|  |  |  |
| --- | --- | --- |
| **Université Badji Mokhtar-Annaba**  Faculté des Sciences de l’Ingeniorat  Département d’Informatique |  | **جامعة باجي مختار – عنابـــــــــــــــة**  كلية علــــــــــــوم الهندســـــــــة  قسم الإعــــــــــــــلام الآلــــــــــــــي |

Année 2017-2018

Année 2008-2009

**Mémoire présenté en vue**

**de l’obtention du diplôme de Master**

|  |
| --- |
| **Étude comparative de construction d'une application distribuée avec RMI et JCSP net.)** |

**Domaine :** Mathématiques, Informatique et Applications aux Sciences

**Filière :** Informatique

**Spécialité :** Ingénierie des Logiciels Complexes

**Par : Melle/Mr Rahim soufiane**

**Jury d’évaluation**

|  |  |  |  |
| --- | --- | --- | --- |
| **Qualité** | **Nom et Prénom** | **Grade** | **Université** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Remerciements

J’ai l’honneur de remercié le dieu qui m’a aider pour contenue mes études et qui m adonner les aptitudes pour faire les efforts, et je ne remercier pas Allah si je remercié pas mes enseignants et de plus prés Mr BenOuhiba mon encadreur, qui m’a aider beaucoup aux cours de ma formation et au cours de ma préparation de mémoire, et sans oublier touts l’équipe de ILC qui est une équipe très sérieux, et c’est sa qui ma encourager a contenue après avoir la licence

Et je remercier l’équipe de jure pour accepter à évaluer mon travaille.

Et sans oublier ma famille et mes amés qui ont encouragé pour contenue mes étude

Et sans oublier mes collèges qui j’ai passé des bons moments avec.

Dédicaces

Les plus brèves possibles

Table des Matières

A générer automatiquement

Table des Illustrations

A générer automatiquement

Table des Programmes

A générer automatiquement (Cette table est facultative)

Introduction

Contexte de recherche

\*\*\* Introduction au sujet traité en allant du plus général au plus spécifique (forme d’entonnoir)

\*\*\* Si le sujet s’étale sur plusieurs domaines, il est important de faire le lien de telle sorte à ce qu’un spécialiste d’un domaine ne se sente pas écarté

De nous jour les systèmes distribués en une grande importance dans les solutions informatiques

Et la théorie est bien spécifier mais il reste l’aspect technique qui se traduit par l’utilisation des machine virtuelle et les middlewares et les plateformes sophistiquées, et tout sa pour facilter l’implimentation des solutions,

Et pour ce la les grandes firmes de l’informatique comme Microsoft et ibm et d’autre en mis en point des technique d’échange des donne comme DCOM et dont de Microsoft et RMI de Sun microsystème et CORBA et d’autre,

Et en remarque l’aspect distribuée des applications qui est très rependue, et en pose la question

Problématique

\*\*\* Poser explicitement le problème

\*\*\* Dire en quoi les approches actuelles ne sont pas satisfaisantes face à ce problème

Faces a ces avance technologique très dévers et très avantageuse l une par a port a l autre les concepteurs et les architectes des systèmes trouvent de deféculter a tranche dans l’apport d’une solution par a port a l’autre,

Parmi les problèmes de comparaisons qui sont apparue en cite la comparaison entre RMI et JCSP.net qui sont des techniques de communication entre des applications distantes qui s’exécutent sur des machines distants

Motivations

\*\*\* Citer les facteurs motivants qui nous poussent à chercher une solution à la problématique

C est pour ca que les comparaisons entre ses système trouve leur impotence, par ce que la comparaison c’est un retour d’expérience qui raccourcie le chemin face au problème et donne une vision très claire et rapide et économe les efforts et diminue le redéveloppement des applications

Objectifs

\*\*\*\* La problématique dégagée peut être très importante, la section des objectifs sera utilisée pour délimiter le champ d’investigation

Dans ce travaille en vas comparer les deux méthodes de communication de point de vues conceptuelle et de point de vus implémentation et de tout autre point techniques qui vas nous attire pour prendre une vision claire sur l’apporte de chaque technique

Contenu du mémoire

\*\*\* Décrire le contenu de chaque chapitre

\*\*\* Dans le cas où le mémoire est volumineux, il est possible de proposer suite au contenu un plan de lecture

Pour exprimes les aspectes de ce travailler je consacre en premier lieu un chapitre sur la présentation de RMI

En suit je présenterai un chapitre sur csp qui est une version local de jcspnet et ensuit je présenterai un JCSP net

Apres la deuxième partie c’est la partie pratique consiste à faire la conception d’une application distribuée intituler MiniIsante qui consiste à échanger les informations médical des patients entre les médecines et ca avec en prendre en considération les deux technique RMI et JCSP.net

Et après en vas parler sur l’aspect implémentation de cette application avec les deux technique RMI et JCSP.net

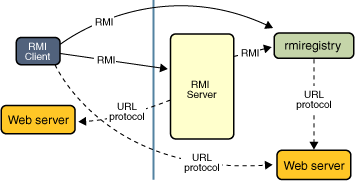
Chapitre 1   
Présentation de RMI

1.1 Introduction

La nature dynamique de la Platform java été étendu par le technique RMI qui permet d’appel des méthodes d’un objet distant comme si il est dans la même machine virtuelle local ;

