

RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE  
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE  
SCIENTIFIQUE

UNIVERSITÉ IBN KHALDOUN TIARET



Faculté des Mathématiques  
et de l'Informatique  
Département Informatique



## MÉMOIRE

Pour l'obtention du Diplôme de Master en Informatique  
Option : Génie Logiciel (GL)

**Présenté par :**

Mlle. Badra BENGUETIB

Mme. Khaldia DEMNI

---

# APPROCHE BASÉE SUR LA DÉTECTION DES COMMUNAUTÉS CACHÉES POUR L'AMÉLIORATION DU PROCESSUS DE DÉCOUVERTE DES SERVICES WEB

---

Mémoire dirigé par :

Mr. Hadj MEGHAZI

Maître assistant à l'université Ibn Khaldoun

Soutenu en 2016

Devant le jury composé de :

---

<i>Président</i> :	Mr. Omar TALBI	Université Ibn Khaldoun
<i>Examineur</i> :	Mr.	Université Ibn Khaldoun
<i>Encadreur</i> :	Mr. Hadj MEGHAZI	Université Ibn Khaldoun

---



## Résumé

Quand il s'agit des services web le problème le plus évident est toujours de trouver la meilleure approche pour permettre aux clients la découverte des services les plus pertinents pour une éventuelle requête. Dans cette perspective plusieurs approches basées sur les descriptions syntaxiques et sémantiques ont été proposées. Cependant, en prenant un peu de recul et en suivant la tendance, il sera plus judicieux de se focaliser sur les relations qui existent entre les services eux-mêmes. Ce sujet vise à proposer une nouvelle approche qui se base sur l'exploitation des particularités sociales qui existent entre les services web afin de mieux cibler les requêtes tout en espérant d'améliorer la qualité du processus de découverte.

**Mots clés :** Services web sociaux, Approche de découverte, Analyse des réseaux sociaux, LICOD.

## Abstract

When it comes to the web services the most obvious problem is always to find the best approach to enable customers to discover the most relevant service for any query. In this perspective several approaches based on syntactic and semantic descriptions have been proposed. However, taking a few steps back and following the trend, it will be wiser to focus on the relationships between the services themselves. The aim of this topic is to propose a new approach based on using social particularities between web services to better target requests, while hoping to improve the quality of the discovery process.

**Keywords :** Social web services, Discovery approach, Social network analysis, LICOD.

## Remerciements

Avant tout, nous remercions Allah soubhanahou wa ta'ala, et nos grand salut sur le premier éducateur notre prophète Mohamed que le salut soit sur lui.

En premier lieu nous tenons à exprimer nos reconnaissances à notre encadreur Mr. MEGHAZI Hadj Maître assistant à l'université de Tiaret pour l'attention qu'il nous a accordée durant la réalisation de ce modeste travail, pour sa disponibilité, les conseils précieux et fructueux qu'ils n'ont cessé de nous prodiguer. Merci Monsieur énormément pour le tout.

Nous sommes infiniment gré à Mr. TALBI Omar de s'être rendu disponible pour l'évaluation de ce mémoire. Nous vous devons nos profonds respects et nos particulières grâces. Ces remerciements seraient incomplets si nous n'en adressions pas à Mr. OUARED Abdelkader, Mr. MERATI Medjeded, Mme LEKHDARI Aicha pour leur soutien et leur encouragement pendant tout notre cursus universitaire.

Et un remerciement spécial à Mr. LABBACI Hamza pour son temps et ses conseils fructueux.

Enfin, à tous ceux qui, par un mot, m'ont donné la force de continuer.

# Dédicaces

Je voudrais dédier ce travail

## **A mes parents**

Aux plus belles créatures que dieu a créées sur terre, à cette source de tendresse, de patience et de générosité. Aucune dédicace ne pourrait exprimer mon respect, ma considération et mes chaleureux sentiments envers mes Chers Parents ; Grâce à leurs tendres encouragements et leurs grands sacrifices, ils ont pu créer le climat affectueux et propice à la poursuite de mes études. Je prie le bon lieu de les bénir, de veiller sur eux, en espérant qu'ils soient toujours fiers de moi.

## **A mes beaux frères**

Abdeslam et Abderrazak

## **A mes belles sœurs**

Kaouthar et Hadjer

## **A mes grands-parents**

## **A mes oncles, mes tantes et tous mes cousins**

## **A tous mes professeurs**

Leur générosité et leur soutien m'obligent de leurs témoigner mon profond respect et ma loyale considération.

## **À toutes mes amies**

## **A mon binôme Khalida**

## **À la promotion de GL 2016**

Et un spécial remerciement à Nejla qui m'aider tout le temps

Et à tous ceux que j'aime et à toutes les personnes qui m'ont prodiguée des encouragements et se sont données la peine de me soutenir durant cette formation

Badra

## Dédicaces

Je dédie ce modeste travail

**A chers parents**

pour leur patience, leur amour et leurs encouragements.

**A mon très cher mari**

ton soutien m'a permis de réussir mon travail. Ce travail soit témoignage de ma reconnaissance et de mon amour sincère et fidèle.

**A ma sœur et mon frère**

**A mon binôme Badra**

**A mes amies et mes camarades**

Khaldia

# Table des matières

Table des figures . . . . .	VI
Liste des tableaux . . . . .	VII
Introduction générale . . . . .	1
<b>1 Les services Web Sémantiques . . . . .</b>	<b>3</b>
1.1 Introduction . . . . .	4
1.2 Architecture Orientée Services . . . . .	4
1.3 Les services web . . . . .	5
1.3.1 Définition . . . . .	6
1.3.2 Caractéristiques des services web . . . . .	6
1.3.3 Fonctionnement des Services Web . . . . .	7
1.3.4 Standards utilisés pour le web service . . . . .	8
1.3.5 Conclusion . . . . .	11
1.4 Les services web sémantique . . . . .	12
1.4.1 Évolution vers le Web sémantique . . . . .	12
1.4.2 Définition des services web sémantique . . . . .	13
1.4.3 L'objectif des services web sémantique . . . . .	13
1.4.4 L'architecture des services web sémantiques . . . . .	14
1.4.5 Descriptions sémantiques des services web . . . . .	15
1.5 Conclusion . . . . .	18
<b>2 Les Services Web Sociaux . . . . .</b>	<b>19</b>
2.1 Introduction . . . . .	20
2.2 Services web en deçà de leur potentiel . . . . .	20
2.3 Pourquoi l'aspect social est utile ? . . . . .	21
2.4 Définition des services Web sociaux . . . . .	23
2.5 Les Services Web Sociaux en action . . . . .	23
2.6 Développement des services Web sociaux . . . . .	24

2.7	Conclusion . . . . .	30
<b>3</b>	<b>L'analyse Des Réseaux Sociaux (ARS) . . . . .</b>	<b>31</b>
3.1	Introduction . . . . .	32
3.2	Réseau social . . . . .	32
3.2.1	Représentation d'un réseau social . . . . .	32
3.2.2	Graphes : Notions et terminologies . . . . .	33
3.3	Analyse des réseaux sociaux . . . . .	35
3.4	Analyse et fouille de données sociales . . . . .	36
3.4.1	Fouille de données sociales . . . . .	36
3.5	Définition d'une communauté . . . . .	39
3.5.1	Définitions basées sur la connectivité des sommets . . . . .	40
3.5.2	Définitions basées sur la similarité des sommets . . . . .	41
3.5.3	Définitions basées sur une fonction de qualité . . . . .	42
3.6	Détection de communautés . . . . .	42
3.6.1	Les méthodes de classification hiérarchique . . . . .	42
3.6.2	Les méthodes d'optimisation d'une fonction objective . . . . .	47
3.6.3	Les méthodes à base du modèle . . . . .	48
3.6.4	Les méthodes centrées graines . . . . .	49
3.7	Conclusion . . . . .	50
<b>4</b>	<b>L'approche adoptée . . . . .</b>	<b>51</b>
4.1	Introduction . . . . .	52
4.2	L'approche Adoptée . . . . .	52
4.3	Description de l'approche . . . . .	52
4.3.1	Identification des leaders . . . . .	54
4.3.2	Regroupement des leaders . . . . .	54
4.3.3	Calcul de vecteur d'appartenance . . . . .	54
4.3.4	Fusion de votes . . . . .	54
4.3.5	Paramètres . . . . .	56
4.4	Implémentation . . . . .	56
4.5	Benchmark utilisé . . . . .	57
	<b>Conclusion générale . . . . .</b>	<b>58</b>
	<b>Annexes . . . . .</b>	<b>59</b>
.1	Annexe A : Mesures locales . . . . .	59
.2	Annexe B : Mesures globales . . . . .	61
.3	Annexe C : Modularités . . . . .	63

.4 Annexe D : Fiche technique sur Gephi . . . . . 65

**Bibliographie . . . . . 70**

# Table des figures

1.1	Acteurs et interaction dans l'architecture orientée service . . . . .	5
1.2	Fonctionnement des services web . . . . .	8
1.3	Exemple de fichier WSDL . . . . .	10
1.4	l'Evolution du web . . . . .	13
1.5	Pile des services web sémantique . . . . .	14
1.6	Processus d'annotation sémantique . . . . .	16
1.7	Model conceptuel d'OWL-S . . . . .	16
2.1	L'expérience de l'utilisateur 1 . . . . .	21
2.2	L'expérience de l'utilisateur 2 . . . . .	22
2.3	Représentation générale de la méthode de conception des SWS . . . . .	25
2.4	Classification des SWS selon la fonctionnalité . . . . .	26
2.5	Illustration d'un réseau social de compétition . . . . .	27
3.1	Un exemple d'un sociogramme. . . . .	32
3.2	Exemple de représentation matricielle du graphe . . . . .	33
3.3	Analyse structurelle basée seulement sur les relations entre les nœuds . . . . .	35
3.4	Exemple d'un graphe contenant deux cliques ( $C1$ et $C3$ ) . . . . .	40
3.5	Indice de Jaccard . . . . .	41
3.6	Exemple de dendrogramme . . . . .	43
3.7	Exemple de réseau à 16 sommets divisé en 2 communautés . . . . .	44
3.8	Mesures de centralité d'intermédiarité d'un réseau. . . . .	45
3.9	Dendrogramme du réseau de Zachary . . . . .	46
3.10	Exemple d'application de la méthode de Louvain sur un graphe à 16 sommets . . . . .	48
3.11	Les différentes structures de communautés trouvées par Label Propaga- tion pour le réseau du club de karaté. . . . .	49

4.1	Algorithme de LICOD . . . . .	53
4.2	Vecteurs d'appartenance des nœuds avec un exemple de 4 communautés	55
4.3	Gephi logo . . . . .	56
4	Structure de communautés . . . . .	63
5	Exemple d'un graphe jouet . . . . .	64
6	L'onglet laboratoire de donnée . . . . .	65
7	L'onglet Vue d'ensemble . . . . .	65
8	L'onglet prévisualisation . . . . .	66

# Liste des tableaux

3.1	Les indicateurs de centralités . . . . .	38
4.1	Les caractéristiques topologiques de réseau Celegans neural . . . . .	57



---

# Introduction générale

L'architecture orientée service (SOA) et sa vedette technologie de mise en œuvre connue sous le nom des services Web ont changé la façon dont les ingénieurs conçoivent et développent les applications d'entreprise d'aujourd'hui. Les services Web aident les organisations à maintenir une présence active sur Internet. Agissant comme des blocs de construction qui peuvent fournir et transformer les données, ils se connectent ensemble pour créer de nouveaux services sur demande à valeur ajoutée composite. Bien que les utilisateurs de SOA défendent régulièrement pour ses avantages, l'état actuel de SOA n'a pas vraiment maintenu ces avantages : les applications actuelles de SOA sont principalement conçues pour des environnements fermés, statiques lors de l'exécution, et reposent principalement sur des caractéristiques et des méthodes formelles. Comme il n'est pas évident pour les services web de découvrir et localiser le meilleur service. Donc comment peut on supprimer ces limitations ?

Dans les approches existantes (syntaxique ou sémantique) les utilisateurs et les services effectuent et font seulement confiance des services et ce sans tenir compte des choix de leurs pairs ainsi de leurs interactions passées. La solution proposée est d'adapter une approche sociale qui permet de mettre en place une certaine mémoire pour le système que nous allons l'exploiter pour détecter d'éventuelles communautés cachées afin de mieux cibler les requêtes et de gagner en performance.

Le présent mémoire est structuré en quatre chapitres. Dans le premier chapitre, intitulé **Services Web Sémantique**, qui se compose de deux parties, dans la première, nous présentons les services web : quelques définitions, caractéristique et fonctionnement des services web ainsi que les standards utilisés. Dans la deuxième, nous présentons les services web sémantiques en évoquant leur évolution du syntaxique au sémantique, ainsi que les standards utilisés dans la description sémantiques des services web.

Dans le deuxième chapitre, intitulé **Services web sociaux**, nous allons parler de la valeur ajoutée de l'aspect social des services web, nous donnons une définition des services web sociaux et comment procéder à leurs développement.

Dans le troisième chapitre, intitulé **Analyse des réseaux sociaux**, qui se compose lui aussi de deux parties, dans la première partie, nous présentons le concept d'un

---

réseau social, les notions et la terminologie issue de la théorie des graphes. Nous terminons ce chapitre par une introduction sur l'analyse et la fouille de données sociale.

Dans le quatrième chapitre intitulé **l'approche adoptée ou proposée**, on fait une présentation générale des outils utilisés pour la mise en œuvre de notre propre solution, tout en expliquant son principe général de fonctionnement pour la détection des communautés des services web.

# Chapitre 1

## Les services Web Sémantiques

### Sommaire

---

<b>1.1</b>	<b>Introduction</b>	<b>4</b>
<b>1.2</b>	<b>Architecture Orientée Services</b>	<b>4</b>
<b>1.3</b>	<b>Les services web</b>	<b>5</b>
1.3.1	Définition	6
1.3.2	Caractéristiques des services web	6
1.3.3	Fonctionnement des Services Web	7
1.3.4	Standards utilisés pour le web service	8
1.3.5	Conclusion	11
<b>1.4</b>	<b>Les services web sémantique</b>	<b>12</b>
1.4.1	Évolution vers le Web sémantique	12
1.4.2	Définition des services web sémantique	13
1.4.3	L'objectif des services web sémantique	13
1.4.4	L'architecture des services web sémantiques	14
1.4.5	Descriptions sémantiques des services web	15
<b>1.5</b>	<b>Conclusion</b>	<b>18</b>

---

*” Un dernier service rendu à propos, fut-il même léger,  
est capable d’effacer une grande offense. ”*

[Thucydide]

## 1.1 Introduction

Les Services Web Sémantiques se situent à la convergence de deux domaines de recherche importants qui concernent les technologies de l'Internet : le Web Sémantique et les Services Web. Le Web sémantique s'intéresse principalement aux informations statiques disponibles sur le Web et les moyens de les décrire de manière intelligible pour les machines. Les services web, quant à eux, ont pour préoccupation première l'interopérabilité entre applications via le Web en vue de rendre le Web plus dynamique.

Dans ce chapitre on va tout d'abord discuter sur le modèle architectural SOA (Service Oriented Architecture) qui est l'un des modèles qui offrent des moyens de communication entre les entreprises dans le web. Nous parlerons aussi plus loin dans cette partie de l'implémentation la plus importante de l'architecture orientée service à savoir les services Web. Vers la fin de ce chapitre, nous allons présenter le concept des services web sémantique.

## 1.2 Architecture Orientée Services

L'Architecture Orientée Service (SOA) est un style d'architecture où le Service désigne le fondement de ce modèle d'interaction entre applications. Son objectif consiste à favoriser le couplage faible entre les modules et la réutilisation de composants logiciels.

SOA est le terme utilisé pour désigner un modèle d'architecture permettant l'exécution d'applications logicielles réparties [1]. C'est le moyen de réorganisation d'un ensemble d'applications logicielles et d'infrastructures dans un groupe interconnecté de services, chacun est accessible par des interfaces standards et des protocoles de messagerie. Cette approche architecturale est particulièrement appliquée lorsque plusieurs applications en cours d'exécution basées sur des technologies et des plates-formes variées ont besoin de communiquer les unes avec les autres. De cette façon, les entreprises peuvent mélanger et assortir les services pour effectuer des transactions métier avec un effort minimal de programmation [2].

