lausanne, Suisse*, 2007.
- [Ingallina 2001]** Ingallina P., Le projet urbain, *Editions Que sais je ?*, pp.3, France, 2001.
- [Jacob et Varone 2003]** Jacob S., et Varone F., Évaluer l'action publique : état des lieux et perspectives en Belgique, *Gand : Academia Press*, pp.243, 2003.
- [Jun et Yikui 2009]** Jun D., & Yiku M., Intelligent Decision Support System for Road Network Planning, *Proceedings of the ISECS International Colloquium on Computing, Communication, Control, and Management*, Sanya, Chine, 2009.
- [Kazi 2009]** Kazi T-A., La modélisation des préférences du décideur dans le modèle du goal programming, *Thèse de doctorat en science de gestion de l'Université de Tlemcen*, Algérie, pp. 31-33, 2009.
- [Keeney 1992]** Keeney R-L, Valued-focused thinking: A path to creative decision, *Harvard University press*, Cambridge, UK, pp. 123, 1992.
- [Keeney et Raifa 1976]** Keeney R., & Raifa H., Decision with Multiobjectives, Preferences and Value Trade-Offs , *Cambridge University Press*, Cambridge, UK, 1976.
- [Keita et al. 07]** Keita A., Laurini R., Roussey C., & Zimmerman M., Towards an ontology for urban planning: The Towntology project. *CD-ROM Proceedings of the 24th UDMS Symposium, Chioggia*, 12 (1). 2007.
- [Kherbache 2010]** Kherbache V., Cloud Computing, *Licence professionnelle de l'Institut Universitaire de Technologie Nancy Charlemagne*, France, 2010.
- [Kraus et Wilkenfeld 1991]** Kraus S., & Wilkenfeld J., Negotiation and cooperation in multi-agent environments, *The Twelfth International Joint Conference on Artificial Intelligent*, 1991.
- [Laaribi 2000]** Laaribi A., SIG et analyse multicritère, *Hermes science publications*, Paris, pp. 51, 2000.
- [Labeled 2007]** Labeled K., Expérimentation des algorithmes génétiques multiobjectifs dans un processus décisionnel multicritère en aménagement du territoire, *Mémoire de magistère de l'Université Es-Senia*, Oran, Algérie, 2007.
- [Laborde 1994]** Laborde P., Les espaces urbains dans le monde, *éditions NATHAN*, pp.183, 1994.
- [Landry 1998]** Landry M., L'aide multicritère à la décision comme support à la construction du sens dans l'organisation, *Système d'information et management*, vol 3, 1998.
- [Laouar 2005]** Laouar R., Contribution pour l'aide à l'évaluation des projets de déplacement urbain, *Thèse de doctorat de l'Université de Valenciennes et du Hainaut-Cambrésis*, France, 2005.
- [Laurini 1998]** Laurini R., Groupware for urban planning: an introduction. *Computers, Environment and Urban Systems*, 22 (4), pp.317-333. 1998.
- [Laurini 2002]** Laurini R., Information systems for urban planning: a hypermedia cooperative approach, *CRC Press*, 2002.
- [Laurini 2007]** R Laurini L., Pre-consensus ontologies and urban databases, *Ontologies for Urban Development*, pp.27-36. 2007.