Ce technique nous a permet de localiser et communique et charger la définition des objets distant, si objet distant sont héberger dans un composent qui s’appel registre ce registre envoi au client demandeur un stub qui est un proxy reprisent l’objet distant, il est a note que tout cette transparence vis-à-vis de programmeur est garante par un simple de serveur web

Ce mécanisme de communication et représentée dans le cheman suivant :



Il est a noté aussi que le serveur aussi peut télécharger la définition des classe a partir de client, comme par exemple les paramètres d »appel de méthodes distant si ces dernier sont des objets,

Et pour comprendre ce mécanisme de communication en va présenter en vas etuliser un exemple de code basique qui reprisent l’appel d une méthode d un objet distant par un client :

Création de serveur RMI avec le registre :

Cette étape est très simple a partir de version 6 de java par ce que la définition de stub est transparent de point de vue de programmeur, elle est gérer par la Platform java,

Cependant le développeur n’a que faire la conception des objets à héberger et sa se fait par une conception d’une simple interface java qui représente les méthodes qui les clients peut appeler a distant cette classe a deux règle simple

1. Il faut qu’il étendre la clase Remot
2. Il faut ajouter trow remotExeption dans chaque méthode distante

Une interface qui représente une seul méthode a le squelette suivant :

public interface MonIterface extends Remote {

TypedeRetour NomMetode(TypeParametre t) throws RemoteException;

}

L’implémentation de cette classe par un objet, cette opération a aussi des règles :

1. Bien sur il faut implémenter toutes les méthodes de cette interface et,
2. Une classe distante doit ´étendre la classe java.rmi.server.UnicastRemoteObject [[1]](#footnote-2)(d’autres possibilités´es existent) [[2]](#footnote-3)
3. Définir un constricteur

Cette implémentation a la forme suivante

public class nomObjetDestant implements NomInterface {

public nomObjetDestant () {

super();

}

public TypedeRetour executeTask(TypeParametre t) {

// Traitement a faire

}

}

Apres il faut créer un programme java qui démarre le service de ce objet :

Ce programme a la forme suivant

public class Lanceur {

public static void main(String[] args) {

if (System.getSecurityManager() == null) {

System.setSecurityManager(new SecurityManager());

}

try {

nomInterface nomObjet = new nomObjetDestant();

nomInterface stub =(nomInterface) UnicastRemoteObject.exportObject(nomObjet, 0);

Registry registry = LocateRegistry.getRegistry();

registry.rebind("nomSercvice", stub);

} catch (Exception e) {

System.err.println("ComputeEngine exception:");

e.printStackTrace();

}

}

}

Explication :

La première instruction

if (System.getSecurityManager() == null) {

System.setSecurityManager(new SecurityManager());

}

Il permet d’installer un manager de sécurité java par ce que en a affaire avec un téléchargement de définition des classe et sa doit être contrôler par des règles sur mesure, cette point va être présenté après dans la compilation et lancement de l’application.

Apres en a l’inspection suivante:

nomInterface nomObjet = new nomObjetDestant();

Cette instruction est une instruction standard il représente la créatine d’un référence local de l’objet serveur

Après en a :

nomInterface stub =

(nomInterface) UnicastRemoteObject.exportObject(nomObjet, 0);

Cette instruction transforme cette référence local en une référence qui pet être hibargé dans un registre RMI

Il est a note que ce 0 et un enter si il est 0 c a d que le porte de l’écoute de ce service, est définie soit par le système d’exploitation ou il est par défaut, et si le programmeur veut spécifier le port il doit changer le 0 par un nombre entier qui représente le porte² représente soit

Registry registry = LocateRegistry.getRegistry();

Cette instruction lance un registre RMI pour héberger l’objet destinent qui vient d’être créé,

Apres en l’instruction :

registry.rebind("nomSercvice", stub);

Cette instruction héberge l’objet dentant dan le registre

Et tout sa doit être envelopper par un try catche, par ce que ces instruction peuvent génère une exception par exemple échec d ouvrir un registre dans ou errer de hébergement de l’objet distant… etc.

Il faut préciser que  bind, unbind, ou rebind peuvent être appelé pour un registre qui se trouve dans la même machine de l’application serveur par contre l’instruction lookup que en vas utiliser dans le client pour récupérer une référence de l’objet distant peut être appelé pour une application distant et la raison de cette restriction et la sécurité

Après en vas créer un client qui vas demander les services de ce ou ses objets distant :

La création d’un client a la forme suivant :

public class ComputePi {

public static void main(String args[]) {

if (System.getSecurityManager() == null) {

System.setSecurityManager(new SecurityManager());

}

try {

Registry registry = LocateRegistry.getRegistry(args[0]);

nomInterface comp = (nomInterface) registry.lookup("Compute");

TypeRetour valeur = comp.NomMethode();

System.out.println(valeur);

} catch (Exception e) {

System.err.println("ComputePi exception:");

e.printStackTrace();

}

}

}

C paragramme et très simple et basique il créé une référence au registre qui se trouve dans une machine qui est déterminée par l’argument 0 du programme et el interroge ce registre pour récupérer une référence de l’objet distant qui est représenter par un stub transparent dans les versions antérieur de la version 5 de java