Ce style architectural repose sur un modèle qui définit un ensemble d'interactions entre différents acteurs (voir la figure ci-dessous). Ainsi les interactions sont :

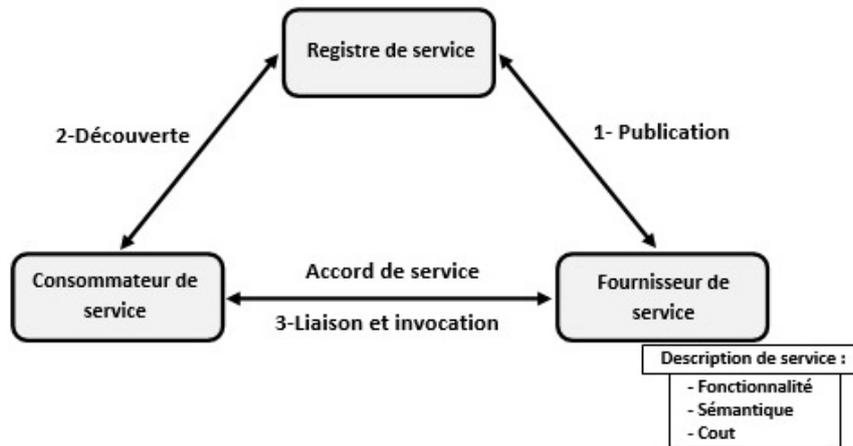


FIGURE 1.1 – Acteurs et interaction dans l'architecture orientée service

1. **publication de services** : les fournisseurs de service enregistrent leur service dans l'annuaire ;
2. **la découverte d'un service** : les consommateurs de service interrogent l'annuaire pour trouver un service qui correspond à leurs besoins ;
3. **la liaison et l'invocation d'un service** : une fois le service choisi, le consommateur de service peut se lier au fournisseur et utiliser le service.

L'avantage certain de cette architecture est que seule la description de service est partagée entre les différents acteurs ; ceci permet d'obtenir un très faible couplage, par conséquent il apparaît un autre avantage : l'hétérogénéité des implémentations et des plates-formes est masquée au consommateur de service, tout comme la localisation du service. Aussi grâce à cette architecture, nous obtenons une nouvelle propriété : la substituabilité. En effet, il est possible de remplacer un service par un autre de façon transparente grâce à l'interface du service dès lors qu'il respecte le contrat que le fournisseur et le client ont passé.

### 1.3 Les services web

Les services web regroupent tout un ensemble de technologies bâties sur des standards. Ils permettent de créer des composants logiciels distribués, de décrire leur interface et de les utiliser indépendamment de la plateforme sur laquelle ils sont implémentés. Les services web fournissent les bases technologiques nécessaires à la réalisation de l'interopérabilité entre les applications en utilisant différentes plateformes, différents systèmes d'exploitation et différents langages de programmation.

### 1.3.1 Définition

Les services Web sont une nouvelle technologie pour la création des applications accessibles à travers l'infrastructure du Web. Cette conception des spécifications services Web a été menée dans l'objectif de répondre au mieux aux enjeux de l'architecture SOA. Ils ont été proposés initialement par IBM<sup>1</sup> et Microsoft [3], puis en partie standardisé sous l'égide du W3C<sup>2</sup>. Parmi les définitions on a choisi les deux définitions suivantes :

**Définition 1 :** Selon IBM, *Les services Web sont des applications modulaires autonomes qui peuvent être décrits, publiés, situés, ou invoquées sur un réseau, en général, le World Wide Web*[4].

**Définition 2 :** le consortium W3C définit un service Web comme étant : *Un service Web est un composant logiciel identifié par une URI, dont les interfaces publiques sont définies et appelées en XML. Sa définition peut être découverte par d'autres systèmes logiciels. Les services Web peuvent interagir entre eux d'une manière prescrite par leurs définitions, en utilisant des messages XML portés par les protocoles Internet*[5].

On peut tirer de ces définitions qu'un service web est vu comme une application accessible pour les autres applications à travers le Web. Le deuxième point, les services Web sont ouverts, ce qui signifie que les services publient des interfaces qui peuvent être invoquées par le biais des messages XML standardisés.

### 1.3.2 Caractéristiques des services web

Les services Web possèdent les caractéristiques suivantes qui leur permettent une meilleure intégration dans les environnements hétérogènes :

- **Basé sur XML**<sup>3</sup> : Les données dans les protocoles et les technologies des services web sont représentées en utilisant XML, ces technologies peuvent être interopérables. Comme un transport de données, XML élimine toute dépendance de gestion de réseau, du système d'exploitation, ou de la plateforme liée à un protocole ;
- **Faiblement couplés** : Dans le développement de logiciels, le couplage se rapporte typiquement au degré de dépendance entre les composants/modules logiciels.

---

1. International Business Machines Corporation

2. World Wide Web Consortium (<http://www.w3.org/>)

3. XML (eXtensible Markup Language) est un langage de balisage extensible s'écrit grâce à des balises. Ces balises permettent de structurer de manière hiérarchisée et organisée les données d'un document.

Contrairement aux composants fortement couplés (tels que CORBA ou COM), les services web sont autonomes et peuvent fonctionner indépendamment les uns des autres. Il n'est pas nécessaire de connaître la machine, le langage ou le système d'exploitation ;

- **Auto-descriptif** : Les services web ont la capacité de se décrire d'une manière qui peut être facilement reconnu. Ainsi, l'interface, les informations de localisation et l'accès au service web est identifié par n'importe quelle application externe ;
- **Modulaire** : Les services web fonctionnent de manière modulaire. Cela signifie qu'au lieu d'intégrer dans une seule application globale toutes les fonctionnalités, on crée plusieurs applications spécifiques et on les fait inter-opérer entre elles, et qui définissent chacune, une de ses fonctionnalités ;
- **Réutilisable** : Une fonctionnalité, développée sous forme de service web, peut être réutilisée et combinée à d'autres fonctionnalités afin de composer de nouveaux services.

### 1.3.3 Fonctionnement des Services Web

Dans sa première génération, le fonctionnement des Services web repose sur trois couches fondamentales présentées comme suit :

- **Description** : dont l'objectif est la description des interfaces des Web services, (paramètres des fonctions, types de données).
- **Découverte** : permettant de localiser un Web service particulier dans un annuaire de services décrivant les fournisseurs ainsi les services fournis.
- **Invocation** : visant à établir la communication entre le client et le fournisseur en décrivant la structure des messages échangés.

Le fonctionnement des services web repose sur un modèle en couches, dont les quatre couches fondamentales sont les suivantes :

- **La couche publication** : repose sur l'UDDI (voir section 1.3.4.2), qui assure le regroupement, le stockage et la diffusion des descriptions de services web.
- **La couche description** : est prise en charge par le langage WSDL (voir section 1.3.4.1), qui décrit les fonctionnalités fournies par le service web, les messages reçus et envoyés pour chaque fonctionnalité, ainsi que le protocole adopté pour la communication. Les types des données contenues dans les messages sont décrits à l'aide du langage XML Schéma.
- **La couche message** : utilise des protocoles reposant sur le langage XML. Actuellement, SOAP (voir section 1.3.4.3) est le protocole utilisé pour cette couche.

- **La couche transport** : repose sur le protocole http. Le protocole le plus utilisé sur internet pour le transport de données et de messages.

On va présenter maintenant le scénario classique du processus d'invocation d'un service web qui est décrit comme suit (voir figure 1.2)

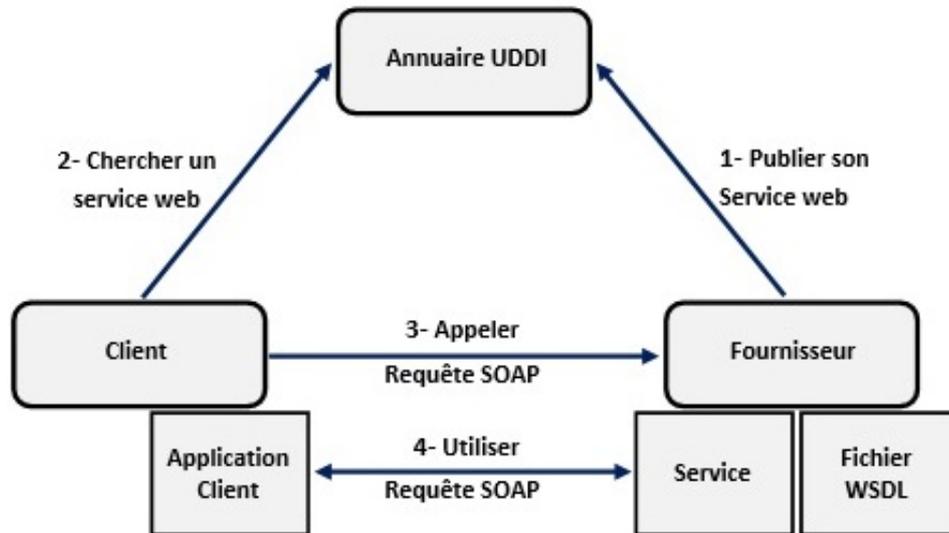


FIGURE 1.2 – Fonctionnement des services web

1. Le fournisseur de service Web déploie son service et le rend accessible à travers internet. Ensuite, il le publie dans un annuaire en décrivant ses caractéristiques dans un fichier WSDL ;
2. Ensuite, le client du service Web envoie une requête SOAP à l'annuaire de service en spécifiant ses propres exigences de recherches. L'annuaire cherche les services demandés selon les critères spécifiés et retourne au client tous les URL des services disponibles ;
3. Une fois le client sélectionne le service voulu, il établit un contrat avec le fournisseur pour invoquer le service Web ;
4. Le client interagit à partir de son application avec le service Web à travers des requêtes SOAP en spécifiant les méthodes à appeler.

### 1.3.4 Standards utilisés pour le web service

Les technologies de service Web se base sur un ensemble de protocoles standards pour l'échange de données entre les applications ou assurer l'interaction entre les opérations de recherche, de lien et de publication des services web. Un ensemble de spécifications considérées comme des standards ont été définies par le consortium W3C. Dans cette section, nous décrivons ces standards.

#### 1.3.4.1 WSDL (langage de description des services Web) :

WSDL est une spécification qui décrit les services Web disponibles pour les clients. C'est un langage basé sur XML, qui spécifie un service Web à l'aide des messages qui fournissent une définition abstraite des données à transmettre et les opérations fournies par un service Web pour transmettre les messages [6], ce qui signifie que WSDL est à la fois neutre par rapport au langage de programmation et à la plateforme. Outre la description du contenu des messages, WSDL définit l'endroit où le service est disponible et le protocole de communications utilisé pour converser avec le service. Cela signifie que le fichier WSDL définit tout ce qui est nécessaire pour écrire un programme fonctionnant avec un service Web.

Un document WSDL définit une suite de descriptions de composants (figure suivante). Le fichier WSDL est principalement composé de six éléments, avec d'autres éléments optionnels :

- **Définitions** : élément racine du document, il donne le nom du service, déclare les espaces de noms utilisés, et contient les éléments du service.
- **Types** : décrit tous les types de données utilisés entre le client et le prestataire. WSDL n'est pas exclusivement lié à un système spécifique de typage, mais utilise par défaut la spécification XML Schema.
- **Message** : décrit un message unique, que ce soit un message de requête seul ou un message de réponse seul. L'élément définit le nom du message et peut contenir (ou pas) des éléments `part`, qui font référence aux paramètres du message ou aux valeurs retournées par le message.
- **PortType** : combine plusieurs messages pour composer une opération. Chaque opération se réfère à un message en entrée et à des messages en sortie.
- **Binding** : décrit les spécifications concrètes concernant la manière dont le service sera implémenté : protocole de communication et format des données pour les opérations et messages définis par un type de port particulier. WSDL possède des extensions internes pour définir des services SOAP ; de fait, les informations spécifiques à SOAP se retrouvent dans cet élément.
- **Service** : définit les adresses permettant d'invoquer le service donné, ce qui sert à regrouper un ensemble de ports reliés. Plupart du temps, c'est une URL invoquant un service SOAP.

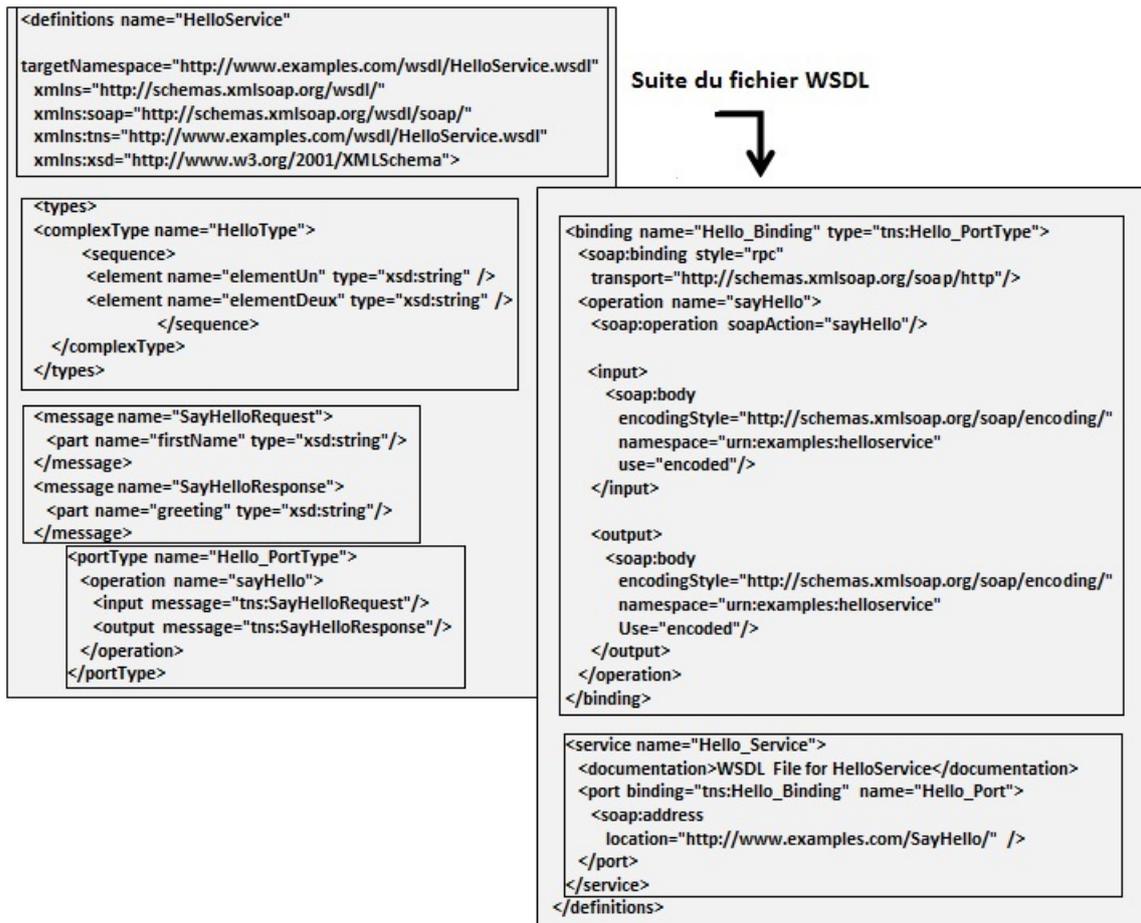


FIGURE 1.3 – Exemple de fichier WSDL

### 1.3.4.2 UDDI : Annuaire des services web :

UDDI est une spécification d'annuaire qui propose aux utilisateurs une façon unifiée et systématique pour trouver des fournisseurs de services à travers un registre centralisé de services [7]. UDDI a été initialement conçu par des industriels, en ayant pour but d'avoir un standard indépendant des plateformes d'implémentation, afin de connaître les entreprises qui fournissent des services Web et de découvrir les services Web disponibles qui répondent aux attentes du client. Une entrée du répertoire UDDI est constituée d'un fichier XML [7].

Chaque entrée du répertoire UDDI est constituée de trois parties.

- Les *pages blanches* décrivent l'entreprise qui offre le service : nom, adresse, contacts, etc.
- Les *pages jaunes* fournissent les informations d'une catégorisation basée sur les types d'entreprises et de services
- Les *pages vertes* décrivent l'interface vers le service avec suffisamment de détail pour qu'il soit possible d'écrire une application permettant d'utiliser le service

Web.

WSDL et UDDI sont deux standards pour les services Web. Il existe un autre point important dans l'architecture des services Web ; ce sont les communications entre les différentes parties qui se font avec le protocole SOAP, présenté dans la section suivante.

#### 1.3.4.3 SOAP (Simple Object Access Protocol) :

SOAP est un protocole de langage neutre, du réseau, du transport, et de la programmation qui permet à un consommateur de service d'appeler un fournisseur de services à distance. Le format du message est basé XML [8]. Un message SOAP contient une enveloppe obligatoire, un en-tête facultatif et un corps obligatoire qui se présente sous la forme d'un document XML.

- **Enveloppe (envelope)** : contient la spécification des espaces de désignation (namespace), du codage de données (version du schéma XML-SOAP supporté), etc...
- **En-tête (header)** : est un élément facultatif qui supporte des informations auxiliaires telles que l'authentification, les transactions et les paiements. Tout élément dans une chaîne de traitement SOAP peut ajouter ou supprimer des éléments de l'en-tête ; les éléments peuvent également choisir d'ignorer les articles s'ils ne sont pas connus.
- **Corps (body)** : contient le nom de la méthode à invoquer ainsi que ses arguments et l'adresse cible de service.