- [Lequin 1992]** Lequin M., Analyse du développement durable en loisir et tourisme: le cas du mont royal, Mémoire de l'Université du Québec, Canada, 1992.
- [Lévine et Pomerol 1989]** Lévine P., et Pomerol J., Systèmes interactifs d'aide à la décision et systèmes experts, *Hermes science publications*, 1989.
- [Lévry et al. 2004]** Lévry J., Golay F., Schuler M., Macquat J., & Beaudé B., Théorie du territoire, *Cours d'aménagement du territoire de l'école polytechnique fédérale de Lausanne*, Suisse, 2004.
- [Lichfield et al. 1975]** Lichfield N., Kettle P., & Whitbread M., Evaluation in the planning process, *Pergamon Press*, Oxford , Angleterre , pp. 326, 1975.
- [Longfei et Hong 2009]** Longfei W., & Hong C. Cooperative Parking Negotiation and Guidance Based on Intelligent Agents. *International Conference on Computational Intelligence and Natural Computing*. 2009.
- [Luce et Raiffa 1957]** Luce R-D., & Raiffa H., Games and decisions, *Wiley*, pp. 16, New York, 1957.
- [Mahjoub 2011]** Mahjoub M., Étude et expérimentations du Cloud Computing pour le monitoring des applications orientées services, *Mémoire de Master de l'Université Sfax*, Tunisie, 2011.
- [Maktav et al. 2011]** Maktav D., Jurgens C., Siegmund A., Sunar F., Esbah H., Kalkan K., Uysa C., Mercan O.Y., Akar I., Thunig H., & Wolf N., Multi-criteria Spatial Decision Support System for Valuation of Open Spaces for Urban Planning. *The 5th International Conference on Recent Advances in Space Technologies (RAST)* .2011.
- [Mammeri 2013]** Mammeri M-L., Une approche d'aide multicritère à la décision pour l'évaluation du confort dans les trains, *Thèse de doctorat de l'Université Paris-Dauphine*, France, 2013.
- [Mangin 2002]** Mangin E., Concertation et vote électronique, *Etude bibliographique*, 2002.
- [Mareschal 2014]** Mareschal B., Stochastic multicriteria decision making under uncertainty, *International transactions in Operation Research*, p 58, 2014.
- [Martel 1999]** Martel J-M., L'aide multicritère à la décision: méthodes et applications, *Canadian Operational Research Society (CORS-SCRO Bulletin)*, 33(1), pp. 6, 1999.
- [Masboungi 2002]** Masboungi A., Projets urbains en France, *Editions du Moniteur*, Paris, France, 2002.
- [MATE 2004]** Ministère de l'Aménagement du Territoire et de l'Environnement (MATE), Aménagement de l'Algérie 2020, Alger, Algérie, 2004.
- [Maystre et al. 1994]** Maystre L-Y., Pictet J., & Simos J., Méthodes multicritères ELECTRE, *Presses Polytechniques et Universitaires Romandes*, Lausanne, Suisse, pp. 323, 1994.
- [Mell et Grance 2011]** Mell P., & Grance T., The NIST definition of Cloud computing (draft), *NIST special publication*, 800(145), 2011.
- [Merlin et Choay 1988]** Merlin P., & Choay F., Dictionnaire de l'urbanisme et de l'aménagement, *Presses universitaires de France*, pp. 502, 1988.

**[MIAT 2005]** Ministère de l'Intérieur et de l'Aménagement du Territoire (MIAT), un projet d'aménagement durable pour le Luxembourg : Les principales idées du programme directeur, Luxembourg, 2005.

**[Miah et Ahamed 2011]** Miah S.J., & Ahamed R., A Cloud-Based DSS Model for Driver Safety and Monitoring on Australian Roads. *International Journal of Emerging Sciences*, 1(4), pp. 634-648? 2011.

**[Mintzberg 1979]** Mintzberg H., The structuring of Organization, *Prentice Hall international editions*, 1979.

**[Montalvo Arangoa et al. 2014]** Montalvo Arangoa I., Izquierdo J-S., Campbell E-O-G., & Pérez-García R., Cloud-Based Decision Making in Water Distribution Systems, *Conference on Water Distribution System Analysis*, Bari, Italie, 2014.

**[Moreira de Oliveira et al. 2012]** Moreira de Oliveira T-H, Painho M., & Henriques R., A spatial decision support system for the Portuguese public transportation sector. *IWGS '12 Proceedings of the Third ACM SIGSPATIAL International Workshop on GeoStreaming*. 2012.

**[Mousseau 1993]** Mousseau V., Problèmes liés à l'évaluation de l'importance relative des critères en aide multicritère à la décision : réflexions théoriques, expérimentations et implémentations informatiques, *Thèse de doctorat de l'Université Paris-Dauphine*, France, 1993.

**[Nafi et Werey 2010]** Nafi A., & Werey C-Y., Aide à la décision multicritère : introduction aux méthodes d'analyse multicritère de type ELECTRE , Cours de l'Université de Strasbourg, France, pp. 11-13, 2010.

**[Natividade et al. 2007]** Natividade-Jesus E., Coutinho-Rodrigues J., & Henggeler Antunes C., A multicriteria decision support system for housing evaluation, *International journal of Decision Support Systems*, Elsevier, vol 43, pp. 779–790, 2007.

**[Noriega 1998]** Noriega P., Agent mediated auctions: The fishmarket metaphor, PhS thesis of University of Barcelona, Spain, 1998.