Apres il doit invoquer une méthode de ce objet

Il est à noter qu’il faut créer une interface identique à l’interface de server pour récupérer la référence de l’objet distant

Une autre remarque très important que dans le cas d’utilisation des objets comme paramètre d un appel de la méthode distant ou que ses objet sont un type de retour dans une méthode distant il faut que ses type soient des type primitif ou des clase distant ou qualifier d’être serialisable

El est à noter que les constricteurs obligatoires ans l’implémentation de l’interface distant appel un contracture de stub qui est une tache transparent a partir de la version 6 de JDK ce stub contient un référence d l’objet distant et une méthode invok() qui a la tache d’appel de méthode distant[[3]](#footnote-4)

Dans le site de oracle vous trouvez un exemple très puisant ,Ce exemple et le serveur de calcule distribuée , il est très utile dans scénario, si en a une puissante machine qui peut exécuter un code arbitraire fourni par le cliente et transmettre le résultat au client, car il nous a permette de voire le cas de transmettre des informations de client ver le serveur (par les paramètres d’appel de méthode qui représente un classe contiens un méthode arbitraire a exécuter par le serveur) et de serveur au client (comme est le cas dans tout les application qui utilisent RMI c a d par le renvoi de définition de l’objet distant et le résultat après l’exécution).

Il est à noter que la gestion de concurrence est la tache de développeur et sa se faite par synchronaize et wait etc. et sa se peut introduire un inter-blocage dans le code par exemple

Note sur la secureter dans le cas d’etulisation d un gestionnaire de securite dans le code de client ou de serveur il faut créer un fichier de confeguration de securite , et voici un exemple de deux fichier de securite qui donne toutles permission a l’execution dans un repertoire donnee :

Pour le serveur un ficher appeler server.policy

grant codeBase "file:/home/ann/src/" {

permission java.security.AllPermission;

};

Pour le client un fichier appeler client.polycie

grant codeBase "file:/home/jones/src/" {

permission java.security.AllPermission;

};

Conclusion :

Comme vous voyiez il n ya ni socket ni porte malgré que en peut paramétrer les portes mais par défaut tout marche bien et le développeur va concentrer sur la conception de la couche métier et il a l’avantage de développer une application distribuée comme si elle est une seule application

--------------------

## Starting the Server

Before starting the compute engine, you need to start the RMI registry. The RMI registry is a simple server-side bootstrap naming facility that enables remote clients to obtain a reference to an initial remote object. It can be started with the rmiregistry command. Before you execute rmiregistry, you must make sure that the shell or window in which you will run rmiregistry either has no CLASSPATH environment variable set or has a CLASSPATH environment variable that does not include the path to any classes that you want downloaded to clients of your remote objects.

To start the registry on the server, execute the rmiregistry command. This command produces no output and is typically run in the background. For this example, the registry is started on the host mycomputer.

**Microsoft Windows** (use javaw if start is not available):

start rmiregistry

**Solaris OS or Linux**:

rmiregistry &

By default, the registry runs on port 1099. To start the registry on a different port, specify the port number on the command line. Do not forget to unset your CLASSPATH environment variable.

**Microsoft Windows**:

start rmiregistry 2001

**Solaris OS or Linux**:

rmiregistry 2001 &

Once the registry is started, you can start the server. You need to make sure that both the compute.jar file and the remote object implementation class are in your class path. When you start the compute engine, you need to specify, using the java.rmi.server.codebase property, where the server's classes are network accessible. In this example, the server-side classes to be made available for downloading are the Compute and Task interfaces, which are available in the compute.jar file in the public\_html\classes directory of user ann. The compute engine server is started on the host mycomputer, the same host on which the registry was started.

**Microsoft Windows**:

java -cp c:\home\ann\src;c:\home\ann\public\_html\classes\compute.jar

-Djava.rmi.server.codebase=file:/c:/home/ann/public\_html/classes/compute.jar

-Djava.rmi.server.hostname=mycomputer.example.com

-Djava.security.policy=server.policy

engine.ComputeEngine

**Solaris OS or Linux**:

java -cp /home/ann/src:/home/ann/public\_html/classes/compute.jar

-Djava.rmi.server.codebase=http://mycomputer/~ann/classes/compute.jar

-Djava.rmi.server.hostname=mycomputer.example.com

-Djava.security.policy=server.policy

engine.ComputeEngine

The above java command defines the following system properties:

* The java.rmi.server.codebase property specifies the location, a codebase URL, from which the definitions for classes originating *from* this server can be downloaded. If the codebase specifies a directory hierarchy (as opposed to a JAR file), you must include a trailing slash at the end of the codebase URL.
* The java.rmi.server.hostname property specifies the host name or address to put in the stubs for remote objects exported in this Java virtual machine. This value is the host name or address used by clients when they attempt to communicate remote method invocations. By default, the RMI implementation uses the server's IP address as indicated by the java.net.InetAddress.getLocalHost API. However, sometimes, this address is not appropriate for all clients and a fully qualified host name would be more effective. To ensure that RMI uses a host name (or IP address) for the server that is routable from all potential clients, set the java.rmi.server.hostname property.
* The java.security.policy property is used to specify the policy file that contains the permissions you intend to grant.