Le protocole SOAP s'appuie sur des standards de communication comme le protocole HTTP, mais il peut aussi utiliser des protocoles autres comme SMTP. L'avantage d'utiliser SOAP avec le protocole HTTP est que la communication est facilitée, en particulier les proxys et les pare-feux peuvent être franchis sans problème. Il est ainsi facilement adaptable à toutes les technologies antérieures, tout en restant simple et extensible.

#### 1.3.5 Conclusion

La technologie des services Web offre de fortes potentialités pour surmonter les problèmes d'interopérabilité des systèmes. Elle constitue un cadre prometteur pour l'intégration des applications, et pour la gestion des interactions entre divers partenaires dans un environnement distribué, hétérogène, ouvert et versatile qui est le Web.

La plupart des travaux existants qui s'intéressent à l'intégration fonctionnelle évitent le problème fondamental de l'automatisation des différentes étapes liées à la fourni-

ture d'un service web (par exemples, découverte et composition) puisqu'ils limitent l'usage des services web aux utilisateurs humains plutôt qu'aux machines. Il semble donc nécessaire de tendre vers des services intelligibles pour des machines : c'est le concept des services web sémantique.

## 1.4 Les services web sémantique

Avant de décrire la notion des services web sémantique, il est important de présenter le web sémantique

### 1.4.1 Évolution vers le Web sémantique

Depuis le début des années 2000, de nombreux axes de recherches concernant Internet se sont tournés vers le Web sémantique. Internet est un immense réservoir de documents en tout genre (articles, sites personnels, audio, vidéo) mais plus ces ressources s'accumulent, plus la recherche d'une information précise s'avère difficile.

Tim Berners-Lee [9], pionnier dans le domaine d'Internet, propose alors d'ajouter à toutes ces ressources une sémantique qui permettrait aux systèmes informatiques de comprendre le sens en accédant à des collections structurées d'informations et a des règles d'inférences qui peuvent être utilisé pour conduire des raisonnements automatisés : c'est la naissance du Web sémantique. Le Web sémantique n'est pas un Web séparé, mais une extension du web actuel, dans lequel le sens d'information est bien défini, permettant aux ordinateurs et aux gens de travailler en coopération. Le Web sémantique apportera une structure pour la signification du contenu des pages Web, et la création d'un environnement où les agents logiciels peuvent naviguer d'une page vers une autre facilement et effectuer des tâches complexes pour les utilisateurs.

Le Web sémantique constitue le point de départ pour le développement des services Web intelligents. Ses objectifs principaux sont le partage et la réutilisation d'information pour l'homme et machine. Donc, on doit ajouter la connaissance à l'information pour améliorer la compréhension pour la machine. Pour le faire, simplement, on annote des ressources avec la signification des concepts. Donc, on a besoin de même concept pour représenter la même chose dans l'indexation de différentes ressources. Dans ce cas, on a besoin le terme "Ontologie"<sup>4</sup>

L'ontologie est un composant essentiel et important du Web sémantique qui permet à l'utilisateur lors d'une recherche sur le Web d'accéder non seulement aux documents liés aux mots clés de la requête, mais aussi à ceux qui sont liés ontologiquement

---

4. une ontologie sert à structurer des concepts d'un domaine. Ces concepts sont rassemblés pour fournir les briques élémentaires et exprimer les connaissances dont on dispose dans ce domaine.

(sémantiquement) à ces derniers, ce qui rend la recherche encore plus pertinente. Elle a pour but de décrire des concepts et les relations qui les lient entre eux, et avec des règles de déduction les rendre plus compréhensibles et utilisables par les différents agents (humains ou logiciels).

### 1.4.2 Définition des services web sémantique

Un service web sémantique est le résultat de l'intégration de concepts sémantiques à la description du service web [10]. Le concept des services web sémantiques se trouve à la convergence de deux domaines de recherche importants concernant les technologies de l'internet : le web sémantique et les services web (voir la figure 1.4). La combinaison de ces deux tendances promet un Web entièrement mécanisée pour l'interaction de l'ordinateur, où les gens organisent leur collaboration et de leurs relations d'affaires [11].

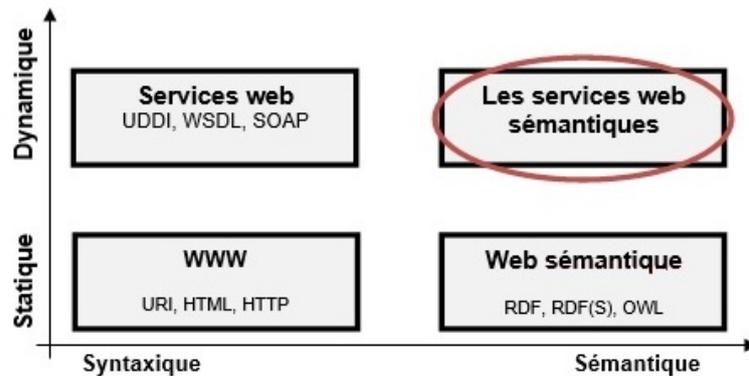


FIGURE 1.4 – l'Evolution du web [12]

Cette tâche de convergence est accomplie en rendant les services Web auto-exploitable par machines, et de réaliser l'interopérabilité entre les applications via le Web en vue de rendre le Web plus dynamique.

La notion de services Web sémantiques est synonyme de l'automatisation des tâches d'utilisation de services tels que la découverte, la sélection, la composition, et l'adoption de services appropriés.

### 1.4.3 L'objectif des services web sémantique

L'objectif des services web sémantique est de créer un web sémantique des services web dont les propriétés, les capacités, les interfaces et les effets sont décrits de manière non ambiguë et exploitable par des machines et ce en utilisant les couches techniques sans pour autant en être conceptuellement dépendants [13].

La combinaison des technologies du Web sémantique à celles des Web services permettra de :

- Sélectionner des services web ;
- Automatiser la découverte des services web, c'est-à-dire la localisation automatique des services web qui fournissent une fonctionnalité particulière et qui répondent aux propriétés demandées par l'utilisateur ;
- Automatiser la découverte des services web, c'est-à-dire la localisation automatique des services web qui fournissent une fonctionnalité particulière et qui répondent aux propriétés demandées par l'utilisateur ;
- Automatiser l'invocation d'un Web service, cela implique l'automatisation de l'exécution du service web par le programme d'utilisateur ou par un agent logiciel.

#### 1.4.4 L'architecture des services web sémantiques

Le groupe architecture du W3C travaille activement à l'élaboration d'une architecture étendue standard. Cette architecture est basée sur des services web, la pile est constituée de plusieurs couches, chaque couche s'appuyant sur un standard particulier. On retrouve, au-dessus de la couche de transport, les trois couches formant l'infrastructure de base décrite précédemment.

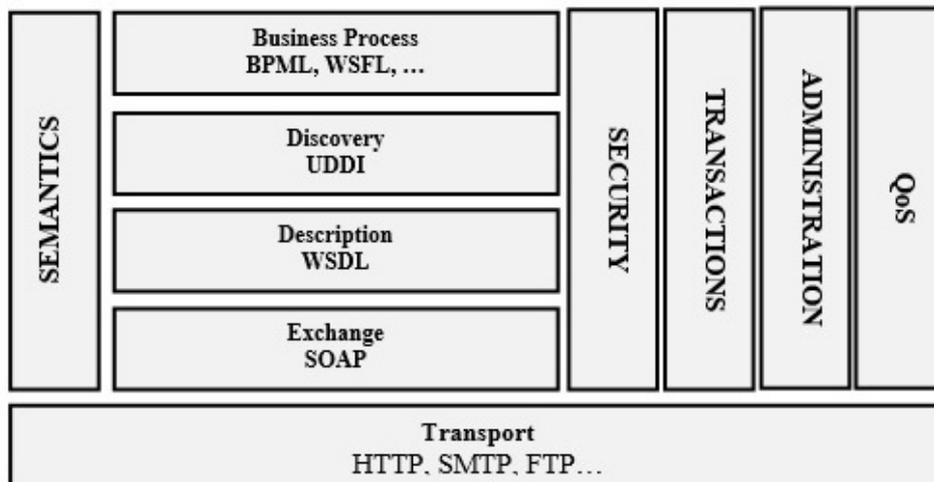


FIGURE 1.5 – Pile des services web sémantique [14]

Ces couches s'appuient sur les standards émergents SOAP, WSDL et UDDI. Une couche de *business process* est ajoutée, permettant de rendre les processus métiers accessibles à l'intérieur d'une entreprise et au-delà même de ces frontières. Les couches transversales : sécurité, transactions, administration et QoS, permettent de compléter

la couche *business process*. La couche *sémantique* vient s'intégrer à chacune des quatre couches supérieures et permettant ainsi d'automatiser ces processus.

### 1.4.5 Descriptions sémantiques des services web

Un service Web est accessible à travers sa description WSDL mais cette dernière contient des détails techniques de communication avec le service. Par exemple les fonctions du service Web, les inputs de ces fonctions, les outputs des fonctions. Ces détails ne sont pas suffisant pour bien décrire la fonctionnalité d'un service. Alors les chercheurs ont rajouté une deuxième couche de description c'est la couche sémantique.

Deux méthodes possibles pour la description sémantique de Web services [15] :

- La première approche consiste à annoter les langages existants (WSDL) avec l'information sémantique. Le principal avantage de ce genre de solutions est la facilité pour les fournisseurs de services d'adapter leurs descriptions existantes aux annotations proposées. Exemple : SAWSDL.
- La deuxième approche réécrit entièrement la description syntaxique du Web services en utilisant les technologies du Web sémantique Exemple : OWL-S.

#### 1.4.5.1 SAWSDL (Semantic Annotation for WSDL) :

SAWSDL s'inscrit dans le cadre des approches à base d'annotations. C'est un langage recommandé par le W3C pour la description sémantique des services Web. Il est évolutif et compatible avec les standards de services Web existants, en particulier avec WSDL. En effet, il augmente l'expressivité du langage WSDL avec la sémantique en utilisant des concepts de l'ontologie. SAWSDL fournit un mécanisme permettant d'annoter sémantiquement les types de données et les opérations de WSDL. En particulier, il ajoute aussi de nouveaux éléments pour des spécifications spéciales. Pour ce faire, il définit un ensemble d'attributs pour l'extension de WSDL en proposant un mécanisme d'annotation indépendant du langage de représentation sémantique. Les annotations sémantiques sont réalisées par le biais des références à des modèles conceptuels comme les ontologies. Au lieu de spécifier un langage pour représenter les modèles sémantiques, SAWSDL prévoit des mécanismes par lesquels les concepts des modèles sémantiques peuvent être référencés à l'aide des annotations.

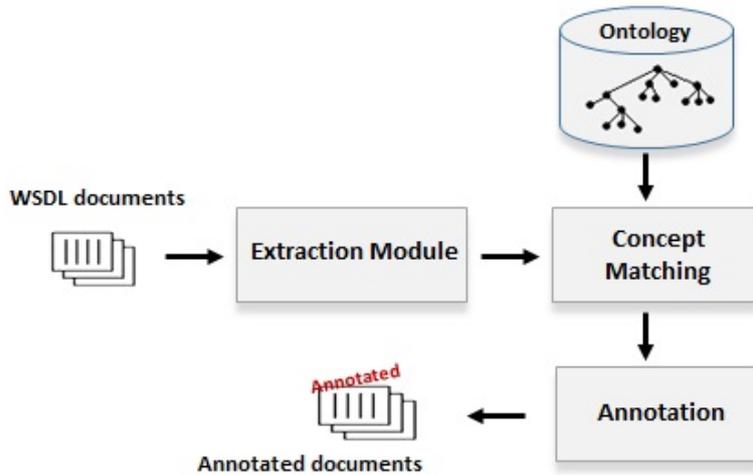


FIGURE 1.6 – Processus d’annotation sémantique

#### 1.4.5.2 OWL-S (Ontology Web Language for Services) :

OWL-S est un sous ensemble du langage OWL dédié à la description sémantique de Web services. Il permet de d’écrire d’une façon non ambiguë les Web services de telle sorte qu’un agent logiciel puisse exploiter automatiquement ces informations. OWL-S permet la découverte automatique et la composition de Web services. Une description OWL-S se compose de trois éléments (voir Figure suivante) : le service Profile, le service Model, et le service Grounding, qui décrivent respectivement que fait le service, comment le service fonctionne et comment accéder au service.

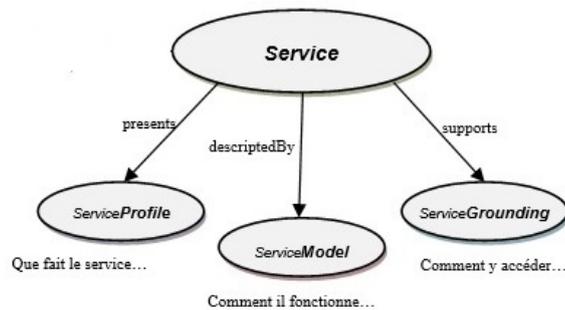


FIGURE 1.7 – Model conceptuel d’OWL-S [11]

— **Service profile** :OWL-S fournit cette classe pour décrire un Web service. Cette classe Service Profile spécifie trois informations :

- **Nom du service**, contacts et description textuelle du service : le nom du service est utilisé comme identificateur du service, tandis que les informations contacts et la description textuelle sont destinées aux utilisateurs humains ;
- **Description fonctionnelle du service** : Elle spécifie ce que l’on peut attendre du service en termes d’entrées attendues et de résultats produits en

sortie. Les transformations d'informations sont représentées par des Inputs et des Outputs. Le changement d'état du monde réel cause par l'exécution du service est représenté par préconditions et effets qui sont les préconditions et les post conditions de son exécution. Les Inputs et les Outputs font références a des classes d'OWL décrivant les types des instances à envoyer au service et aux réponses respectives attendues ;

- **Propriétés additionnelles** : Plusieurs propriétés s sont utilisées pour définir un Web service. La première est la catégorie du Web service. La seconde est la qualité de Web service QoS. Enfin, le Web service peut fournir une liste de paramètres de fa con libre.

— **Service model** : Les Web services peuvent être décrits en OWL-S en tant que processus grâce à la classe Process. La classe ainsi définie est une sous-classe de Service model. Pour décrire un processus, on spécifie ses entrées, sorties et ses états (préconditions et effets). Les transitions d'un état à un autre sont décrites par les préconditions et les effets de chaque processus. Il existe trois types de processus :

- **Les processus atomiques (AtomicProcess)** : exécutable en une seule étape, un processus atomique ne peut pas être décomposé de fa con plus profond. Son exécution correspond à une unique avancée dans l'exécution du service, il est directement invoqué par l'utilisateur du service ;
- **Les processus composites (CompositeProcess)** : un processus composite est constitué par l'assemblage d'autres processus (eux-mêmes peuvent être composites ou non composite). Les processus composites associent des processus à l'aide de structures de contrôle permettant de décrire leur logique d'exécution ;
- **Les processus simples (SimpleProcess)** : Les processus simples ne sont pas invocables mais comme les processus atomiques, leurs exécutions s'effectuent en une seule étape. Les processus simples sont employés comme éléments d'abstraction. Un processus simple peut être employé pour fournir une vue d'un certain processus atomique, ou une représentation simplifiée d'un certain processus composite.

— **Service grounding** : La classe OWL-S Service Grounding définit les détails techniques permettant d'accéder au Web service. Les deux premières classes Service Profile et Service Model d'une description OWL-S s'attachent à abstraire la représentation d'un Web service. Service Grounding est la forme concrète d'une représentation abstraite, elle fournit les détails concrets d'accès au Web service, tels les protocoles, les URIs, les messages envoyés, etc.

## 1.5 Conclusion

Les technologies du web sémantique ont été utilisées pour enrichir les services Web sémantique, ce qui permet l'automatisation des divers aspects relatifs aux services Web. Les avantages de l'utilisation du Web sémantique pour la description des services Web sont nombreux. En plus, de rendre l'interface du service Web accessible automatiquement par des machines, ils permettent également, la description de propriétés non fonctionnelles telles que la qualité de services, les contraintes de sécurité, et l'intégration effective des services Web dans des applications industrielles, d'une manière uniforme compréhensible par tous.

Dans le chapitre suivant, nous allons présenter la notion des services Web sociaux du au limites du l'approche orienté services découvrent par un ensemble de chercheurs dont ils ont proposé une solution avec l'utilisation du l'aspect social aux services web.