**[Oliveira et Pinho 2008]** Oliveira V., & Pinho P., Evaluation in urban planning: from theory to practice, *1st Annual Conference on Planning Research-Evaluation in Planning*, Porto, Portugal, pp. 31-49. 2008.

**[Pelletier 1999]** Pelletier C., Application des techniques d'aide à la décision à la planification sanitaire régionale, *Thèse de doctorat de l'Université Joseph Fourier*, Grenoble, France, 1999.

**[Pelletier 2009]** Pelletier M., La décision territoriale en conflit : Analyse spatiale de l'activité conflictuelle, Ville de Québec, 1989 à 2000, *Thèse de doctorat de l'Université Laval*, Quebec, Canada, 2009.

**[Perny 1998]** Perny P., Multicriteria filtering methods based on concordance and non-discordance principales, *Annals of Operations Research Journal*, Springer, pp. 137, 1998.

**[Pomerol 1992]** Pomerol J-C., Aide à la décision et IA, *Conférence internationale de l'intelligence Artificielle une discipline et un carrefour interdisciplinaire (AFIA)*, pp. 147-149. 1992.

**[Poulton 1994]** Poulton E-C., Behavioral decision theory: A new approach. *Cambridge University Press*, Cambridge, UK, 1994.

**[PROMATECH 2014]** PROMATECH, Professional Maritime Technologies, CLOUD-VAS a cloud based vessel allocation decision support system for Vessel Chartering , Istanbul, Turquie, 2014

**[Prunie 2011]** Prunie A., Le Cloud Computing : Réelle évolution ou simple évolution?, Wygwam bureau d'expertise technologique, 2011.

**[Rabinovich 2008]** Rabinovich A., Projet urbain: entre innovation et tradition dans l'action urbaine, *Observatoire universitaire de la ville et du développement durable*, Université de Lausanne, Suisse, 2008.

**[Rahmoun 2013]** Rahmoun N., La planification urbaine à travers les PDAU-POS et la problématique de la croissance et de l'interaction villes/villages en Algérie. Référence empirique à la Wilaya de Tizi- Ouzou, *Thèse de doctorat de l'Université de Tizi-Ouzou*, Algérie, 2013.

**[Ratti 2012]** Ratti C., Lecture, *Meet The GURU Media*, 2012.

**[Risse 2004]** Risse N., Evaluation environnementale stratégique, *Thèse de doctorat de l'Université Libre de Bruxelles*, Belgique, 2004.

**[Roussey et al. 2004]** Roussey C., Laurini R., Beaulieu C., Tardy Y., & Zimmermann M., Le projet Towntology : un retour d'expérience pour la construction d'une ontologie urbaine, *Revue internationale de Géomatique*, 14 (2), pp.217-237. 2004.

**[Roy 1968]** Roy B., Classement et Choix en Présence de Points de Vue Multiples (la Méthode Electre), *Revue française d'informatique et de recherche opérationnelle*, 2(8). pp. 57-75, 1968.

**[Roy et Bertier 1971]** Roy B., & Bertier P., La Méthode Electre II, une Méthode de Classement en Présence de Critères Multiples, *Direction Scientifique, Note de travail n°142,Sema*, Paris, France, pp. 25, 1971.

**[Roy 1978]** Roy B., Electre III, un Algorithme de Classement Basé sur une Représentation Floue des Préférences en Présence de Critères Multiples, *Cahiers du centre Etudes Recherche Opérationnelle(CERO)*, 20(1), pp. 3-24, 1978.

**[Roy 1985]** Roy B., Méthodologie multicritère d'aide à la décision, *Economica*, Paris, France, 1985.

**[Roy 1990 a]** Roy B., The outranking approach and the foundations of ELECTRE methods, *Theory and Decision*, pp. 155-83, 1990.

**[Roy 1990 b]** Roy B., Decision-aid and decision-making, *European Journal of Operational Research*, pp. 17-35, 1990.

**[Roy et Bouyssou 1993]** Roy B., & Bouyssou D., Aide multicritère à la décision : méthodes et cas, *Economica*, Paris, France, pp.19-21, 1993.

**[Saaty 1980]** Saaty T-L., The Analytic Hierarchy Process, *McGraw-Hill companies*, New York , 1980.

**[Sahli 2010]** Sahli S., Une approche formelle pour la négociation automatique entre agents (application e-commerce), *Thèse de Magister de l'Université de Mohamed Khider*, Biskra, Algérie, 2010.