## Starting the Client

Once the registry and the compute engine are running, you can start the client, specifying the following:

* The location where the client serves its classes (the Pi class) by using the java.rmi.server.codebase property
* The java.security.policy property, which is used to specify the security policy file that contains the permissions you intend to grant to various pieces of code
* As command-line arguments, the host name of the server (so that the client knows where to locate the Compute remote object) and the number of decimal places to use in the the pi symbolcalculation

Start the client on another host (a host named mysecondcomputer, for example) as follows:

**Microsoft Windows**:

java -cp c:\home\jones\src;c:\home\jones\public\_html\classes\compute.jar

-Djava.rmi.server.codebase=file:/c:/home/jones/public\_html/classes/

-Djava.security.policy=client.policy

client.ComputePi mycomputer.example.com 45

**Solaris OS or Linux**:

java -cp /home/jones/src:/home/jones/public\_html/classes/compute.jar

-Djava.rmi.server.codebase=http://mysecondcomputer/~jones/classes/

-Djava.security.policy=client.policy

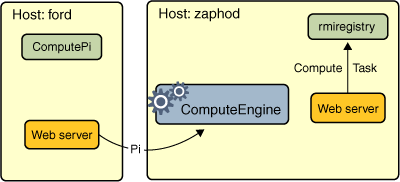
client.ComputePi mycomputer.example.com 45

Note that the class path is set on the command line so that the interpreter can find the client classes and the JAR file containing the interfaces. Also note that the value of the java.rmi.server.codebase property, which specifies a directory hierarchy, ends with a trailing slash.

After you start the client, the following output is displayed:

3.141592653589793238462643383279502884197169399

The following figure illustrates where the rmiregistry, the ComputeEngine server, and the ComputePi client obtain classes during program execution.



When the ComputeEngine server binds its remote object reference in the registry, the registry downloads the Compute and Task interfaces on which the stub class depends. These classes are downloaded from either the ComputeEngine server's web server or file system, depending on the type of codebase URL used when starting the server.

Because the ComputePi client has both the Compute and the Task interfaces available in its class path, it loads their definitions from its class path, not from the server's codebase.

Finally, the Pi class is loaded into the ComputeEngine server's Java virtual machine when the Pi object is passed in the executeTask remote call to the ComputeEngine object. The Pi class is loaded by the server from either the client's web server or file system, depending on the type of codebase URL used when starting the client.

1.4 Conclusion

\*\*\*\* Rappeler ce qu’on a présenté avec synthèse

\*\*\*\* Donner une conclusion personnelle

\*\*\*\* Faire l’enchaînement avec le chapitre suivant

Chapitre 2  
La séparation des préoccupations

Résumé sur 5 à 6 lignes

2.1 Principe

2.2 Les approches actuelles

2.3 La séparation des préoccupations dans le cycle de vie d’un logiciel

2.4 Conclusion

\*\*\*\* Rappeler ce qu’on a présenté avec synthèse

\*\*\*\* une conclusion personnelle

\*\*\*\* Faire l’enchaînement avec le chapitre suivant

Chapitre 3  
Composants et Séparation des Préoccupation : Une Synthèse des Approches de Combinaison

Résume sur 5 à 6 lignes

3.1 A la recherche d’une synergie

\*\*\*\* Les deux approches font des séparations

\*\*\*\* Les deux approches apportent des plus

\*\*\*\* Avantages de la combinaison

3.2 Les approches actuelles

\*\*\*\* Les classes et leurs caractéristiques

\*\*\*\* Les travaux représentatifs des classes (description)

3.3 Comparaison des approches

\*\*\*\* avantage, inconvénients des diverses approches

\*\*\*\* Tableau comparatif

Conclusion

\*\*\*\* Rappeler ce qu’on a présenté avec synthèse

\*\*\*\* une conclusion personnelle

\*\*\*\* Faire l’enchaînement avec le chapitre suivant

Chapitre 4  
Vers un approche de combinaison qui …

Résumé sur 5 à 6 lignes

4.1 Critique des approches actuelles de combinaison

\*\*\*\* Les insuffisances

4.2 Notre proposition

\*\*\* Caractéristiques

\*\*\* Principe

4.3 Exemple d’utilisation

\*\*\*\* une petite application exemple

4.4 Comparaison avec les approches existantes

\*\*\*\* Comment notre approche comble les insuffisances citées en 3.1

Conclusion

\*\*\*\* Rappeler ce qu’on a présenté avec synthèse

\*\*\*\* Une conclusion brève et optimiste

\*\*\*\* Faire l’enchaînement avec le chapitre suivant

Chapitre 5  
Aspects Pratiques

Résumé sur 5 à 6 lignes

5.1 Mise en œuvre de notre approche

5.2 Proposition d’outils support

\*\*\* Si un environnement a été proposé, décrire son architecture fonctionnelle et opérationnelle.

5.3 Les outils réalisés

\*\*\* Les environnements utilisés

Conclusion

\*\*\*\* Rappeler ce qu’on a présenté avec synthèse

\*\*\*\* Présenter une brève conclusion après ce chapitre

Conclusion et Perspectives

Rappeler le sujet et la problématique en allant du plus général au plus spécifique. Insister sur la difficulté du sujet traité.