# Chapitre 2

## Les Services Web Sociaux

### Sommaire

---

2.1	Introduction . . . . .	20
2.2	Services web en deçà de leur potentiel . . . . .	20
2.3	Pourquoi l'aspect social est utile ? . . . . .	21
2.4	Définition des services Web sociaux . . . . .	23
2.5	Les Services Web Sociaux en action . . . . .	23
2.6	Développement des services Web sociaux . . . . .	24
2.7	Conclusion . . . . .	30

---

*” Qualifions de don toute prestation de bien ou de service effectuée ,sans garantie de retour,en vue de créer, nourrir ou recréer le lien social entre les personnes.”*

[Jacques Godbout]

## 2.1 Introduction

Les utilisateurs de SOA défendent régulièrement pour ses avantages, l'état actuel de SOA n'a pas vraiment maintenu ces avantages : les applications actuelles de SOA sont principalement conçues pour des environnements fermés, statiques lors de l'exécution, et reposent principalement sur des caractéristiques et des méthodes formelles. Donc comment peut-on supprimer ces limitations ?

## 2.2 Services web en deçà de leur potentiel

Les limitations des services web se présentent en cherchant à répondre à plusieurs questions importantes qui sont restées sans réponse, par exemple :

- où faire de la publicité pour des services de meilleure qualité ?
- Comment découvrir les services par rapport aux besoins des utilisateurs ?
- Comment faire confiance à des services quand ils sont trouvés ?
- et comment remplacer les services facilement quand ils échouent ? [16]

Il est important de mentionner que les approches de découverte utilisée portent sur des services Web en tant que des composants isolés qui n'ont pas d'interaction entre eux, ou ne prennent pas les connaissances des utilisateurs en considération. Cela nous amène à la conclusion que le processus de découverte doit prendre en compte non seulement le service Web souhaité, mais il doit tenir compte de ses interactions avec l'utilisateur et les pairs ainsi. Pour illustrer l'importance de ce point, Z. Maamar et al [17] listent les caractéristiques des services Web quand ils fonctionnent comme des entités séparés : qui ont rendu les services en deçà de leur potentiel :

- ne savent que sur eux-mêmes, et non pas sur leurs utilisateurs ou les pairs ;
- limitent l'intervention considérable des utilisateurs (consommateurs) et fonctionnent comme des boîtes noires ;
- ne peuvent déléguer leurs invocations ;
- ne coopèrent pas instantanément et spontanément les uns avec les autres ou l'auto-organisation ;
- ne peuvent pas rapprocher les ontologies entre eux ou avec leurs utilisateurs.
- ne considèrent que leurs propres détails internes fonctionnels et non fonctionnels pendant l'exécution et ignorent les autres détails extérieurs, tels que les interactions passées de l'utilisateur ;

Les incarnations actuelles des services Web ne sont pas pratiques et presque inutilisable, sauf dans des environnements d'entreprise rigoureusement contrôlés. Le problème se situe au niveau de la sémantique utilisée pour caractériser les services ; la sémantique dans la description des services WSDL, ou dans les extensions proposées à WSDL, est insuffisante pour la découverte automatique. Il a été suggéré que la communauté des utilisateurs pourrait être en mesure de fournir une description sémantique par le biais d'un effort comme Wikipédia. Cependant, pour des services appropriés et utiles à découvrir et à invoquer, une sémantique suffisamment précise pour décrire ces derniers doit être combinée avec un logiciel assez intelligent pour comprendre les descriptions sémantiques. Par ailleurs, l'amélioration de la description sémantique améliore seulement la partie découverte dans le problème des services et non la partie exécution. Heureusement, des sémantiques de comportement sont disponibles sur la façon dont les services sont utilisés et combinés, et comment ils se comportent et interagissent. Ceci est l'aspect social des services. [17].

## 2.3 Pourquoi l'aspect social est utile ?

Nous présentons un exemple qui illustre la façon dont les services sont utilisés sans aucune référence à des éléments sociaux.

- Pour trouver la définition d'un mot en anglais et le traduire en français, puis l'envoyer par courrier, Utilisateur1 crée un mashup composé des services disponibles par exemple un service de Dictionnaire et un autre appelé PostTwitter voir la figure 2.1.
- Pour trouver les prévisions météorologiques, les traduire en français de l'anglais, et les afficher sur son profil Twitter, Utilisateur1 crée un autre mashup en utilisant Météo, Traducteur, et PostTwitter.



FIGURE 2.1 – L'expérience de l'utilisateur 1

Ici, l'utilisateur 1 utilise le service Traducteur pour la deuxième fois sans s'interroger sur la façon dont il s'est comporté/opéré quand il a été utilisé pour la première fois. En effet, il a connecté Traducteur et PostTwitter de nouveau sans tenir compte

de l'issue de la première connexion si elle a réussi ou non ? Ce détail n'est mentionné nulle part, sauf si l'utilisateur 1 décide de garder une trace de tout ce qu'il a fait, ce qui n'est ni approprié ni faisable. Est-ce que lors de l'élaboration du mashup, le service Traducteur peut par exemple conseiller l'utilisateur 1 en douceur de ne pas utiliser PostTwitter ?

- Un autre utilisateur 2, veut créer un mashup qui trouve la description météo pour sa ville afin qu'il puisse la mettre sur son blog. Il utilise les services suivants : My-Location, Météo, Traducteur, BlogPost, TinyURL, et soit PostTwitter ou Courriel voir figure 2.2.



FIGURE 2.2 – L'expérience de l'utilisateur 2

- Supposons que Utilisateur 3 veut aussi créer un mashup basé sur un service de prévision météorologique. Peut-il bénéficier d'un des services du mashup de l'utilisateur 2 ? Traditionnellement, la réponse est "non". Les invocations et les compositions de services réussis ne sont pas sauvegardés pour un usage ultérieur, ni même les échecs. Les services sont considérés comme des éléments isolés malgré leurs interactions précédentes avec d'autres pairs où les services complexes sont construits.

La capture des interactions du service en utilisant, par exemple, les réseaux sociaux pourrait être utile pour les ingénieurs qui peuvent tirer parti sur les interactions connues avec succès selon les besoins [16].

La première interaction concerne la sélection qui mène à l'identification, dans ce cas, Prévisions météo un autre service pair comme Météo. Ces deux services sont en compétition parce qu'ils font le même travail, qui consiste à fournir des informations météorologiques. La seconde interaction concerne les dépendances d'exécution entre les services qui peuvent devenir récurrentes au fil du temps. Traducteur et PostTwitter ont participé à plusieurs compositions communes. Enfin, la troisième interaction concerne la fiabilité du service. Lorsque PostTwitter échoue, Courriel prend automatiquement le relais. Si les réseaux :

- Être au courant des homologues qui sont en compétition avec lui en cas de sélection, comme Météo et Prévisions météo ;
- Recommander les homologues avec qui il souhaiterait collaborer dans le cas de composition, telles que Météo et Traducteur ;
- Recommander les homologues pouvant lui substituer en cas de défaillance, comme PostTwitter et Courriel.

**La compétition, la collaboration et la substitution** sont quelques liens qui peuvent connecter ensemble les services Web.

## 2.4 Définition des services Web sociaux

Nous considérons les services Web sociaux comme étant le résultat de la fusion de deux domaines : le Social Computing (l'approche sociale) avec Service-Oriented Computing SOC (l'approche orientée service). D'une part, le Social Computing consiste à faciliter le calcul des études sociales et des dynamiques sociales de l'homme ainsi que la conception et l'utilisation des technologies de l'information et de la communication qui prennent en compte le contexte social [18]. Social Computing concerne également les actions collectives, le partage de contenu et la diffusion d'informations en général. D'autre part, SOC construit des applications sur les principes de l'offre et de demande des services, et sur le faible couplage [19].

La fusion des deux domaines conduit à des services web sociaux (SWSs) qui permet de savoir avec qui ils ont interagi dans le passé et avec qui ils souhaiteront éventuellement travailler à l'avenir [16]. Ces deux éléments constituent la mémoire des actions que les SWSs peuvent accumuler avec le temps et appliquer dans l'avenir. En outre, ils montrent les collectifs de développement complexes à valeur ajoutée des services composites. Les SWSs devraient prendre l'initiative de conseiller les utilisateurs comment développer et réutiliser les services à valeur ajoutée.

## 2.5 Les Services Web Sociaux en action

l'établissement et le maintien des réseaux sociaux des services Web peut se produire de trois façons : [16]

**Collaboration** : En combinant leurs fonctionnalités respectives, les SWSs ont la capacité de travailler ensemble sur les requêtes complexes des utilisateurs. Par conséquent, un SWSs gère son propre réseau de collaborateurs, de sorte qu'il décide s'il souhaite

collaborer avec des pairs basés sur des expériences précédentes. Il peut également recommander des pairs.

**Compétition** : Les SWSs rivalisent les uns contre les autres quand ils offrent des fonctions semblables. Leurs propriétés non-fonctionnelles les différencient lorsque les exigences des utilisateurs non fonctionnels doivent être remplies. Par conséquent, un SWS fait de l'apprentissage pour connaître son propre réseau de compétiteurs, de sorte qu'il peut tenter d'améliorer ses propriétés non fonctionnelles par rapport à d'autres pairs.

**Substitution** : Bien que les SWSs rivalisent les uns contre les autres, ils peuvent tout de même s'entraider lorsqu'ils échouent, s'ils offrent des fonctions semblables. Par conséquent, un SWSs gère son propre réseau de substitution, afin qu'il puisse respecter ses accords de service quand il rencontre un échec potentiel. Il peut alors identifier ses meilleurs substituts pour répondre aux besoins des utilisateurs non fonctionnels.

Ces trois moyens, de maintien des réseaux sociaux ; peuvent être considérés indépendamment comme un réseau des comportements sociaux. Ils peuvent être le point de départ de la construction d'autres réseaux, en fonction des interactions qui arrivent entre les services Web tels que la délégation et la supervision.

## 2.6 Développement des services Web sociaux

De nombreux chercheurs ont analysé les réseaux des services web sociaux, leurs relations et leurs comportements. Certains de leurs travaux donnent des détails sur la construction de services web sociaux et l'analyse de leur comportement [20] [21]. Les auteurs de [21] discutent de la structure et la gestion d'une communauté d'un service Web en mettant l'accent sur les échanges intra-communautaires coopération pour soutenir la croissance de la communauté et de la concurrence entre la communauté de faire des communautés similaires en concurrence pour attirer plus d'utilisateurs et des fournisseurs.

On va présenter la méthode proposée par Z. Maamar et al [21], cette méthode est composée de quatre étapes où chaque étape aborde l'une des questions suivantes :

- quel genre de relations pouvant mettre les services Web en contact,
- à quel type de réseaux sociaux correspondent ces relations,
- quels éléments constituent les réseaux sociaux des services Web,
- et quels sont les comportements sociaux que peuvent manifester les services Web.

Comme toute méthode de conception, cette méthode comporte plusieurs étapes et des modèles qui correspondent à la fois à la phase d'analyse et de conception (Figure 2.3 ).

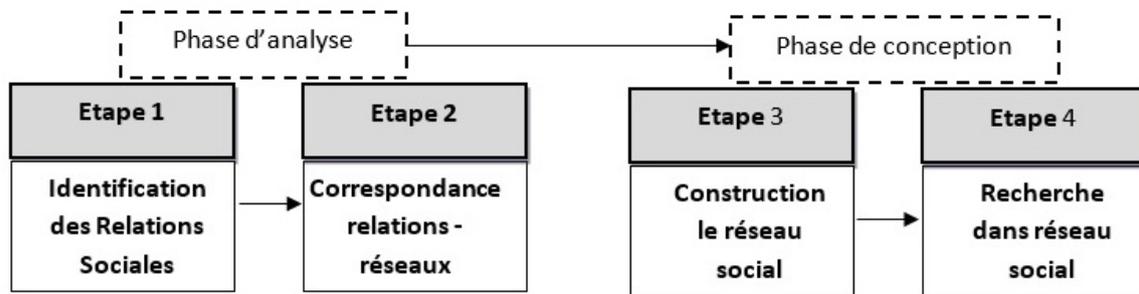


FIGURE 2.3 – Représentation générale de la méthode de conception des SWS

Au cours de la **phase d'analyse**, le concepteur effectue deux étapes. Premièrement, il établit les relations entre les services Web selon la nature de l'étude de cas. Puis, il fait une correspondance des relations établies précédemment sur les réseaux sociaux.

Au cours de la **phase de conception**, le concepteur effectue deux étapes. D'abord, il détermine les caractéristiques de chaque réseau social en termes de nombre de nœuds, les types d'arêtes reliant ces nœuds, et les formules de poids pour ces arêtes. Enfin, il analyse les comportements sociaux que les services Web présentent en intervenant à ces réseaux.

**Étape 1 : Identification des relations entre les Services web** L'objectif de cette étape est d'identifier les relations qui peuvent exister entre les services Web (figure 2.4).

1. les services Web qui offrent des fonctionnalités sémantiquement semblables
  - font de la compétition lors de la sélection, par conséquent, un seul service Web est considéré à la fois [22];
  - font de la substitution les uns aux autres en cas de panne, de sorte que la continuité opérationnelle de l'application est maintenue.
2. les services Web qui offrent des fonctionnalités distinctes collaborent à l'élaboration de nouveaux services à valeur ajoutée composites.

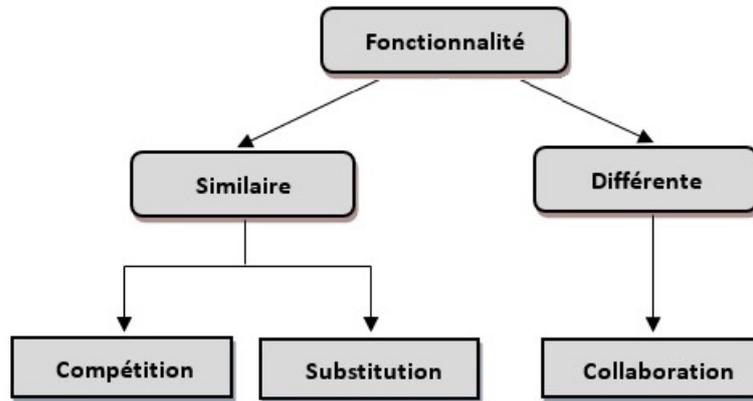


FIGURE 2.4 – Classification des SWS selon la fonctionnalité

### Étape 2 : Correspondance entre les relations et les réseaux

L'objectif de cette étape est d'identifier les réseaux sociaux potentiels qui sont susceptibles de mettre en contact les services Web. L'étape 1 abouti à l'identification des relations de compétition, de substitution et de collaboration. Chaque relation constitue une base sur laquelle un réseau social particulier est développé. Par conséquent, les services Web peuvent s'inscrire dans trois types de réseaux sociaux : la compétition, la substitution et la collaboration.

- L'objectif d'un réseau social de compétition est de mettre les services Web au courant de leurs compétiteurs. Cette sensibilisation entraîne des améliorations possibles des services Web, dans le cas où, ils se tournent régulièrement sur moins compétitifs au moment de la sélection.
- L'objectif d'un réseau social de substitution est de mettre les services Web à haute disponibilité en cas où des dysfonctionnements se produisent. Les substituts potentiels sont signalés dans ce réseau, ce qui facilite leur identification dans l'avenir.
- L'objectif d'un réseau social de collaboration est de garder une trace de tous les pairs qui ont travaillé avec un service Web dans la réalisation des compositions. Les collaborateurs potentiels sont signalés dans ce réseau, ce qui facilite leur identification dans l'avenir.

### Étape 3 : Construction des réseaux sociaux

L'objectif de cette étape est de définir les caractéristiques de chaque réseau social (à savoir, le nombre de noeuds, des arêtes, des types et les poids des matrices pour ces arêtes). Les réseaux sociaux ont deux composantes : les noeuds (correspondant aux services Web), et les arêtes (correspondant aux relations entre ces noeuds) qui formeront les trois types mentionnés précédentes du réseau social. Ici, nous allons les détailler.

**Réseau social de compétition** : La figure 2.5 illustre un simple réseau social de compétition. Puisque ce réseau implique seulement les services Web ayant les mêmes fonctions, ils sont tous en compétition les uns contre les autres et, par conséquent, tous reliés les uns aux autres à travers des arêtes bidirectionnelles.