**[Schärlig 1985]** Schärlig A., Décider sur plusieurs critères: Panorama de l'aide à la décision multicritère, *Presses Polytechniques et Universitaires Romandes*, Lausanne, Suisse, pp. 33, 1985.

**[Sehgal 2012]** Sehgal A., Introduction to OpenStack, Running a Cloud Computing Infrastructure With OpenStack, *6th International Conference on Autonomous Infrastructure, Management and Security*, Luxembourg, 2012.

**[Shanian et Savadogo 2006]** Shanian A., & Savadogo O., A non-compensatory compromised solution for material selection of bipolar plates for polymer electrolyte membrane fuel cell (PEMFC) using ELECTRE IV, *International journal of Electrochimica Acta* , 51(25), 2006.

**[Shi et Li 2010]** Shi Y., & Li J., Improving the Decisional Context: New Integrated Decision Support System for Urban Traffic-related Environment Assessment and Control. *International Conference on Mechanic Automation and Control Engineering (MACE)*. 2010.

**[Simon 1983]** Simon H-A., Administration et processus de décision, *Economica*, Paris, France, pp. 8, 1983.

**[SNAT 1995]** SNAT Schéma National d'Aménagement du Territoire, Demain l'Algérie: l'aménagement du territoire, *Ministère de l'équipement et de l'aménagement*, Volume I, 1995.

**[Sprague 1987]** Sprague R., DSS in context, *Decision Support System*, 3(197), pp. 202, 1987.

**[Takahashi et al. 2013]** Takahashi J., Kanamori R., & Ito T., Evaluation of Automated Negotiation System for Changing Route Assignment to Acquire Efficient Traffic Flow. *The 6th International Conference on Service-Oriented Computing and Applications (SOCA)*. 2013.

**[Teller et al. 2007]** Teller J., Keita A-K., Roussey C., & Laurini R., Urban Ontologies for an improved communication in urban civil engineering projects, *Presentation of the COST Urban Civil Engineering Action C21 "TOWNTOLOGY"*, *Cybergeog: European Journal of Geography*. 2007.

**[Tomas 98]** Tomas F., Vers une nouvelle culture de l'aménagement des villes, *In: Toussaint J-Y, & Zimmermann M., Projet urbain, ménager les gens, aménager la ville, Mardaga, Sprimont, Belgique* , pp. 13-34, 1998.

**[Tsoukiàs 2008]** Tsoukiàs A., From decision theory to decision aiding methodology, *European Journal of Operational Research*, 187(1), pp. 138-161, 2008.

**[Vaillancourt 1998]** Vaillancourt J., Évolution conceptuelle et historique du développement durable, *Regroupement national des Conseils Régionaux de l'environnement du Québec (RNCREQ)*, 1998.

**[Vansnick 1990]** Vansnick J-C., Measurement theory and decision aid, *Bana e Costa edition. Readings in multiple criteria decision aid:Springer-verlag*, pp. 8, 1990.

**[Vansnick 1995]** Vansnick J-C., L'aide multicritère à la décision: une activité profondément ancrée dans son temps, *Newsletter of the European Working Group*, pp.20, 1995.

**[Velte et al. 2010]** Velte A-T., Velte T-J., & Elsenpeter R., Cloud Computing: a practical approach, *The McGraw-Hill Companies*, 2010.

**[Verrons 2004]** Verrons. M-H., GenCA : un modèle général de négociation de contrat entre agents, *Thèse de Doctorat de l'Université des sciences et technologie de Lille*, France, 2004.

**[Vickrey 1961]** Vickrey W., Counterspeculation, auctions, and competitive sealed tenders. *Journal of Finance*, pp. 8-27, 1961.

**[Vilain 2007]** Vilain C., Méthode flowsort au problème du tri multicritère, Mémoire de Licence en sciences mathématiques de l'Université libre de Bruxelles, Belgique, pp. 15, 2007.

**[Vincke 1989]** Vincke P., L'aide multicritère à la décision, *Editions de l'Université de Bruxelles-Editions Ellipses*, Belgique, 1989.

**[Voogd 1983]** Voogd H., Multicriteria Evaluation for Urban and Regional Planning, *Pion Ltd*, London, 1983.

**[Voogd et Faludi 1985]** Voogd H., & Faludi A., Evaluation of complex policy problems : some concluding comments , *In Evaluation of complex policy problems*, pp. 210-206. 1985.