Exemple : Depuis quelques années, la sécurité des systèmes informatiques est devenue une préoccupation majeure de la communauté des chercheurs en systèmes distribués et réseaux. L’un des points importants dans les solutions proposées actuellement est qu’elles conduisent à un enchevêtrement au niveau du code. La séparation avancée des préoccupations …

Résumé des contributions

Citer clairement chaque contribution du travail, comme si le lecteur avait lu le mémoire depuis 3 mois et il n’en garde qu’un vague souvenir.

Exemple : Nos investigations dans le domaine de la séparation avancée des préoccupations, et particulièrement celui de la représentation des points de jointures, nous ont permit d’apporter des contributions substantielles que nous partageons en trois catégories. Dans la première catégorie, nous nous sommes intéressé au modèle de points de jointure, où nous avons proposé un nouveau modèle dans l’utilisation est largement intuitive. En effet, concernant les prédicats, nous avons opté pour …, concernant la portée des points de jointure, nous avons, ….

Dans la second catégorie, nous avons situé les problèmes liés à la capture des points de jointure et nous un avons proposé un nouveau mécanisme dont nous avons démontré la performance.

…

Perspectives de recherche

Citer les perspectives qui complèteraient ce travail. Noter que l’ensemble des perspectives ne devrait pas être supérieur au travail lui-même. Si de nouvelles voies de recherche se dégagent, il faut les présenter en tant qu’issues de recherche et non comme des perspectives de ce travail.

Eviter les perspectives du type : nous avons proposé X et il nous reste maintenant à proposer Y, avec Y = 10\*l’effort consenti pour X.

Opter plutôt pour l’inverse.