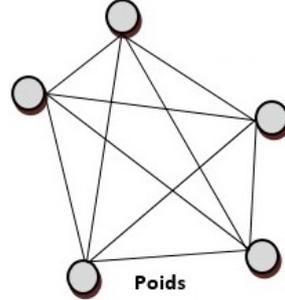


FIGURE 2.5 – Illustration d'un réseau social de compétition

Pour évaluer le poids d'une arête de compétition que nous qualifions de niveau de compétition "*Competition Level*" (CompL, équation 2.1) entre deux services Web  $w_{s_i}$  et  $w_{s_j}$ , nous utilisons le niveau de similarité de fonctionnalité "*Functionality Similarity Level (FSL)*" afin de comparer leur fonctionnalités respectives et le niveau de similarité de non-fonctionnalité "*No-Functionality Similarity Level (NFSL)*" pour comparer leurs propriétés non-fonctionnelles (QoS, par exemple, le niveau de fiabilité, temps de réponse). Nous supposons que les propriétés non-fonctionnelles des services Web sont définies avec la même taxonomie.

$$CompL_{w_{s_i}, w_{s_j}} = FSL_{w_{s_i}, w_{s_j}} \times (1 - NFSL_{w_{s_i}, w_{s_j}}) \quad (2.1)$$

où :

- $FSL_{w_{s_i}, w_{s_j}}$  correspond au niveau de similitude entre les fonctionnalités de  $w_{s_i}$  et de  $w_{s_j}$ . Ce niveau est déterminé en utilisant des approches existantes telles que ceux présentées dans [23] et devrait être soit voisin ou égal à 1.
- $NFSL_{w_{s_i}, w_{s_j}} = w_1 \times (|P_{w_{s_i},1} - P_{w_{s_j},1}|) + \dots + w_n \times (|P_{w_{s_i},n} - P_{w_{s_j},n}|)$  avec  $P_{w_{s_i},k}$  est la valeur de la k-ième propriété non fonctionnelle du ième service Web (supposé compris entre 0 et 1),  $w_k$  est un facteur de pondération représentant l'importance d'une propriété non-fonctionnelle et  $\sum_{k=1}^n w_k = 1$ .

Selon l'équation 1, plus le niveau de compétition est proche de Un, plus  $w_{s_i}$  est proche de  $w_{s_j}$ . Par conséquent  $w_{s_i}$  menace la capacité de compétitivité de  $w_{s_j}$ .

Nous rappelons que seul un service Web peut être sélectionné à la fois pour traiter la requête d'un utilisateur. Un réseau social de compétition est utile lorsqu'un service Web décide de rejeter la requête de traitement d'un utilisateur pour différentes raisons telles que la garantie de ses propriétés non fonctionnelles [24].

**Réseau social de substitution** : La figure 2.5 illustre également un réseau social de substitution après la mise à jour des noms des arêtes de compétition à substitution. Il est à noter que toutes les arêtes sont bidirectionnelles. Puisque tous les services Web dans un réseau social de substitution offrent les mêmes fonctionnalités, tout homologue peut être sélectionné comme un candidat potentiel qui remplace un service Web qui échoue.

Pour évaluer le poids d'une arête de substitution, que nous appelons le niveau de substitution "*Substitution Level (SubL)*" entre  $ws_i$  et  $ws_j$ , nous utilisons le niveau de similarité de fonctionnalité "*Functionality Similarity Level (FSL)*" et le niveau de similarité de non-fonctionnalité "*No-Functionality Similarity Level (NFSL)*" comme précédemment, en plus du niveau de fiabilité "*Reliability Level (RL)*" qui montre à quel point  $ws_i$  est, quand il remplace  $ws_j$  (équation 2.2).

$$SubL_{ws_i,ws_j} = FSL_{ws_i,ws_j} \times RL_{ws_i,ws_j} \times (1 - NFSL_{ws_i,ws_j}) \quad (2.2)$$

où :

- $FSL_{ws_i,ws_j}$  et  $NFSL_{ws_i,ws_j}$  sont définis dans l'équation 1 ;
- $RL_{ws_i,ws_j} = \frac{\sum(SR_{ws_i,ws_j})}{\sum(TR_{ws_i,ws_j})}$  avec  $\sum SR_{ws_i,ws_j}$  est le nombre total de remplacements réussis que  $ws_i$  a fait pour  $ws_j$  (pas de panne) et  $\sum TR_{ws_i,ws_j}$  est le nombre total de requêtes que  $ws_i$  a reçues pour remplacer  $ws_j$ . Si  $ws_i$  n'a jamais remplacé  $ws_j$  alors le niveau de substitution est 0.

**Réseau social de collaboration** : La figure 2.6 illustre un simple réseau social de collaboration. Il est construit lorsqu'au moins une composition de services Web est complète. Pour des raisons de navigation dans un réseau social de collaboration, un nœud d'entrée est nécessaire, et il est représenté de manière différente par rapport au reste des nœuds. Nous faisons référence à ce nœud d'entrée comme un service Web central (ou principal). Toutes les arêtes issues de ce service Web central sont unidirectionnelles pointant vers d'autres services Web.

Pour évaluer le poids d'une arête de collaboration, que nous appelons niveau de collaboration "*Collaboration Level (Coll)*" entre  $ws_i$  (central) et  $ws_j$ , nous traquons le nombre de fois que les deux services Web ont participé à des compositions mixtes,

en mettant l'accent sur le nombre total de compositions que  $ws_i$  y a participé.

$$ColL_{ws_i,ws_j} = \frac{\sum JC_{ws_i,ws_j}}{\sum TP_{ws_i}} \quad (2.3)$$

où :

- $\sum JC_{ws_i,ws_j}$  est le nombre total de participation de  $ws_i$  et  $ws_j$  dans des compositions mixtes.  $\sum JC_{ws_i,ws_j}$  et  $\sum JC_{ws_j,ws_i}$  sont égaux ;
- $\sum TP_{ws_i}$  est le nombre total de participation de  $ws_i$  dans des compositions.

#### Étape 4 : comportements sociaux des services Web

Le but de cette étape est d'identifier les comportements sociaux potentiels que les services Web peuvent présenter sur la base des informations que chaque type de réseau social (substitution, compétition et collaboration) exerce.

Différents types de comportements sociaux existent dans la réalité comme l'égoïsme, la confiance, l'opportunisme, la malveillance, la fiabilité, etc. [24]

**Comportement social d'égoïsme** : La substitution révèle l'égoïsme d'un service Web lorsque ce dernier refuse en permanence de remplacer les pairs qui échouent. Toutefois, ces pairs acceptent en continu de remplacer ce service Web quand il échoue. Un service Web peut utiliser différentes raisons pour appuyer ses décisions de refus, y compris "la peur" qu'on ne respecte pas ses propriétés non-fonctionnelles, ou l'inadéquation de remplacement d'un pair en échec selon le réseau social de compétition (*CompL* tends vers 0).

Un service Web qui est connu comme égoïste peut être ignoré par ses pairs similaires du moment qu'ils ne peuvent pas compter sur lui quand ils échouent. Des mesures correctives pourraient être prises en révisant les propriétés non-fonctionnelles de ce service Web.

**Comportement social de malveillance** : La compétition révèle la malveillance d'un service Web lorsqu'il accepte de traiter les requêtes des utilisateurs qu'il reçoit des autres pairs, bien que ce service Web ne soit pas sûr de garantir le niveau de qualité de service. Au départ, ces pairs ont refusé de traiter les requêtes de l'utilisateur pour des raisons énumérées dans la description de la qualité sociale d'égoïste, et ils espèrent que ce service Web ne les décevra pas. Ce service Web est signalé dans les réseaux sociaux de compétition de ces pairs.

**Comportement social de dominance** : La collaboration révèle la dominance d'un service Web sur un pair, lorsque ce service Web participe dans les compositions de ce pair plus que ce qu'a fait ce pair dans les compositions de ce service Web.

## 2.7 Conclusion

La convergence de l'approche sociale (illustré par les réseaux sociaux) avec l'approche orientée services (illustré par les services Web) on obtient Services Web sociaux qui peuvent offrir des solutions innovantes à certains problèmes actuels des services Web. Les réseaux sociaux illustrent la volonté des utilisateurs à partager des informations, à travailler avec les autres, et à recommander des services et des applications. Ces diverses et riches formes d'interactions reposent sur les principes de base de :

- Je propose des services que quelqu'un d'autre peut avoir besoin et
- Je besoin des services que quelqu'un d'autre peut offrir", les principes qui régissent le fonctionnement des services Web aussi.

Suite à cette intuition et avec une comparaison des services web sociaux (SWSs) aux services Web réguliers, les SWSs établissent et maintiennent des réseaux de contacts, compter sur leurs contacts en cas de besoin, former des groupes sociaux de collaboration durables solides et à long terme.

Le nombre croissant d'initiatives qui reflètent le mélange de Social Computing avec SOC est certainement un signe positif de l'importance croissante de ce domaine.

# Chapitre 3

## L'analyse Des Réseaux Sociaux (ARS)

### Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>32</b>
<b>3.2</b>	<b>Réseau social</b>	<b>32</b>
3.2.1	Représentation d'un réseau social	32
3.2.2	Graphes : Notions et terminologies	33
<b>3.3</b>	<b>Analyse des réseaux sociaux</b>	<b>35</b>
<b>3.4</b>	<b>Analyse et fouille de données sociales</b>	<b>36</b>
3.4.1	Fouille de données sociales	36
<b>3.5</b>	<b>Définition d'une communauté</b>	<b>39</b>
3.5.1	Définitions basées sur la connectivité des sommets	40
3.5.2	Définitions basées sur la similarité des sommets	41
3.5.3	Définitions basées sur une fonction de qualité	42
<b>3.6</b>	<b>Détection de communautés</b>	<b>42</b>
3.6.1	Les méthodes de classification hiérarchique	42
3.6.2	Les méthodes d'optimisation d'une fonction objective	47
3.6.3	Les méthodes à base du modèle	48
3.6.4	Les méthodes centrées graines	49
<b>3.7</b>	<b>Conclusion</b>	<b>50</b>

---

*” Le signe premier de la certitude scientifique, c'est qu'elle peut être revécue aussi bien dans son analyse que dans sa synthèse ”*

[Bachelard]

## 3.1 Introduction

L'analyse des réseaux sociaux (ARS) est une approche sociologique fondée sur l'étude de la théorie des réseaux sociaux qui conçoit les relations sociales en termes de nœuds et de liens. Les nœuds sont habituellement les acteurs sociaux dans le réseau mais ils peuvent aussi représenter des institutions, et les liens sont les relations entre ces nœuds. Dans sa forme la plus simple un réseau social se modélise pour former une structure analysable. Dans ce chapitre nous allons représenter les réseaux sociaux et leurs différentes représentations. Et après nous allons voir l'analyse structurelle et fouille de données sociales. Dans cette dernière on va focaliser sur la détection de communauté.

## 3.2 Réseau social

Quand on entend l'expression "analyse des réseaux sociaux", le terme le plus frappant est "réseau social (RS)". Il est souvent fondé sur un objectif précis ou un intérêt commun. C'est le cas pour les personnes travaillant dans un même secteur, tel que la santé, les ressources naturelles ou le commerce. Dans ce suit nous allons voir Les deux visualisations les plus connues ce sont les représentations nœuds-liens et les représentations matricielles [25] .

### 3.2.1 Représentation d'un réseau social

La première personne à avoir représenté un réseau social est Jacob Levy Moreno<sup>1</sup> au début des années 1930 [26]. Son objectif étant de visualiser graphiquement un réseau social. Il a représenté les personnes par des points (acteurs) et une relation entre deux personnes par des flèches. Cette représentation est depuis désignée par le terme sociogramme. Il peut aussi être connu sous le nom de diagramme de réseau social ou graphe. Vous trouverez un exemple d'un sociogramme à la Figure (suivante). Les mathématiciens ont rapidement fait le rapprochement entre les représentations sociogrammes et la théorie des graphes au sens mathématique.

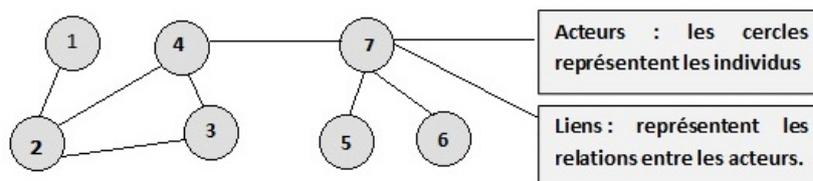


FIGURE 3.1 – Un exemple d'un sociogramme.

1. Moreno à partir d'une étude portant sur 506 jeunes filles d'un même pensionnat construisit un sociogramme des affinités ou au contraire des rejets pouvant être observées au sein du groupe

La représentation matricielle des graphes est une représentation beaucoup moins utilisée que les diagrammes nœuds-liens. On distingue deux types de matrices dans un RS, les matrices d'incidence et *les matrices d'adjacence* (figure suivante). On parle de matrice d'adjacence lorsqu'on a les mêmes ressources en ligne et en colonne. Les matrices d'incidence contiennent deux types de ressources, les lignes représentent un type et les colonnes un autre type.

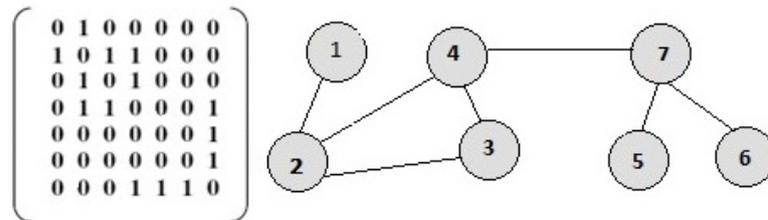


FIGURE 3.2 – Exemple de représentation matricielle du graphe (i.e. La matrice d'adjacence).

Dans la section qui suit nous évoquons les notions et terminologies relatives aux graphes nécessaires pour la modélisation des réseaux sociaux.

### 3.2.2 Graphes : Notions et terminologies

Les Graphes modélisent de nombreuses situations concrètes où interviennent des objets en interaction : Les interconnexions routières, ferroviaires ou aériennes entre différentes agglomérations, Les liens entre les composants d'un circuit électronique, le plan d'une ville et de ses rues en sens unique, Les définitions suivantes listent quelques notions manipulées par la théorie des graphes pour les réseaux sociaux [26] :

- Un **sommet** est l'unité de base d'un réseau, il en représente une ressource. Dans un réseau social on parle d'acteur. Le terme nœud est également utilisé pour désigner un sommet ;
- Une **arête** est une connexion entre deux sommets. On parle également d'arc ou de lien ;
- Une **hyperarête** est une arête qui connecte deux ou plusieurs sommets ;
- Une arête est **orientée** si elle ne s'utilise que dans une seule direction. Inversement, on parle d'arête *non orientée* pour une arête qui s'utilise dans les deux directions ;
- Une arête est **pondérée** lorsqu'on lui attribue un poids ;
- Une arête est **étiquetée** lorsqu'on lui attribue un label ;
- Un **graphe** est défini par un ensemble de sommets et un ensemble d'arêtes ;

- Un **hypergraphe** est défini par un ensemble de sommets et un ensemble d'hyperarête ;
- Un **graphe orienté** désigne un graphe avec des arêtes orientées ;
- Un **graphe pondéré** désigne un graphe avec des arêtes pondérées ;
- Un **graphe étiqueté** désigne un graphe avec des arêtes étiquetées ;
- Un **graphe multipartite** désigne un graphe avec des sommets de types différents ;
- Le **degré** d'un sommet est le nombre de ses arêtes adjacentes ;
- Un **chemin** est une séquence d'arêtes qui relie deux sommets ;
- Un **chemin orienté** est une séquence d'arêtes qui relie deux sommets en respectant l'orientation du parcours de chaque arête ;
- Une **géodésique** est l'un des plus courts chemins entre deux sommets donnés ;
- Le **diamètre d'un graphe** est le plus long chemin géodésique de ce graphe ;
- **Un graphe est complet** lorsqu'il existe une arête entre toute paire de sommets ;
- Un graphe est dit **connexe** lorsqu'il existe un chemin entre toute paire de sommets.

Nous utiliserons la notation suivante pour la suite de ce document :

- Nous notons un graphe  $G = (V, E)$  avec  $V$  l'ensemble des sommets,  $E$  l'ensemble des arêtes,  $n = |V|$  et le nombre de sommets et  $m = |E|$  et le nombre d'arêtes.
- Un sous graphe de  $G$  est noté  $G' = (V', E')$  avec  $V \subset V'$ ,  $E \subset E'$  et restreint à des arêtes reliant des sommets de  $V'$ ,  $n' = |V'|$  et  $m' = |E'|$ .
- $V_i$  désigne le  $i$ ème sommet.
- $(v_i, v_j)$  désigne une arête entre les sommets  $v_i$  et  $v_j$ .
- Le degré d'un sommet  $v_i$  est noté  $k_{v_i}$ .
- $d_{ij}$  représente la longueur d'une géodésique entre les sommets  $v_i$  et  $v_j$ . La moyenne des géodésiques est notée  $l$ .
- $\Gamma(v_i)$  est l'ensemble des voisins directes de  $v_i$ .

Les graphes non orientés sont adaptés pour les réseaux sociaux avec des relations non orientés. Les graphes orientés sont adaptés pour représenter des relations non symétriques comme les réseaux de confiance par exemple. Les graphes pondérés sont adaptés aux réseaux sociaux qui contiennent différents niveaux d'intensités dans les relations. Les graphes étiquetés permettent de représenter différents types de relations. Les graphes multipartites sont adaptés pour des réseaux sociaux incluant différent type de ressources manipulées par les acteurs et qui sont le support d'interactions.