**[Wiliquet 2011]** Wiliquet C., Le développement durable, un concept multiforme, *Extrait de la revue En Question n°98 du centre AVEC*, Bruxelles, Belgique.

**[Yang et Chen 2002]** Yang X.M., & Chen, X.W., Integrated decision support system designed for Chinese city public transportation planning and operation, *The 5th International Conference on intelligent Transportation Systems*, 2002.

**[Yassa 2014]** Yassa S., Allocation optimale multi-contraintes des workflows aux ressources d'un environnement Cloud Computing, Thèse de doctorat de l'Université de Cergy-Pontoise, France, 2014.

**[Yujing et al. 2011]** Yujing W., Lin L., Hui Z., & Zhihua L., Decision Model of Urban Infrastructure Projects Based on Comprehensive Evaluation. *The 2nd International Conference on Emergency Management and Management Sciences (ICEMMS)*. 2011.

**[Zoller et Béguin 1989]** Zoller H-G., Béguin H., Aide à la décision : l'évaluation des projets d'aménagement, *Economica*, Paris, France, 1989.

# Webographie

- [1] AbiCloud, <http://community.abiquo.com/> (consulté le 02/01/2014)
- [2] Cloudbees, <https://www.cloudbees.com/> (consulté le 20/01/2014)
- [3] CloudBees AnyCloud,  
[http://pages.cloudbees.com/rs/cloudbees/images/AnyCloud\\_White\\_Paper\\_FR\\_Final.pdf](http://pages.cloudbees.com/rs/cloudbees/images/AnyCloud_White_Paper_FR_Final.pdf)  
(consulté le 10/01/2014)
- [4] Cloudify, <http://www.cloudify.cc/> (consulté le 20/01/2014)
- [5] CloudFoudry, <http://cloudfoundry.org/index.html> (consulté le 20/01/2014)
- [6] Eucalyptus, <http://www.eucalyptus.com/> (consulté le 03/01/2014)
- [7] IAAS, <https://www.servernest.com/infrastructure-as-a-service/> (consulté le 19/11/2015)
- [8] Outils open source, <http://philippe.scoffoni.net/5-outils-open-source-pour-le-cloud-computing/> (consulté le 03/01/2014)
- [9] Openstack, <https://www.openstack.org/> (consulté le 02/01/2014)
- [10] Platform as a service (PAAS) for cloud applications (Scalable cluster of services),  
<http://socialcompare.com/fr/comparison/platform-as-a-service-paas-for-cloud-applications-scalable-cluster-of-services> (consulté le 26/11/2015)
- [11] Xen, <http://wiki.xen.org/> (consulté le 03/01/2014)
- [12] Yes we Cloud, <http://www.yeswecloud.fr/cloud/a-chacun-son-type-de-cloud-1286.html>  
(consulté le 15/12/2015)
- [13] Zenoss, own IT, <http://www.zenoss.com> (consulté le 01/02/2016)

# Annexes

## Annexe A : Liste des acronymes

**AMQP** : «Advanced Message Queue Protocol»

**API**: «Application Programming Interface»

**AWS** : «Amazon Web Service»

**BDD** : «Base De Données»

**CLI**: «Comande Line Interface»

**CPU**: «Central Processing Unit»

**CRM** : «Customer Relationship Management»

**DB** : «Data Base»

**DSS** : «Decision Support System»

**EC2** : «Elastic Compute Cloud»

**HP** : «Hewlett Packard»

**I / O** : «Input/ Output»

**IBM** : «International Business Machines»

**IP** : «Internet Protocol»

**ISVs** : «Independent Software Vendors»

**IT** : « Information technology»

**JVM** : «Java Virtual Machine»

**MDSS**: « Multi-criteria Decision Support System»

**NASA** : «National Aeronautics and Space Administration»

**NIST** : «National Institute of Standards and Technology»

**PDA**: «Personal Digital Assistant»

**RAM** : «Random Access Memory»

**REDIS**: «REmote DIctionary Server»

---

**REST:** «Representational State Transfer»

**RH :** «Ressource Humaine»

**S3 :** «Simple Storage Service»

**SDK:** «Software development kit»

**SQL:** «Structured Query Language»

**UML :** «Unified Modeling Language»

**VE :** «Virtual Environment»

**VLAN :** « Virtual Local Area Network»

**VM :** « Virtual Machine»

**VPS :** « Virtual Private Server»

**WS :** « Web Service»

**XCP:** « Xen Cloud Platform»

---

## Annexe B: Lexique

**Azure** : une plateforme de Cloud Computing ouverte et flexible qui offre des services Cloud IaaS et PaaS.

**Cassandra** : est un SGBD NoSQL. Il permet la gestion de grandes quantités d'informations distribuées sur une multitude de serveurs, tout en garantissant spécialement des données disponibles de façon optimale et en supprimant les points individuels de défaillance.