Bibliographie

Les lignes du tableau ne doivent pas apparaître

Exemples de référence d’article de journal

|  |  |
| --- | --- |
| [ABU 94] | Abuhaiba I.S.I., Mahmoud S.A., Green R.J., "Recognition of handwritten cursive Arabic characters", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 16, pp. 664-672, 1994. |
|  | Cas de références ayant le même code, rajouter a et b pour les différencier |
|  |  |
|  |  |
|  |  |
| [ALM 02a]  C’est la référence qui doit obligatoirement apparaître dans le texte des chapitres, les références sont classées par ordre alphabétique | Al-Ma'adeed S., Higgens C., Elliman D., A database for Arabic handwritten text recognition research, Proceedings of the Eighth IWFHR, Ontario, Canada, pp. 485-489, 2002. |
| [ALM 02b] | Al-Ma'adeed S., Higgens C., Elliman D., "Recognition of Off-Line Handwritten Arabic Words Using Hidden Markov Model Approach", ICPR 2002, Quebec City, August, 2002. |
| [ALM 04a] | Al-Ma’adeed S., Elliman D., Higgins C., "A Data-Base for Arabic Handwritten Text Recognition Research", The International Arab Journal of Information Technology (IAJIT), Vol. 1, No. 1, pp. 118-123, 2004. |
| [ALM 04b] | Al-Ma’adeed S., Elliman D., Higgins C., "Off-line recognition of handwritten Arabic words using multiple hidden Markov models", Knowledge-Based Systems, Vol. 17, pp. 75-79, 2004. |
| [ALO 00] | Al-Ohali Y., Cheriet M., Suen C., "Databases for recogntion of handwritten Arabic cheques",7th International Workshop on Frontiers in Handwriting Recognition, Amsterdam, The Netherlands, IWFHR’2000, pp. 601-606, September 2000. |
| [ALO 02] | Al-Ohali Y., "Handwritten Word Recognition – Application to Arabic Cheque Processing", PhD Thesis, Concordia University, Montreal, Canada, 2002. |
| [ALO 03] | Al-Ohali Y., Cheriet M., Suen C., "Databases for recognition of handwritten Arabic cheques", Pattern Recognition Vol. 36, pp. 111-121, 2003. |
| [ALP 97] | Alper Atici A., Yarman-Vural F. T., "A heuristic algorithm for optical character recognition of Arabic script", Signal Processing, Vol. 62, pp. 87-99, 1997. |
| [ALY 92] | Al-Yousefi H., Udpa S.S., "Recognition of Arabic characters", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, pp. 853-857, Aug 1992. |
| [AME 94]  Exemple de Thèse | Ameur A., Romeo-Pakker K., Miled H., Cheriet M., "Approche globale pour la reconnaissance de mots manuscrits Arabes", Actes CNED’94, 3ème Colloque National sur l’Ecrit et le Document, pp. 151-156, Juillet 1994. |
| [AMI 82] | Amin A., "Machine recognition of hand written arabic words by the IRAC II system", Proceedings ICPR’82, 6th International Conference on Pattern Recognition, Vol. 1, pp. 34-36, Oct 1982. |
| [AMI 85] | Amin A., Masini G., "Deux méthodes de reconnaissance de mots pour l’écriture arabe manuscrite", Actes RFIA’85, 5 ème congrès Reconnaissance des Formes et Intelligence Artificielle, Tome 2, pp. 837-848, 1985. |
| [AMI 89] | Amin A., Mari J. F., "Machine recognition and correction of printed arabic text", IEEE Transactions on systems, man and cybernetics, Vol. 19, No. 5, pp. 1300-1306, Sept/ Oct 1989. |
| [AMI 96] | Amin A., Al-Sadoun H., Fischer S., "Hand-printed arabic character recognition system using an artificial network", Pattern Recognition, Vol. 29, No. 4, pp. 663-675, 1996. |
| [AMI 97a] | Amin A., "Recognition of printed and handwritten arabic characters", Lecture Notes in Computer Science, No. 1339, pp. 40-59, Advances in Document Image Analysis, Proceedings of First Brazilian Symposium, BSDIA’97, Curtiba, Brazil, November, 1997. |
| [AMI 97b] | Amin A., "Arabic character recognition", , in H. Bunke, P.S.P. Wang (Editors), Handbook of character recognition and document image analysis, World Scientific, pp. 397-420, 1997. |
| [AMI 97c] | Amin A., Mansoor W., "Recognition of printed arabic text using neural networks", Proceedings of ICDAR’97, 4th International Conference on Document Analysis and Recognition, Vol. 2, pp. 612-615, 1997. |
| [AMI 98] | Amin A., "Off-line Arabic character recognition : The state of the art", Pattern Recognition, Vol. 31, No. 5, pp 517-530, 1998. |
| [AMI 00a] | Amin A., "Recognition of printed Arabic text based on global features and decision tree learning techniques", Pattern Recognition, Vol. 33, pp. 1309-1323, 2000. |
| [AMI 00b] | Amin A., Murshed N., "Off-line recognition of printed Arabic words through global features and neural networks", 4th international workshop on Document Analysis Systems, DAS'2000, Rio de Janeiro, Brazil, pp. 267-277, 2000. |
| [AMI 03] | Amin A., "Recognition of hand-printed characters based on structural description and inductive logic programming", Pattern Recognition Letters, Vol. 24, pp. 3187-3196, 2003. |
| [ANQ 97] | Anquetil E., "Modélisation et reconnaissance par la logique floue :application à la lecture automatique en-ligne de l’écriture manuscrite omni-scripteur", Thèse de Doctorat, Université de Rennes 1, Institut de Formation Supérieure en Informatique et Communication (Sciences pour l’Ingénieur), 1997. |
| [AUG 00]  Exemple d’article de conférence ou de workshop | Augustin E., Price D., Baret O., "Le système de reconnaissance de documents A2IA Check ReaderTM", CIFED’2000, Colloque International Francophone sur l’Ecrit et le Document, pp. 241-250, Lyon, 2000. |
| [AVI 96] | Avila M., "Optimisation des modèles Markoviens pour la reconnaissance de l’écrit", Thèse de Doctorat, Université de Rouen, Décembre 1996. |
| [AZI 02a] | Azizi N., Sari T., Souici-Meslati L., Sellami. M., "Une architecture de combinaison floue de classifieurs neuronaux pour la reconnaissance de mots arabes manuscrits", CIFED’2002, Colloque International Francophone sur l’Ecrit et le Document, pp. 89-96, Hammamet, Tunisie, Octobre 2002. |
| [AZI 02b] | Azizi N., "Combinaison de classifieurs neuronaux basée sur la logique floue : application à la reconnaissance des mots arabes manuscrits", Mémoire de magister, Laboratoire LRI, Département d'informatique, Université d'Annaba, Algérie, 2002. |
| [BAD 82] | Badie K., Shimura M., "Machine recognition of Arabic handprinted scripts", The transactions of the IECE of Japan, Vol. E65, No. 2, February, 1982. |
| [BAZ 99] | Bazzi I., Schwartz R., Makhoul J., "An omnifont open-vocabulary OCR system for english and arabic", IEEE PAMI, Vol. 21, No. 6, pp. 495-504, June 1999. |
| [BEC 87] | Becker J. D., "Arabic word processing", Communications of the ACM, Vol. 30, pp. 600-610, July 1987. |
| [BED 95] | Bedda M., Ramdani M., Doghmane N., "Classification des caractères arabes manuscrits par un perceptron multi-couches ", Conférence méditeranéenne sur l’automatique MCEA, France, 1995. |
| [BEL 92] | Belaid A., Belaid Y., "Reconnaissance des formes: Méthodes et applications", InterEditions, 1992. |
| [BEL 97] | Belaid A., Saon G., "Utilisation des processus markoviens en reconnaissance de l'écriture", Traitement du signal, Vol. 14, No. 2, pp. 161-177, 1997. |
| [BEL 01] | Belaid A., " Reconnaissance automatique de l’écriture et du document", Pour la science, 2001. |
| [BEN 96] | Benyoub B., "Une application industrielle de reconnaissance d’adresses", Actes CNED’96, 4ème Colloque National sur l’Ecrit et le Document, pp. 93-100, Nantes, France, Juillet 1996. |
| [BEN 99] | Bennasri A., Zahour A., Taconet B., "Extraction des lignes d’un texte manuscrit arabe", Proceedings Vision Interface’99, Trois rivières, Québec, Canada, pp : 41-48, Mai 1999. |