### 3.3 Analyse des réseaux sociaux

comme le souligne l'auteur de [27], "ARS est définie comme étant l'étude des entités sociales ( les personnes dans les organisations qu'on appelle acteurs ) ainsi que leurs interactions et leurs relations. Ces interactions et relations peuvent être représentées par un graphe ou un réseau, dans lequel chaque nœud représente un acteur et chaque lien est une relation ". Plus généralement, l'analyse des réseaux sociaux pose l'importance des interactions entre individus, cette analyse dite structurale porte spécifiquement sur la description et l'analyse des différents modes de relations possibles (Figure 3.3) : interdépendance des membres, réciprocité ou non des relations ; place centrale de certains, absence de relations créant des "trou" relationnels au sein du réseau ; fréquence des relations ( i.e liens forts versus liens faibles). La force de l'analyse structurale réside dans sa capacité à représenter de façon simplifiée la complexité et la diversité des relations entre acteurs [28].

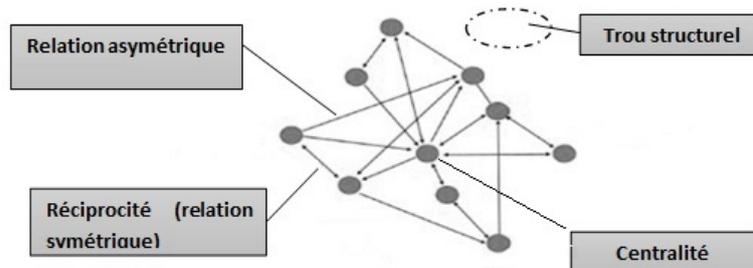


FIGURE 3.3 – Analyse structurale basée seulement sur les relations entre les nœuds

Selon Mercanti Guérin [28], *l'analyse des réseaux sociaux apparaît comme celle d'une conjonction progressive de problématiques, d'objets et de méthodes clairement distincts à l'origine, puisées à trois domaines : la sociométrie et plus largement la psychologie sociale, l'anthropologie, les mathématiques appliqués et en particulier la théorie des graphes et l'algèbre linéaire*. La théorie des graphes a été utilisée par l'ARS pour plusieurs raisons : premièrement, elle fournit un vocabulaire qui peut être employé pour étiqueter et désigner beaucoup de propriétés structurelles des relations sociales ; deuxièmement, elle offre des formulations mathématiques et des idées avec lesquelles beaucoup des propriétés relationnelles peuvent être quantitativement évaluées et mesurées ; troisièmement, elle permet de représenter les RS comme des modèles de jeux d'acteurs et relations [29]. Cette extrême simplification de la relation graphique constitue un avantage certain dans la représentation de réseaux simples<sup>2</sup>. Néanmoins, la

2. Il est à noter qu'un des principaux problèmes de la théorie des graphes est que le placement des sommets dans l'espace est arbitraire ce qui fait qu'un nombre infini de diagramme peut être dessiné bien qu'ils contiennent les mêmes données relationnelles.

complexité rencontrée dans la visualisation de bon nombre de réseaux a donné lieu à une modélisation par matrices [28].

Un petit réseau peut être visualisé directement par son graphe, quant au grand réseau, il serait plus difficile de l'imaginer ou de le décrire. Par conséquent nous définissons un ensemble de statistiques récapitulatives ou de mesures de performances quantitatives afin de décrire et de comparer les réseaux. De nombreuses mesures ont été proposées pour décrire quantitativement la structure des réseaux. D'une façon générale, ces mesures caractérisent les réseaux à la fois d'un point de vue local (voir Annexe A), mais également d'un point de vue global (voir Annexe B). Les mesures locales s'intéressent uniquement aux propriétés des nœuds et des liens, alors que les mesures globales considèrent l'ensemble du réseau à travers des propriétés statistiques calculées sur l'ensemble de la structure [30]. Finalement, qu'elles soient globales ou locales, toutes ces mesures permettent de caractériser les réseaux de façon statique, en fournissant des informations pertinentes sur l'état de leur structure topologique.

## 3.4 Analyse et fouille de données sociales

Les données sociales font référence à toutes les données qui représentent l'activité sur un réseau social. Deux grandes familles de méthodes d'analyse de ces données peuvent être distinguées. Les méthodes traditionnelles, qui s'appuient uniquement sur des propriétés structurelles locales ou globales pour caractériser les nœuds et la structure, et les méthodes d'extraction de connaissances qui appliquent les principes de la fouille de données aux réseaux sociaux [30].

### 3.4.1 Fouille de données sociales

Dans [31] L'ARS est défini comme une approche de la sociologie par la fouille de graphes. L'un des défis de la recherche dans le domaine de l'étude des réseaux est de proposer de nouveaux algorithmes, ou adaptations d'algorithmes existants, capables d'extraire efficacement de la connaissance à partir de données sociales. *Getoor* et *Diehl* dans [30] définissent la fouille de réseaux sociaux, comme " *l'ensemble des techniques de data Mining qui considèrent explicitement les liens lors de la construction de modèles descriptifs ou prédictifs à partir de données relationnelles* ". Ainsi, la fouille de données sociales aborde deux grandes tâches que nous détaillons ci-après : l'identification des acteurs importants, l'identification de groupes.

### 3.4.1.1 L'identification des acteurs importants :

L'identification des acteurs au sein d'un réseau social est un problème issu du fait que nous disposons d'un nombre important de données sur les liens entre les différents nœuds d'un graphe. Cette connaissance sur les relations entre les individus a amené les chercheurs à étendre les tâches de fouilles de données classiques puisqu'il est envisageable de faire des classifications à partir des liens que possède un nœud (et non plus à partir de ses attributs seuls).

Tout d'abord, Qu'est ce qui caractérise un sommet important ? Le qualificatif important a un grand nombre d'interprétations variées, conduisant à de nombreuses définitions différentes de la centralité. La table (suivante) présente quelques principales mesures proposées dans le domaine de l'analyse des réseaux sociaux.

Lors de l'étude de la notion de centralité, il est important de distinguer le cas des graphes orientés du cas des graphes non-orientés. Dans les graphes orientés, les nœuds possèdent deux types de liens, à savoir des liens entrants et des liens sortants. Pour une définition donnée de la notion de centralité, chaque nœud aura alors deux mesures d'importance : une mesure relative à ses liens sortants, appelée mesure de centralité, d'influence ou de centralité sortante, et une autre mesure relative à ses liens entrants, appelée mesure de prestige, de popularité, d'autorité ou de centralité entrante [32].

Indicateurs d'importance	Interprétation	Complexité [33]
Centralité de degré (voir Annexe 2)	Elle est basée sur l'idée que l'importance d'un individu au sein d'un groupe dépend du nombre total de personnes qu'il connaît ou avec lesquelles il interagit directement [34] .	$O(n)$
Centralité de proximité (voir Annexe 2)	Un nœud occupe une position stratégique dans un graphe s'il est globalement proche des autres nœuds de ce graphe. Par exemple dans un réseau social, cette mesure correspond à l'idée qu'un acteur est important s'il est capable de contacter facilement un grand nombre d'acteurs avec un minimum d'effort (l'effort ici est relatif à la taille des chemins).	$O(n \log(n) + m)$
Centralité d'intermédiarité (voir Annexe 2)	Un sommet ayant une forte centralité d'intermédiarité est un sommet par lequel passe un grand nombre de chemins géodésiques dans le graphe. Dans un réseau social, un acteur ayant une forte centralité d'intermédiarité est un sommet tel qu'un grand nombre d'interactions entre des sommets non adjacents dépend de lui. Dans un réseau de communication, la centralité d'intermédiarité d'un nœud peut être considérée comme la probabilité qu'une information transmise entre deux nœuds passe par ce nœud intermédiaire.	$O(nm + n^2 \log(n))$
Centralité spectrale (voir Annexe 2)	la centralité d'un nœud soit déterminée par la centralité des nœuds auxquels il est connecté. Dans un réseau social, cela correspond à l'idée qu'un acteur est d'autant plus important qu'il soit connecté à des acteurs qui sont eux même importants.	$O(n^2)$

TABLE 3.1 – Les indicateurs de centralités

Comme exemple d'application de la notion de centralité. Le classement des sites Web de façons à répondre à une requête d'un internaute avec le plus pertinence possible. Le *PageRank* (la popularité d'une page) est un système développé par les fondateurs de Google qui affecte une notation à une page web en fonction des liens externes pointant vers cette page et de la nature et qualité des sites sur lesquels ces liens sont présents. Combiné aux autres critères SEO<sup>3</sup>, le *PageRank* contribue au score global attribué à une page en fonction d'une requête donnée et donc à la position de la page dans les SERPs<sup>4</sup> Google.

3. Search Engine Optimization (SEO) consiste à optimiser un site web pour que le site apparaisse dans les premiers résultats sur des recherches bien précises ou des mots clés déterminés sur lesquels nous souhaitons apparaître. <http://www.indg.fr/SEO/criteres-google.html>

4. Search Engine Results Page (SERP) est anglais et se traduit littéralement par "Page de résultats"

### 3.4.1.2 L'identification de groupes, ou Clustering sur les nœuds

Fait référence à une famille de méthodes qui a pour objectif d'identifier des groupes de nœuds qui partagent des caractéristiques communes. Dans le contexte des réseaux, les groupes sont souvent définis comme des amas de nœuds densément connectés ; on parle également de communautés. Nous revenons sur ces méthodes au la suite de ce chapitre, car elles sont en relation avec la problématique que nous abordons. Le problème d'identification de communautés est important puisqu'il peut être rencontré dans plusieurs domaines d'application et des situations du monde réel. À titre d'exemple, les réseaux sociaux peuvent dévoiler des communautés représentant des individus ayant des intérêts communs ou de fortes connexions entre eux. De ce fait, on peut prédire la consommation ou le comportement des individus en analysant les achats et les comportements des autres éléments de la même communauté.

Nous allons voir dans ce suit ce qu'une communauté, et les différentes approches de détection de cette dernière.

## 3.5 Définition d'une communauté

Le mot communauté est défini selon Larousse par :

*" Etat, caractère de ce qui est commun a plusieurs personnes : Une communauté de biens, d'intérêts.*

*Ensemble de personnes unies par des liens d'intérêts, des habitudes communes, des opinions ou des caractères communs : Communauté ethnique, linguistiques."*

Le terme communauté désigne, comme son radical l'indique, un ensemble d'individus ayant une ou plusieurs caractéristiques (propriété, centre d'intérêt, etc.) en commun. Ce mot utilise dans plusieurs disciplines telles que l'informatique, la physique ou encore la biologie. En informatique, la notion de communauté est devenue dès la fin des années 90 très populaire avec le développement du web et de la recherche d'information basée sur les liens hypertextes. La définition la plus courante de la notion de communauté dans un graphe (ou réseau) est celle d'un ensemble de sommets ayant beaucoup de liens (i.e. une forte densité des liens) entre eux et peu de liens (i.e. une faible densité des liens) avec les autres nœuds du graphe [35]. Cette définition étant très générale, il est souvent nécessaire de l'affiner afin de pouvoir l'utiliser dans la pratique, comme par exemple pour vérifier si un sous-graphe correspond à une communauté ou encore pour mettre en œuvre un outil d'identification automatique de communautés. Malheureusement, il n'existe dans la littérature aucune définition formelle et universelle de ce qu'est une

---

d'un moteur de recherche". Cette page liste tous les résultats qui ont été obtenu par l'engin de recherche pour une requête particulière. <http://glossaire.infowebmaster.fr/serp>

communauté [36] , pour [37] ” *la notion de communautés ne peut être envisagée sans le but que l'on se donne, sans savoir pourquoi on cherche des communautés... Cette recherche de communautés est donc naturellement spécifique au contexte du graphe étudié, à la finesse de l'analyse souhaitée, même à l'expérimentateur qui devra choisir parmi toutes les méthodes de recherche de communautés et donc introduire un biais à ce niveau-là.* Dans [32] , les définitions de communauté sont regrouper en trois catégories, celles basées sur la connectivité des nœuds, celles basées sur la proximité des nœuds et autres basées sur une fonction de qualité.

### 3.5.1 Définitions basées sur la connectivité des sommets

Une des premières définitions de la notion de communauté a été proposée par Luce et Perry [38] dans le cadre de leurs travaux sur l'analyse des réseaux sociaux. Ils ont ainsi propose le terme clique (i.e. un ensemble de sommets deux-a-deux adjacents) pour faire référence à un groupe compose d'au moins trois personnes qui se connaissent toutes. La figure suivante illustre des exemples de cliques,  $C1$  et  $C3$  sont des cliques,  $C2$  n'est pas une clique car ses nœuds ne sont pas tous connectes.

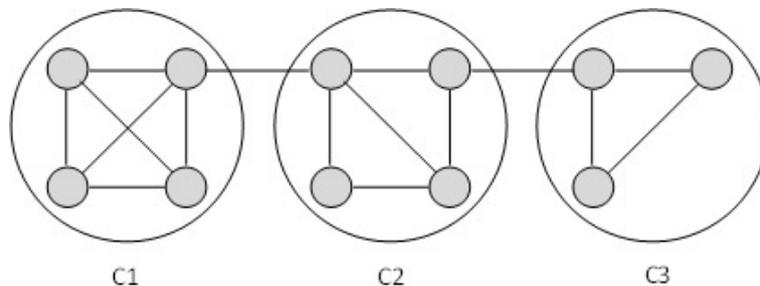


FIGURE 3.4 – Exemple d'un graphe contenant deux cliques ( $C1$  et  $C3$ ).

Bien que simple et théoriquement très intéressante, a définition d'une communauté en termes de cliques possède de nombreux inconvénients. Une de ses limites est que dans les graphes réels, il est très rare de rencontrer des cliques de grande taille, on trouve plutôt des triangles (i.e. le graphe  $C3$ ). Un autre problème avec cette définition est qu'elle est très exigeante dans la mesure où elle nécessite la présence de liens entre tous les sommets du sous-graphe ( i.e. le graphe  $C2$ ). Pour cette raison de nombreux chercheurs qui ont proposé des notions plus flexibles que celle de clique, comme :

**Une n-clique :** un ensemble maximal de nœuds tel que la distance entre chaque paire de nœuds est inférieure ou égale à  $n$  [39] [40], Une clique est une 1-clique avec cette définition.

**Un k-core :** un sous-graphe maximal tel que chaque nœud du sous-graphe soit adjacent à au moins  $k$  autres nœuds du sous-graphe [41].

**Des k-cliques percolées** : un ensemble maximal de cliques de taille k tel qu'il existe une série de cliques de taille k adjacentes sur k-1 sommets entre chaque paire de cliques de l'ensemble [42].

### 3.5.2 Définitions basées sur la similarité des sommets

Une autre définition de la notion de communauté est celle d'un groupe de sommets qui sont à la fois fortement similaires entre eux et faiblement similaires aux sommets appartenant aux autres groupes [36]. Cette définition, repose sur l'utilisation d'une mesure de proximité adaptée au type des objets à regrouper. En pratique, cette mesure de proximité peut correspondre soit à une mesure de similarité, soit à une mesure de distance.

#### 3.5.2.1 Mesure de proximité

Les mesures de proximité sont des fonctions prenant en entrée un graphe et deux nœuds et retournant une quantité permettant d'évaluer combien ces deux nœuds sont proches l'un de l'autre. La littérature sur le Clustering contient un grand nombre de travaux sur les mesures de similarité et de distance. Nous présentons ici celles qui sont les plus utilisées. Dans le cas où les objets à comparer peuvent être représentés sous forme de vecteurs caractéristiques à n dimensions sur un espace euclidien, on peut alors utiliser des mesures de proximité entre vecteurs telles que la distance euclidienne définies respectivement par [43] :

$$d_{euc}(v_i, v_j) = \sqrt{\sum_{k=1}^n (v_{i_k} - v_{j_k})^2}$$

Si l'on considère par contre que chaque objet est représenté par l'ensemble de ses attributs (par exemple un sommet sera représenté par l'ensemble des sommets qui lui sont adjacents), le calcul de la proximité entre deux objets peut dans ce cas être fait en utilisant des mesures telles que l'indice ou bien similarité de Jaccard défini par [43] :

$$S_{jac}(v_i, v_j) = \frac{|\Gamma(v_i) \cap \Gamma(v_j)|}{|\Gamma(v_i) \cup \Gamma(v_j)|} \quad (3.1)$$

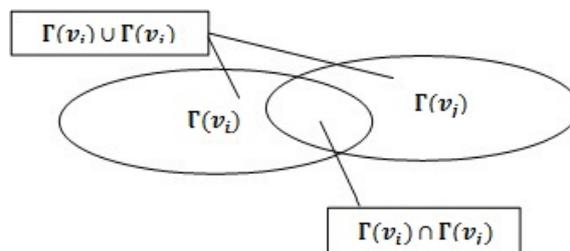


FIGURE 3.5 – Indice de Jaccard

Comme illustre la figure l'indice de Jaccard est le rapport entre le cardinal de l'intersection des ensembles considérés et le cardinal de l'union des ensembles. Il permet d'évaluer la similarité entre les ensembles.