**Datacenter**: est un site physique dans lequel il y a un ensemble d'équipements composants un système d'information de l'entreprise tel que les serveurs, les équipements de stockage,... Il est situé à l'extérieure ou bien à l'interne de l'entreprise.

**Datastore** : un référentiel de stockage persistant d'une collection de données, par exemple une base de données, un système de fichier ou un répertoire.

**ElasticFox / HybridFox** : des extensions de Firefox dans lequel sont rajoutées des tentatives visant à obtenir le meilleur des environnements de Cloud public et privé.

**Git** : un système distribué de contrôle libre et open source. Le Git a été conçu pour gérer les projets quelque soit sa taille (du plus petit au plus grand) avec rapidité et efficacité.

**Github** : est le plus grand hôte de dépôts Git. C'est le point central de la collaboration pour des millions de développeurs et de projets. Un grand pourcentage de tous les dépôts Git sont hébergés sur GitHub, et de nombreux projets open-source utilise le Git hub afin d'utiliser les Git hébergés, suivre les problèmes, revoir le code...

**HSQL**: un système qui permet de gérer des bases de données relationnelles. Il est écrit en langage Java.

**Hyper - V**: une solution de paravirtualisation qui peut être appliquée pour Microsoft.

**Hyperviseur**: une plateforme de virtualisation permettant à une multitude de systèmes d'exploitation de fonctionner simultanément dans la même machine.

**KVM** : est une forme de virtualisation complète qui permet d'assurer une virtualisation des processeurs.

**Libvirt** : une boîte à outils pour interagir avec les capacités de virtualisation de versions récentes de Linux (et d'autres systèmes d'exploitation).

**Maven**: un outil qui gère et automatise la production logiciels Java généralement et Java EE particulièrement.

**Memcache** : un système d'usage général servant à gérer la mémoire cache distribuée. Il est souvent utilisé pour augmenter la vitesse de réponse des sites web créés à partir de bases de données.

---

**mongoDB** : un système qui gère les bases de données orientées documents, répartissables sur un nombre quelconque d'ordinateurs et ne nécessitent pas un schéma de données préalablement défini.

**MySQL** : un système qui gère une base de données relationnelle (SGBDR).

**PostgreSQL** : un système qui gère une base de données relationnelle et objet.

**Proxy** : un composant logiciel informatique qui joue le rôle d'intermédiaire en se plaçant entre deux hôtes pour faciliter ou surveiller leurs échanges.

**QEMU** : un logiciel libre de machine virtuelle, pouvant émuler un processeur et, plus généralement, une architecture différente si besoin. Il permet d'exécuter un ou plusieurs systèmes d'exploitation via les hyperviseurs KVM et Xen, dans l'environnement d'un système d'exploitation déjà installé sur la machine.

**REDIS**: un système de stockage de donnée, il supporte les données qui ont une structure spécifique telle que : les listes, les chaînes de caractère, les ensembles...

**SQLite** : un système qui gère une base de données ayant la particularité de fonctionner sans serveur, on dit aussi "standalone" ou "base de données embarquée". On peut l'utiliser avec beaucoup de langages : PHP, Python, C# (.NET), Java, C/C++, Delphi, Ruby...

**SVN** : est un système de contrôle open source. Le SVN est similaire au Git mais le Git est beaucoup plus qu'un système de contrôle de version, il est considéré comme étant un workspace ou comme un système de gestion de contenu.

**Visual force** : un framework qui permet aux développeurs de créer des interfaces utilisateur personnalisées et sophistiquées qui peuvent être hébergées en mode natif sur la plate-forme Force.com.

**VMware vSphere** : un logiciel d'infrastructure de Cloud Computing de l'éditeur VMware.

**VMware** : une société informatique américaine qui développe divers logiciels propriétaires reliés à la virtualisation des architectures x86.