| [BEN 00] | Benouareth A., "Reconnaissance de l'écriture arabe manuscrite par une approche hybride", Mémoire de magister, Labo. LRI, Département d'informatique, Université d'Annaba, Algérie, Février 2000. |
| [BIS 95] | Bishop C. M., "Neural networks for pattern recognition", Clarendon Press, Oxford, 1995. |
| [BOZ 89] | Bozinovic R. M., Srihari R. N., "Off-line cursive script word recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11, No. 1, pp. 68-83, Jan 1989. |
| [BRO 83] | Brown M. K., "Preprocessing techniques for cursive script word recognition", Pattern Recognition, Vol. 16, No. 5, pp. 447-458, 1983. |
| [BUN 97] | H. Bunke, P.S.P. Wang (Editors), "Handbook of character recognition and document image analysis", World Scientific, 1997. |
| [BUN 03] | BunkeH., "Recognition of Cursive Roman Handwriting – Past, Present and Future", International Conference on Document Analysis and Recognition, ICDAR, Edinburgh, Scotland, 2003. |
| [CAK 02] | Cakmakov D., Bennani Y., "Feature selection for pattern recognition", Skopje, Informa, 2002. |
| [CAS 96] | Casey R. G., Lecolinet E., "A survey of methods and strategies in character segmentation", IEEE Transactions on Pattern Analysis and Machine Intelligence,Vol. 18, No. 7, pp. 690-706, July 1996. |
| [CHE 94] | Chen M. Y., Kundu A., Zhou J., "Off-line handwritten word recognition using a hidden markov model type stochastic network", IEEE Transactions on PAMI, Vol. 16, No. 5, pp. 481-496, May 1994. |
| [CHE 98] | Cheriet M., Miled H., Olivier C., "Visual aspect of cursive Arabic handwriting recognition", Proceedings Vision Interface, VI'98, pp. 262-270, 1998. |
| [CHE 01] | Cheung A., Bennamoun M., Bergmann N.W., "An Arabic optical character recognition system using recognition-based segmentation", Pattern Recognition, Vol. 34, pp. 215-233, 2001. |
| [COR 02] | Cornuéjols A., Miclet A., Kodratoff Y., "Apprentissage artificiel: concepts et algorithmes", Eyrolles, 2002. |
| [COT 96a] | Côté M., Lecolinet E., Cheriet M., Suen Y.C., "Using Reading Models for Cursive Script Recognition", in Handwriting and Drawing Research: Basic and Applied Issues , M.L. Simner, C.G. Leedham and A.J.W.M. Thomassen Eds., pp. 299-313, IOS Press, Amsterdam, 1996. |
| [COT 96b] | Côté M., Lecolinet E., Cheriet M., Suen C.Y., "Lecture automatique d'écriture cursive utilisant des concepts perceptuels", Actes du congrès de l'Association canadienne-française pour l'avancement de la science, pp. 131-140, Montréal, Canada, Mai 1996. |
| [COT 97a] | Côté M., "Utilisation d'un modèle d'accès lexical et de concepts perceptifs pour la reconnaissance d'images de mots cursifs", Thèse de doctorat, Département signal, Ecole nationale supérieure des télécommunications ENST, Paris, France, Juin 1997. |
| [COT 97b] | Côté M., Cheriet M., Lecolinet E., Suen C.Y., "Automatic Reading of Cursive Scripts Using Human Knowledge", Int. Conf. On Document Analysis and Recognition (ICDAR), Ulm, Germany, Vol. 1, pp. 107-111, August 1997. |
| [COT 98] | Côté M., Lecolinet E., Cheriet M., Suen C.Y., "Automatic reading of cursive scripts using a reading model and perceptual concepts: The percepto system",  IJDAR, Vol. 1, pp. 3-17, 1998. |
| [DAV 93] | Davalo E., Naim P., "Des réseaux de neurones", Eyrolles, 1993. |
| [DAV 97] | Davidson R., Hopely R., "Arabic and persian OCR training and test data sets," in Proc. Of Symp. On Document Image Understanding Technology, 1997. |
| [DEH 01a] | Dehghan M., Faez K., Ahmadi M., Shridhar M., "Handwritten Farsi (Arabic) word recognition: a holistic approach using discrete HMM", Pattern Recognition, Vol. 34, pp. 1057-1065, 2001. |
| [DEH 01b] | Dehghan M., Faez K., Ahmadi M., Shridhar M., "Unconstrained Farsi handwritten word recognition using fuzzy vector quantization and hidden Markov models", Pattern Recognition letters, Vol. 22, pp. 209-214, 2001. |
| [DEN 97] | Dengel A., Hoch R., Hönes F., Jäger T., Malbuerg M., Weigel A., "Techniques for improving OCR results", in [BUN 97], Chapter 8, pp. 227-258, 1997. |
| [DIL 00] | Di Lecce V., Dimauro G., Guerriero A., Impedovo S., Pirlo G., Salzo A., "A New Hybrid Approach for Legal Amount Recognition ", 7th International Workshop on Frontiers in Handwriting Recognition, Amsterdam, The Netherlands, IWFHR’2000, pp. 199-208, September 2000. |
| [DIM 97] | Dimauro G., Impedevo S., Pirlo G., Salzo A., "Handwriting recognition: State of the art and future trends", Lecture Notes in Computer Science, No. 1339, pp. 1-18, Advances in Document Image Analysis, Proceedings of First Brazilian Symposium, BSDIA’97, Curtiba, Brazil, November, 1997. |
| [DOM 88] | Dominé C. H., "Techniques de l'intelligence artificielle, un guide structuré", Dunod Informatique, 1988. |
| [DRE 04] | Dreyfus G et al., "Réseaux de neurones, méthodologie et applications", Eyrolles, 2004. |
| [DUN 92] | Dunn C. E., Wang P. S. P., "Character segmentation techniques for handwritten text: a survey", Proceedings of ICPR’92, 11th International Conference on Pattern Recognition, pp. 577-580, 1992. |
| [DUP 04] | Dupré X., "Contributions à la reconnaissance de l'écriture cursive à l'aide de modèles de Markov cachés", Thèse de doctorat, Université René Descartes, Paris 5, Janvier 2004. |
| [ELD 90] | El-Dabi S. S., Ramsis R., Kamel A., "Arabic character recognition system: a statistical approach for recognizing cursive typewritten text", Pattern Recognition, Vol. 23, No. 5, pp. 485-495, 1990. |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| [TRI 96] | Trier O. D., Jain A. K., Taxt T., "Feature extraction methods for character recognition: a survey", Pattern Recognition, Vol. 29, No. 4, pp. 641-662, 1996. |
| [VIN 02] | Vinciarelli A., "A survey on off-line Cursive Word Recognition", Pattern Recognition, Vol. 35, pp. 1433–1446, 2002. |
| [WEB 96] | Weber V., Wermter S., "Using hybrid connectionist learning for speech/language analysis", In Wermter S., Rilo E., Scheler E. (editors), Connectionist, Statistical and Symbolic Approaches to Learning for Natural Language Processing, pp. 87-101, Springer-Verlag, 1996. |
| [WER 00] | Wermter S., Sun R., "Hybrid neural systems", Springer-Verlag, 2000. |
| [XU 92] | Xu L., Krzyzak A., Suen C. Y., "Methods of combining multiple classifiers and their application to handwriting recognition", IEEE transactions on systems, man and cybernetics, Vol. 22, No. 3, pp. 418-435, May /June 1992. |
| [XU 02] | Xu Q., Kim J. H., Lam L., Suen C. Y., "Recognition of handwritten month words on bank cheques", IWFHR’2002, 8th International Workshop on Frontiers in Handwriting Recognition, pp. 111-116, Niagara-on-the-Lake, Ontario, Canada, August 2002. |
| [ZAH 98] | A. Zahour, A. Djematene, S. Kebairi, A. Bennasri, B. Taconet, "Contribution à la reconnaissance de l’écriture manuscrite arabe", Actes CIFED’98, 1er Colloque International Francophone sur l’Ecrit et le Document, pp. 218-227, Québec, Canada, Mai 1998. |
| [ZOU 02] | Zouari H., Heutte L., Lecourtier Y., Alimi A., "Un panorama des méthodes de combinaison de classifieurs en reconnaissance de formes", RFIA'2002, 11ème Congrès francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle, pp. 499-508, Angers, France, Janvier 2002. |
| Sites Web | |
| [1] | http://www.a525g.com/intelligence-artificielle/algorithme-genetique.html |
| [2] |  |
| [3] |  |
| [4] |  |
| [5] |  |
| [6] |  |
| [7] |  |
| [8] |  |
| [9] |  |
| [10] | UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/ |