### 3.5.3 Définitions basées sur une fonction de qualité

Une définition plus récente de la notion de communauté est basée sur l'utilisation d'une fonction de qualité. Cette famille de définitions a d'ailleurs reçu l'attention d'un grand nombre de chercheurs depuis la publication en 2004 d'un article par Newman et Girvan [44]. Suite à cet article, plusieurs fonctions de qualité ont été proposées par les chercheurs. La fonction la plus connue celle introduite par Newman sous le nom de la modularité (voir Annexe C).

## 3.6 Détection de communautés

La détection de communautés a pour but d'identifier toutes les communautés présentes dans un graphe donné. Elle nous permet également de déterminer le rôle de différents acteurs au sein des communautés et dans le réseau dans sa globalité. Le nœud avec une position centrale dans sa communauté, comme un nœud qui partage un grand nombre de liens avec les autres nœuds de la même communauté, peut exercer un important rôle de contrôle. Notons que les nœuds situés sur les frontières entre les communautés jouent un rôle important de médiation et de contrôle de communications entre les communautés.

La détection de communautés dans les réseaux est aujourd'hui un domaine ayant donné lieu à une abondante littérature. Depuis les travaux de Girvan et Newman en 2002, des centaines de travaux ont été menés sur le sujet, notamment la proposition d'un nombre important de méthodes de plus en plus élaborées, ces algorithmes peuvent être classés en plusieurs catégories. Diverses classifications ont été proposées dans la littérature [45], nous citons ci-après quatre classes. La première contient les méthodes de classification hiérarchiques. La deuxième catégorie quant à elle englobe les méthodes d'optimisation d'une fonction objective, qui identifient les communautés en maximisant une fonction de qualité. Plus une catégorie porte sur les méthodes à base de modèle. Finalement, la dernière repose sur les approches centrées graines.

### 3.6.1 Les méthodes de classification hiérarchique

Une structure hiérarchique de communautés représentée sous une forme arborescente appelée dendrogramme. Les feuilles sont les communautés avec un seul nœud et la

racine représente le graphe entier.

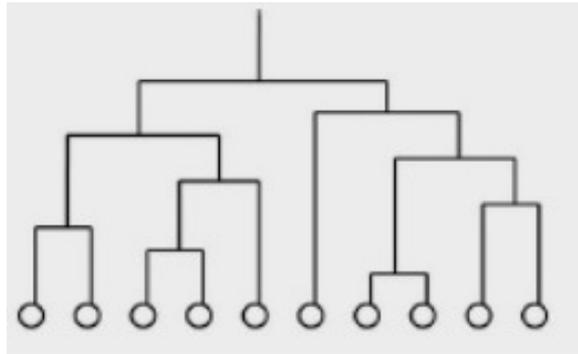


FIGURE 3.6 – Exemple de dendrogramme [46]

Les méthodes hiérarchiques peuvent être divisées en deux : les méthodes hiérarchiques agglomératives (ascendantes) et les méthodes hiérarchiques divisives (descendantes). Les deux méthodes sont basées sur la définition d'une mesure de similarité entre chaque paire de nœuds.

### 3.6.1.1 Les algorithmes agglomératifs

L'approche agglomérative part d'une structure dans laquelle chaque nœud du graphe représente une communauté. Nous avons initialement  $n$  communautés (où  $n = |V|$ ). On commence par calculer les distances entre les communautés et fusionner les deux communautés les plus proches pour former une nouvelle communauté. À chaque étape, on recalcule toutes les distances entre les communautés et on fusionne deux communautés. Lorsqu'il n'y a qu'une seule communauté représentant le graphe entier, il n'existe plus de distance à calculer. Dans ce qui suit, nous allons nous baser sur l'algorithme Walktrap [47] comme exemple d'algorithme agglomératif. Au début de l'algorithme, on a une partition  $P_0 = \{\{v_i\}, v_i \in V, i = 1 \dots n\}$  du graphe  $G$  qui représente  $n$  communautés. Chaque communauté contient un seul nœud. La première étape consiste à calculer toutes les distances entre les  $n$  communautés. Ensuite, on répète les étapes suivantes jusqu'à l'obtention d'une seule communauté :

**Étape 1** : sélectionner les deux communautés  $C1$  et  $C2$  de la partition  $P_k$  qui ont une distance minimale entre elles.

**Étape 2** : fusionner les deux communautés en une seule communauté  $C3 = C1 \cup C2$  et créer un nouveau partitionnement  $P_{k+1} = (P_k \setminus \{C1, C2\}) \cup \{C3\}$ .

**Étape 3** : calculer les distances entre les communautés et aller à l'étape 1. Après  $n - 1$  itérations, l'algorithme a comme partition  $P_n = \{V\}$  représentant le graphe entier. Chaque fusion entre deux communautés engendre une nouvelle partition. Après

la dernière fusion, nous disposons du dendrogramme (Figure suivante) qui représente les différentes communautés obtenues après chaque itération.

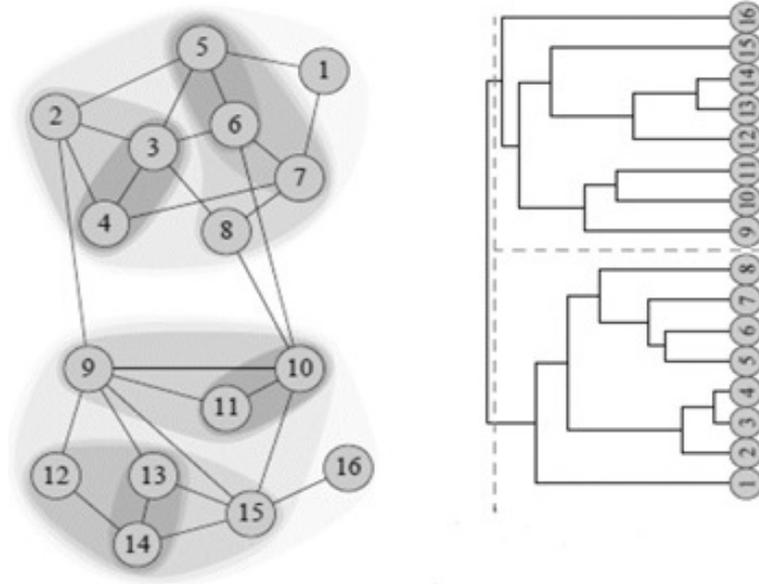


FIGURE 3.7 – Exemple de réseau à 16 sommets divisé en 2 communautés [47]

Le problème qui se pose est de choisir la bonne structure de communautés au cours des itérations. L'algorithme [48] fait appel à la modularité de Newman pour traiter le dendrogramme et dégager les communautés finales. La procédure *Walktrap* appelle la modularité pour chaque nouvelle structure de communautés (après chaque fusion). Une coupe du dendrogramme qui correspond à la valeur maximale de la modularité  $Q$  donne les communautés dégagées du graphe. La figure au-dessus, montre les résultats d'exécution de l'algorithme *Walktrap* avec un réseau de 16 nœuds. L'évolution de  $Q$  au cours des itérations de fusion montre qu'elle est maximale pour une valeur de 0,4. La projection de cette valeur sur le dendrogramme divise le réseau en deux communautés de huit nœuds chacune.

La complexité de *Walktrap* est de  $O(nmH)$  où  $H$  est la hauteur du dendrogramme. Dans le cas où les sommets sont fusionnés un par un,  $H$  atteint son maximum et vaut  $n - 1$ .

### 3.6.1.2 Les algorithmes divisifs

Les algorithmes divisifs consistent à scinder un réseau en plusieurs communautés en éliminant itérativement les liens entre les nœuds. Ils commencent par une seule communauté (le réseau entier), en haut du dendrogramme, jusqu'à avoir  $n$  communautés à un seul nœud représentant les feuilles du dendrogramme. Dans chaque itération, tout réseau connexe est considéré comme une communauté. Les méthodes existantes se dis-

tinguent par le choix des liens à éliminer et par les poids accordés aux liens. Nous retenons l'algorithme *Edge Betweenness* [44] comme exemple d'algorithme divisif. Les méthodes de détection de communautés s'intéressent généralement aux nœuds du réseau. Newman [44] s'est penché sur les liens plutôt que sur les nœuds. En effet, l'identification des liens se trouvant entre les communautés, ainsi que leur élimination permet d'identifier les différentes communautés dans un réseau. Afin de trouver les liens inter-communautés, l'algorithme *Edge Betweenness* accorde à chaque lien une mesure de centralité d'intermédiarité. En effet, un nœud intermédiaire est un nœud porteur de plusieurs chemins géodésiques reliant les différents couples de nœuds d'un réseau. Cette mesure de centralité peut être calculée pour les nœuds et les liens de la même façon. Néanmoins, le calcul de cette mesure est un processus trop lent car un parcours de tous les chemins possibles entre tous les couples de nœuds doit être fait pour chaque lien. La figure suivante illustre les mesures de centralité d'intermédiarité pour un exemple du réseau.

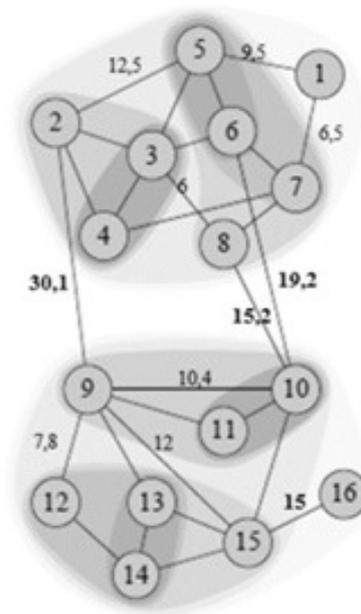


FIGURE 3.8 – Mesures de centralité d'intermédiarité d'un réseau.

Cette approche se base sur le principe qu'un lien se trouvant entre les communautés fait partie d'un très grand nombre de chemins géodésiques entre les différents nœuds. Il a de ce fait une grande valeur de centralité d'intermédiarité. Cependant, quelques liens se trouvant entre les communautés et qui sont plus longs que les chemins géodésiques, ont des  $C_b$  (centralités d'intermédiarité d'un lien) faibles. D'où l'idée d'éliminer, dans chaque itération, le lien le plus intermédiaire (i.e. ayant la plus grande valeur  $C_b$ ), puis recalculer à nouveau les  $C_b$  de tous les liens restants.

L'algorithme est constitué de quatre étapes principales :

1. Calculer les  $C_b$  pour tous les liens du graphe ;
2. Éliminer le lien le plus intermédiaire ;
3. Recalculer les  $C_b$  de tous les liens restants ;
4. Répéter les étapes 2 et 3 jusqu'à l'élimination de tous les liens.

La méthode commence par une seule communauté  $C$  contenant tous les nœuds du graphe. Après chaque itération, et s'il y a un nouveau sous-graphe qui n'est pas connexe au graphe initial,

de nouvelles communautés  $C1$  et  $C2$  se génèrent avec  $C = C1 \cup C2$ . Ce processus se répète jusqu'à l'obtention des feuilles du dendrogramme qui sont constituées par  $n$  communautés. Nous citons ici comme exemple le réseau social du club de karaté de Zachary<sup>5</sup> [49]. La figure donnée ci-dessous, montre le dendrogramme obtenu après l'exécution de l'algorithme avec ce réseau. Une meilleure structure de communautés, qui correspond à une valeur maximale de la modularité  $Q$  dans le dendrogramme, est représentée par cinq communautés. Si nous ignorons le nœud 10, les communautés trouvées contiennent 5, 10, 4 et 14 membres.

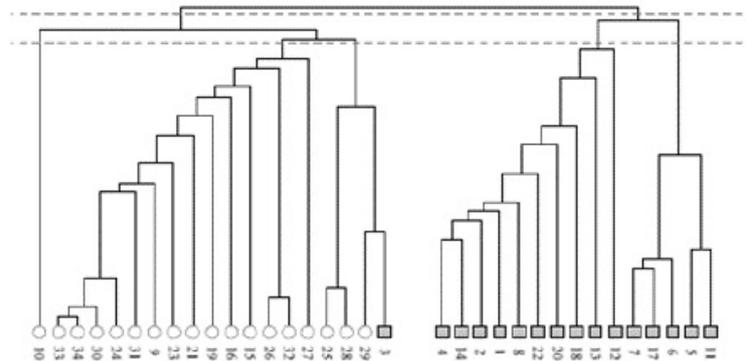


FIGURE 3.9 – Dendrogramme du réseau de Zachary [44]

Les résultats de cet algorithme prouvent sa performance pour la détection des communautés. Cependant, son temps de calcul est très élevé. La phase de calcul des degrés d'intermédiarité est trop lente. Ajoutons à cela qu'elle est exécutée à chaque itération. L'algorithme a une complexité de l'ordre de  $O(m^2n)$  [44].

5. Ce réseau représente les liens d'amitié entre 34 membres d'un club dans une université aux États-Unis.

### 3.6.2 Les méthodes d'optimisation d'une fonction objective

Il existe un certain nombre d'algorithmes se basant sur des heuristiques<sup>6</sup> pour définir la structure de communauté des réseaux. Ce type d'algorithme consiste à définir une fonction objective dont la valeur varie selon les communautés dégagées. La fonction est maximale pour la meilleure structure de communauté. Un exemple de ce type d'algorithme est Louvain.

La méthode Louvain [46] est une méthode simple, efficace et facile à mettre en œuvre pour identifier les communautés dans les grands réseaux. L'algorithme est composé de deux phases qui sont répétées de manière itérative jusqu'à obtenir un maximum local de la modularité.

**Première phase** : Initialement, une communauté différente est attribuée à chaque sommet du réseau. Dans cette partition initiale il y a de nombreuses communautés que de sommets. Ensuite, pour chaque sommet  $v_i$  tour à tour, les voisins  $\Gamma(v_i)$  de  $v_i$  sont considérés et le gain de modularité obtenu en retirant le sommet  $v_i$  de sa communauté et en le plaçant dans la communauté de ces voisins est évalué. Le sommet  $v_i$  est alors déplacé dans la communauté pour laquelle ce gain est positif et maximum, sauf si aucun gain positif n'est possible auquel cas le sommet  $v_i$  reste dans sa communauté d'origine. Ce processus est appliqué de façon séquentielle pour tous les sommets à plusieurs reprises jusqu'à ce qu'aucune amélioration supplémentaire ne puisse être réalisée.

**Deuxième phase** : La deuxième phase de l'algorithme consiste à construire un nouveau réseau dont les sommets sont les communautés trouvées lors de la première phase et les poids des liens entre les communautés sont donnés par la somme des poids des liens entre les sommets dans les deux communautés correspondantes. Les liens entre les sommets d'une même communauté induisent des boucles pour cette communauté dans le nouveau réseau dont le poids vaut deux fois le nombre de liens internes (voir l'agrégation de communautés sur la figure suivante).

Une fois cette deuxième phase terminée, il est alors possible de réappliquer l'algorithme à ce nouveau réseau pondéré. Une combinaison des deux phases est appelée une passe et celles-ci sont itérées jusqu'à ce qu'il n'y ait plus de changement et qu'un maximum de modularité soit atteint.

---

6. Une heuristique est une méthode de calcul qui fournit rapidement une solution réalisable, pas nécessairement optimale ou exacte.