**XCP** : une version open source de Xen server qui est un produit commercial Citrix., il est bien sûr basé sur l'hyperviseur Xen. XCP est principalement écrit en langage OCaml, avec certaines parties également écrites en Python et C.

**Xen** : une solution open source de virtualisation qui permet de faire fonctionner une multitude de systèmes d'exploitation sur la même machine. Xen est basé sur le principe de la paravirtualisation, vu que le SE invité doit être modifié afin de cohabiter.

---

# Annexe C: Codes sources

## C.1. Code source de la méthode PROMETHEE II

Le code source de la méthode PROMETHEE II en langage Java pour le rangement des différents projets urbains est le suivant :

```
Publicclass promethee {
    /**
     * @param args
     */
    Static int c[];
    public String[] calss={"", "", "", "", ""};
    public promethee(int[] poids){
        c= poids;}

    publicvoid rangement() {
        int som1,som2;
        int fi[][]={{0,0},{0,0},{0,0},{0,0},{0,0}};
        int fipm[]={0,0,0,0,0};
        # mat1 Matrice des projets
        # mat1 Matrice des calculs

        for (int i=0;i<=4;i++){
for (int k = i+1; k <= 4; k++) {som1=0;som2=0;
            for (int j = 0; j <= 5; j++) {
                if (mat1[i][j]<mat1[k][j]){
                    som1+=0*c[j];
                    som2+=1*c[j];}
                else if (mat1[k][j]<mat1[i][j]){
                    som1+=1*c[j];
                    som2+=0*c[j];
                }
                else if (mat1[k][j]==mat1[i][j]){
                    som1+=0*c[j];
                    som2+=0*c[j];
                }
            }
            mat2[i][k]=som2;
            mat2[k][i]=som1;
        }
    }
    for (int i = 0; i <= 4; i++) {som1=0;som2=0;
        for (int j = 0; j <=4; j++) {
            if(i!=j){
                som1+=mat2[i][j];
            }
        }
        for (int k = 0; k <= 4; k++) {
            if(i!=k){
                som2+=mat2[k][i];
            }
        }
        fi[i][0]=som1;
        fi[i][1]=som2;
        fipm[i]=som2-som1;
    }
}
```

```

int[] trie={0,0,0,0,0};
for (int i = 0; i <=4; i++) {
    trie[i]=fipm[i];
}
int min;
for (int i = 0; i <= 4; i++) {
    for (int j = i+1; j <= 4; j++) {
        if (trie[i]<trie[j]){min=trie[i];trie[i]=trie[j];
        trie[j]=min;}
    }
}
for (int i = 0; i <=4; i++) {
    for (int j = 0; j <= 4; j++) {
if (fipm[j]==trie[i]){this.calss[i]="project"+(j+1);}
}
}
for (int i = 0; i <=4; i++) {
    System.out.println(this.calss[i]);
}
}

publicvoid setCalss(String[] calss) {this.calss = calss;
}
public String[] getCalss() {returncalss}}

```

## C.2. Code source de la méthode Hare

Le code source de la méthode Hare pour la négociation des projets urbains est le suivant :

```

import java.util.Random;
import java.util.Scanner;
import java.util.StringTokenizer;
import java.util.Vector;
public class hah {

    static Random randGen = new Random();
    static int m1;    static int m2;
    static int pos1;
    static int pos2;
    static String tok = "";
    staticchar ch;
    static Vector<Character> vec=new Vector();
    public static void main(String[] args) {
        int max=0;
        Vector minimum= new Vector();
        Scanner s = new Scanner(System.in);
        System.out.println("la méthode de hare");
        System.out.println();
        System.out.println("entrer le nombre de projets :");
        System.out.println("*****");
        int cand=s.nextInt();
        System.out.println("entrer le nombre de décideurs qui votent
        :");
        System.out.println("*****");
        int per=s.nextInt();
        char t_cand[]=newchar [cand] ;
    }
}