Annexe A

Mettre dans les annexes, par exemple, des informations sur des éléments importants pour comprendre votre travail, pouvant être des compléments théoriques ou des éléments pratiques que vous avez utilisés mais qui, s’ils sont mis dans les chapitres, risquent de les alourdir et de rendre leur lecture difficile.

Annexe B

***Résumé***

Le résumé ne doit pas excéder une douzaine de lignes. Il doit être écrit de sorte à permettre aux lecteurs de déterminer rapidement si le mémoire l’intéresse ou pas. Pour cela, il doit donner un bref aperçu sur le sujet, la problématique, la motivation qui pousse à la recherche de la solution ainsi que ce qui a été réalisé (aperçu de la solution proposée). Le résumé se termine par une note d’optimisme qui situe la solution par rapport à ce qui existe. Le résumé ne comporte pas de sous titre.

La page des résumés apparaît au verso de la couverture (carton) du mémoire

De 3 à cinq mots clés. Les mots clés doivent refléter le sujet traité. Les mots clés doivent être ordonnés alphabétiquement. Le rôle des mots clés est de permettre la sélection automatique de l’article dans une base de données. Si une personne travaille sur le même sujet, elle devrait pouvoir sélectionner ce mémoire par ses mots clés. Elle ne devrait pas être induite en erreur par des mots clés qui ne correspondent pas au fond du sujet traité

**Mots clés**. Plug-in, Eclipse, Composant logiciel,

**ملخص**

Le résumé en arabe est administrativement obligatoire. Utilisez SWEBOK version arabe, pour trouver la terminologie correspondant à celle en français.

مع مرور الزمن

**كلمات مرشدة.** برمجيات، فصل الانشغالات،

1. « 3\_rmi.pdf ». [↑](#footnote-ref-2)
2. Voir l’exemple de site oracle le moteur de calcule qui n’etendre pas la classe java.rmi.server.UnicastRemoteObject [↑](#footnote-ref-3)
3. « 3\_rmi.pdf ». [↑](#footnote-ref-4)