```

```

int t_comp[]=new int[cand];
char t_vote[][]=new char[cand][per];
System.out.println("entrer les noms des projets :");
System.out.println("*****");
for (int i=0; i<cand;i++) {
    t_cand[i]=s.next().charAt(0);
}
for (int i=0; i<cand;i++) {
    System.out.print(t_cand[i]+"");
}
System.out.println();
System.out.println("le vote :");
System.out.println("*****");
for (int c=0; c<cand;c++) {
    for (int p=0;p<per; p++) {
        t_vote[c][p]=s.next().charAt(0);
    }
}
//affichage de vote :
for (int c=0; c<cand;c++) {
    for (int p=0;p<per; p++) {
        System.out.print(t_vote[c][p]+"");
    }
    System.out.println();
}
//-----
//calculer l'occurrence:
String sss="";
for (int m=0;m<=cand-1;m++) {
    for (int k=0;k<cand;k++) {
        t_comp[k]=0;
    }
    for (int k=0;k<cand;k++) {
        for(int i=0; i<cand;i++) {
            for(int j=0; j<per;j++) {
                if(t_cand[k]==t_vote[i][j]) {
                    t_comp[k]=t_comp[k]+1;
                }
            }
        }
        break;
    }
}
//-----
int min1 = per;
//leminde vote:
for (int k=0;k<cand;k++) {
    if (min1>t_comp[k]) {
        min1=t_comp[k];
        ch=t_cand[k];
    }
}
for(int i=0;i<cand;i++) {
    if(min1==t_comp[i]&& !minimum.contains(t_cand[i])) {

        minimum.add(t_cand[i]);
    }
}
if(minimum.size()==1) {
    for(int pp=0;pp<per;pp++) {
        vec.removeAllElements();
    }
}

```

```

        for(int i=0; i<cand;i++){
            if (ch!=t_vote[i][pp]){
                vec.add(t_vote[i][pp]);
            }
        }
        sss=vec.toString();

        StringTokenizer token = new
        StringTokenizer(sss," , [ ]");
        while (token.hasMoreTokens ()) {
            tok = tok+token.nextToken();
        }
        //System.out.println(tok);
        for(int nn=0;nn<vec.size(); nn++){
            t_vote[nn][pp]=tok.charAt(nn);
        }
        tok="";
        sss="";
        for(int nn=0;nn<vec.size(); nn++){
            t_cand[nn]=vec.get(nn);
        }
    }
}
//affichage du resultat:
System.out.println("*-*-*-*-*-*-*-*-*-*-*-*-*-*-*");
for (int c=0; c<cand-1;c++){
    for (int p=0;p<per; p++){
        System.out.print(t_vote[c][p]+"");
    }
    System.out.println();
}
System.out.println();
minimum.removeAllElements();
cand--;
}else{
    if(minimum.size(>1){
        int var=minimum.size()-1;
        m2=randGen.nextInt(var);
        char car=minimum.get(m2).toString().charAt(0);
        for(int pp=0;pp<per;pp++){
            vec.removeAllElements();

            for(int i=0; i<cand;i++){
                if (car!=t_vote[i][pp]){
                    vec.add(t_vote[i][pp]);
                }
            }

            sss=vec.toString();
            StringTokenizer token = new
            StringTokenizer(sss," , [ ]");
            while (token.hasMoreTokens ()) {
                tok = tok+token.nextToken();
            }
            for(int nn=0;nn<vec.size(); nn++){
                t_vote[nn][pp]=tok.charAt(nn);
            }
            tok="";
            sss="";
            for(int nn=0;nn<vec.size(); nn++){
                t_cand[nn]=vec.get(nn);
            }
        }
    }
}

```

```

        }
    }
    //affichage resultat:
    System.out.println("*-*-*-*-*-*-*-*-*-*-*");
    for (int c=0; c<cand-1;c++){
        for (int p=0;p<per; p++){
            System.out.print(t_vote[c][p]+"");
        }
        System.out.println();
    }
    cand--;
    minimum.removeElementAt(m2);
}
}

//affichage du resultat:
System.out.println("matrice finale");
for (int c=0; c<2;c++){
    for (int p=0;p<per; p++){
        System.out.print(t_vote[c][p]+"");
    }
    System.out.println();
}
for (int k=0;k<cand;k++){
    t_comp[k]=0;
}
for (int k=0;k<cand;k++){
    for(int j=0; j<per;j++){
        if(t_cand[k]==t_vote[0][j]){
            t_comp[k]=t_comp[k]+1;
        }
    }
}
for (int k=0;k<cand;k++){
    if (max<t_comp[k]){
        max=t_comp[k];
        System.out.println("le choix sociale est:
        "+t_cand[k]+" ==> "+t_comp[k]);
    }
}
}
}}

```