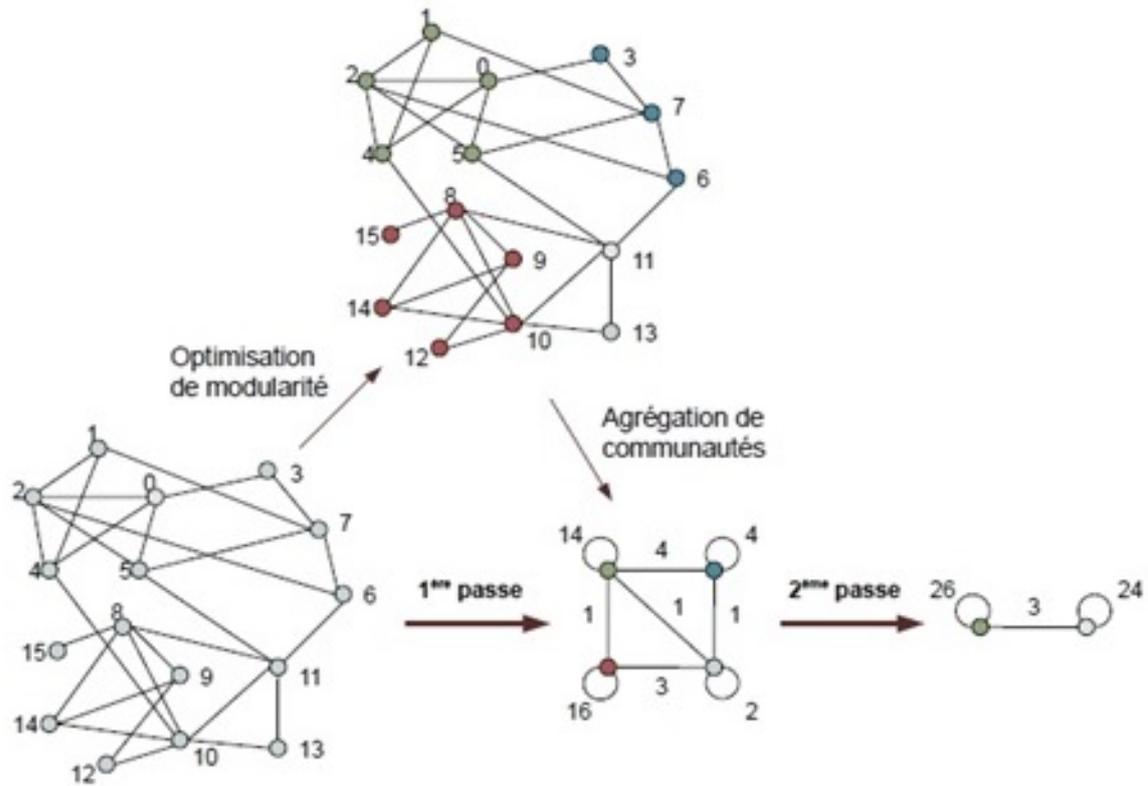


FIGURE 3.10 – Exemple d’application de la méthode de Louvain sur un graphe à 16 sommets [46]

### 3.6.3 Les méthodes à base du modèle

Les algorithmes à base du modèle sont des algorithmes de classification non supervisée utilisant des méthodes à base de prototypes exprimés dans un formalisme de modèles. Pour chaque type de données, un modèle d’apprentissage adapté à la nature des données traitées est proposé. Parmi ces algorithmes, Label Propagation décrit en [50]. Elle se base sur le principe qu’un nœud fait partie de la communauté qui contient le plus grand nombre de nœuds voisins. Ce processus est le modèle d’apprentissage pour Label Propagation, qui s’exécute sur tous les nœuds dans chaque itération. Au début de l’algorithme, chaque nœud se trouve dans une seule communauté. Puis, les nœuds changent de communautés tout en respectant le modèle d’apprentissage. Avec cette méthode, un groupe de nœuds, fortement liés entre eux, finit par se trouver dans la même communauté.

L’algorithme Label Propagation est décrit principalement par les étapes suivantes :

1. Initialement chaque nœud  $v_i$ , à l’itération  $t = 0$ , appartient à une seule communauté  $C_i : C_{(v_i)}(0) = i, i = 1 \dots n$ ;
2. Définir  $t = 1$ ;

3. Pour chaque nœud  $v_i$ , trouver la communauté du nœud  $v_i$  à la  $t$ ème itération  $v_i$  en fonction des communautés de voisins de  $v_i$  à l'itération  $t$  ou  $t - 1$ .
4. Si tous les nœuds ne changent plus de communautés ( $C_{v_i}(t) = C_{v_i}(t - 1)$  pour  $i = 1 \dots n$ ), l'algorithme s'arrête. Sinon,  $t = t + 1$  et aller à l'étape 3.

Le problème de cette méthode est non déterministe, puisque son exécution avec le même réseau peut donner plusieurs résultats comme le montre cette figure.

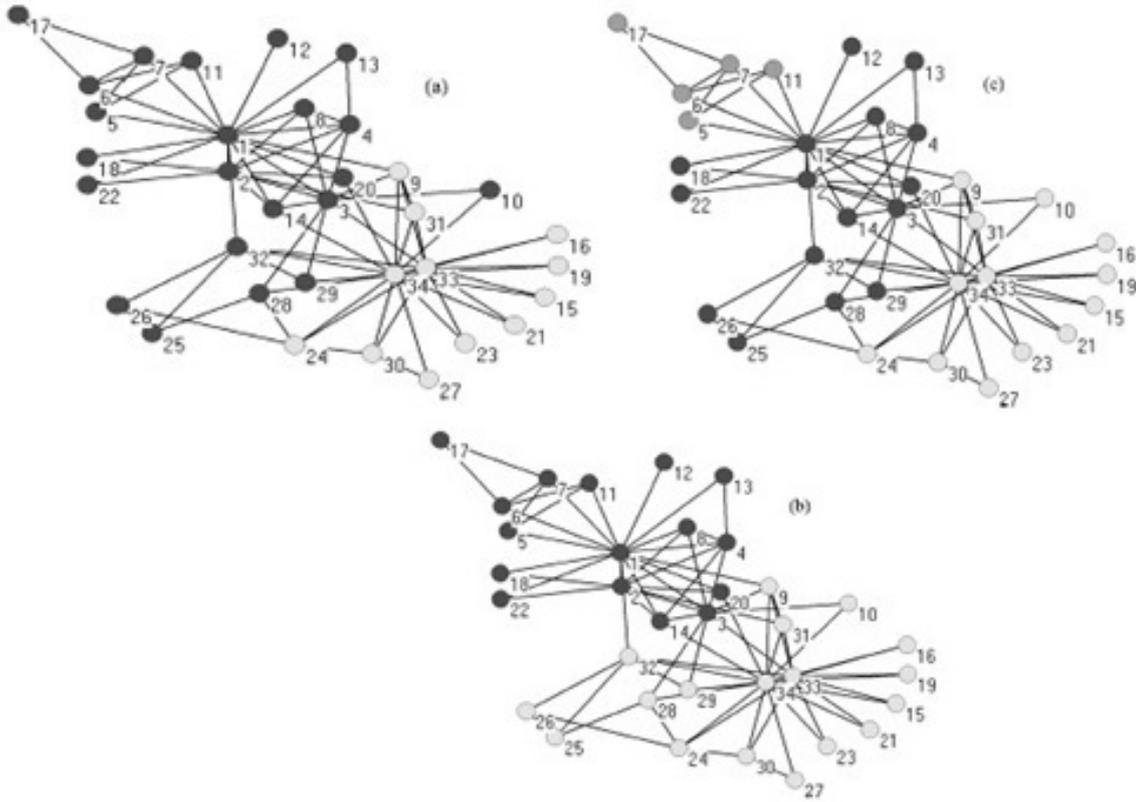


FIGURE 3.11 – Les différentes structures de communautés trouvées par Label Propagation pour le réseau du club de karaté.

### 3.6.4 Les méthodes centrées graines

Approches centrées graines où la structure communautaire est construite autour d'un ensemble de nœuds choisis d'une manière informée. Le schéma général d'une approche de cette catégorie est structuré en deux étapes :

- Déterminer un ensemble de nœuds ou groupes de nœuds dans le graphe qu'on désigne par des graines et qui constituent en quelque sorte les centres de communautés à retrouver.
- Appliquer une procédure d'expansion<sup>7</sup> autour des graines afin d'identifier les com-

7. Procédure d'expansion permet d'élargir les communautés autour des graines. Le but de cette procédure est d'itérer sur l'ensemble des graines afin de trouver la structure communautaire du réseau.

munautés dans le réseau.

Différentes heuristiques de choix de graine ont été proposées. Une graine peut être composée d'un seul nœud sélectionné en utilisant les mesures classiques de centralité. Dans d'autres algorithmes la graine est composée d'un ensemble de nœuds qui ont une certaine connectivité. Dans [51] une approche plus originale est proposée où après la détection de graines, chaque nœud dans le graphe (graine ou non) calcule un vecteur de préférence d'appartenance aux communautés de chaque graine. L'appartenance communautaire des nœuds est le résultat d'un processus de vote local impliquant le nœud et ses voisins directs.

LICOD [51] se résume en trois étapes :

1. Identification de l'ensemble des Leaders  $L$ . Un nœud est déclaré leader si sa centralité est supérieure à la centralité de la plupart de ses voisins. Différentes mesures de centralités peuvent être employées notamment la centralité de degré, la centralité d'intermédiarité et la centralité de proximité. Après,  $L$  est réduit en un ensemble  $C$  de communautés de Leaders. Deux leaders sont regroupés s'ils ont un nombre de voisins communs élevé ;
2. Chaque  $v_i \in V$  forme un vecteur de préférence  $P_{v_i}^0$  où les communautés identifiées dans  $C$  sont triées en ordre décroissant ;
3. Une fois chaque nœud a son vecteur de préférence, on commence une phase d'intégration où le vecteur de préférence d'un nœud est fusionné avec ceux de ses voisins directs. Ceci permet de favoriser la classe dominante dans l'ensemble des nœuds voisins.

## 3.7 Conclusion

Nous avons tout d'abord défini dans ce chapitre les réseaux sociaux et leurs différentes représentations. Aussi on a présenté l'ARS, l'analyse structurelle et fouille de données sociales.

Dans le chapitre qui suit , nous allons détailler l'approche adoptée à notre problématique.

# Chapitre 4

## L'approche adoptée

### Sommaire

---

<b>4.1</b>	<b>Introduction</b>	<b>52</b>
<b>4.2</b>	<b>L'approche Adoptée</b>	<b>52</b>
<b>4.3</b>	<b>Description de l'approche</b>	<b>52</b>
4.3.1	Identification des leaders	54
4.3.2	Regroupement des leaders	54
4.3.3	Calcul de vecteur d'appartenance	54
4.3.4	Fusion de votes	54
4.3.5	Paramètres	56
<b>4.4</b>	<b>Implémentation</b>	<b>56</b>
<b>4.5</b>	<b>Benchmark utilisé</b>	<b>57</b>

---

*” On ne peut pas vivre dans une communauté  
en se tenant complètement à l'écart ”*

[Franz Werfel]

## 4.1 Introduction

Ce chapitre présente la méthode que nous la voyons comme solution pour la détection de communautés cachées pour l'amélioration du processus de découverte des services web. Cette approche appartient à la classe des approches centrées graines. La catégorie centrée graine permet de construire une communauté autour d'un nœud choisi d'une manière informée. Nous allons voir dans ce suit, une description détaillée sur l'approche choisi.

## 4.2 L'approche Adoptée

Nous choisissons de travailler avec la méthode LICOD (Leaders Identification for COmmunity Detection in Complex Networks) [51] propose par *Rushed Kanawati* en 2011. Cette méthode été défini pour des graphes non-orientés et non-pondérés concerne les réseaux complexes (i.e. de grande taille). Mais que, le type de graphe qu'on va étudier orienté et pondéré. On prend, par exemple le réseau de substitution. Soit un service 1 qui peut remplacer un éventuel service 2 avec un certain degré, cette valeur correspond à l'accumulation d'interaction avec le service 2; cependant, le degré de la relation entre le service 2 vers le service 1 peut être différent que le degré de la relation entre service 1 vers le service 2. Comme nous l'avons dit précédemment, nous proposons une adaptation de LICOD pour ce type de graphe avec une nouvelle mesure de centralité.

## 4.3 Description de l'approche

Nous présentons ci-dessous l'approche LICOD

1. LICOD commence par identifier les nœuds qui semblent être leaders. Plusieurs heuristiques peuvent être proposées pour estimer le rôle d'un nœud. Soit  $L$  l'ensemble des leaders identifiés;
2. Afin de faire face à un éventuel grand nombre de leaders, ce qui va affecter directement le nombre de communauté final, LICOD intègre une étape de regroupement des leaders estimés être dans la même communauté. Chaque groupe de leaders représente une communauté. Notons  $C$  l'ensemble de communautés identifiées;
3. LICOD adopte ensuite une construction agglomérative des communautés. Chaque nœud dans le réseau, qu'il soit leader ou non, calcule son vecteur d'appartenance aux communautés dans  $C$ . Puis, selon ce vecteur, il trie les communautés dans un ordre décroissant;

4. Une fois que chaque nœud a son vecteur d'appartenance, on commence une phase d'intégration où le vecteur d'appartenance d'un nœud est fusionné avec ceux de ces voisins directs. Ceci permet de favoriser la classe dominante d'un nœud dans l'ensemble des nœuds voisins. Des algorithmes de choix social sont utilisés dans cette phase de fusion de votes. Ce processus est répété jusqu'à la stabilisation du vecteur d'appartenance de chaque nœud ;
5. A la stabilisation, chaque nœud est affecté à la communauté placée en tête de vecteur d'appartenance.

---

**Algorithme : LICOD**

---

**Entrée :** un graphe  $G = \langle V, E \rangle$   
**Sortie :** une structure communautaire

**Début**  
 $\mathcal{L} = \emptyset$   
**Pour**  $v_i \in V$  **faire**  
     **Si**  $\text{EstLeaders}(v_i)$  **alors**  
          $\mathcal{L} \leftarrow \mathcal{L} \cup \{v_i\}$   
     **Fin si**  
**Fin pour**  
 $C \leftarrow \text{StructureCommunautaire}(\mathcal{L})$   
**Pour**  $v_i \in V$  **faire**  
     **Pour**  $c_j \in C$  **faire**  
          $M[v_i, c_j] \leftarrow \text{Appartenance}(v_i, c_j)$   
     **Fin pour**  
      $P[v_i] \leftarrow \text{TrierEtClasser}(M[v_i])$   
**Fin pour**  
**Répéter**  
     **Pour**  $v_i \in V$  **faire**  
          $P^*[v_i] \leftarrow \text{FUSION}_{\text{vxel}(\Gamma(v_i))} P[x]$   
          $P[v_i] \leftarrow P^*[v_i]$   
     **Fin pour**  
**Jusqu'à** Stabilisation de  $P^*[v_i] \forall v_i$   
**Pour**  $v_i \in V$  **faire**  
     **Pour**  $c_j \in P[v_i]$  **faire**  
         **Si**  $|M[v_i, c_j] - M[v_i, P[0]]| \leq \epsilon$  **alors**  
              $\text{COM}(c_j) \leftarrow \text{COM}(c_j) \cup \{v_i\}$   
         **Fin si**  
     **Fin pour**  
**Fin pour**  
**Return**  $C$   
**Fin**

---

FIGURE 4.1 – Algorithme de LICOD

Par la suite, nous énumérons les mesures appropriées pour chacune de ces étapes.

### 4.3.1 Identification des leaders

L'identification de nœuds leaders est une étape primordiale pour cette approche. Nous utilisons les mesures de centralité, elles sont des fonctions qui attribuent pour chaque nœud une valeur positive indiquant à quel point il est central. Les nœuds leader sont censés avoir des centralités les plus élevées que les nœuds non-leaders. Nous proposons d'appliquer les trois mesures suivantes :

- Centralité spectrale (voir la table 3.1) ;
- Centralité d'intermédiarité (voir la table 3.1 et pour l'algorithme voir [52]) ;
- Notre mesure de centralité ( $C_{new}$ ).

La mesure proposée c'est une combinaison entre la position stratégique du nœud dans le réseau et l'influence de ce nœud sur leurs voisins directs. Un nœud occupe une position stratégique dans un graphe s'il est capable de contacter facilement un grand nombre d'acteurs avec un minimum d'effort. Nous proposons l'équation suivante :

$$C_{new}(v_i) = C_c(v_i) \times k(v_i) \quad (4.1)$$

### 4.3.2 Regroupement des leaders

Dans cette phase, nous nous intéressons à regrouper les leaders qui partagent le plus de voisins en commun. Chaque groupe de leaders définira une communauté. Soient  $l_i$  et  $l_j$  deux leaders, ils seront regroupés dans la même communauté si  $S_{jac}(l_i, l_j) > \sigma \in [0, 1]$  (voir l'équation 3.1). Le regroupement de leaders permet d'éviter un éventuel grand nombre de communautés.

### 4.3.3 Calcul de vecteur d'appartenance

Après l'identification des nœuds leaders de chaque communauté, chaque nœud  $v_i \in V$  calcule son degré d'appartenance à chaque communauté  $C_j$ . Nous proposons d'utiliser une mesure basée sur la distance entre les nœuds (i.e. le plus court chemin). Tant que le type de graphe étudié est orienté et pondéré, nous proposons de restructuré notre graphe juste pour cette étape par cette formule  $T = 1/2(W + W^t)$ , et après nous calculons les plus courts chemins (SP) entre chaque leader et le reste des nœuds.

### 4.3.4 Fusion de votes

L'étape de fusion de votes consiste à fusionner le vecteur d'appartenance de chaque nœud avec les vecteurs de ses voisins. L'influence des voisins permet de favoriser la communauté majoritaire et d'avoir plus d'homogénéité locale. Et comme nous travaillons

sur un graphe pondéré, nous voyons que l'influence de chaque voisin se diffère selon le poids de relation, et pour cela, nous proposons l'équation suivante :

$$P[n] = \frac{1}{k_n} \times \sum_{i=1}^{|\Gamma(n)|} P[v_i] \times (W(n, v_i) + W(v_i, n)) \quad (4.2)$$

Différentes méthodes de fusion peuvent être appliquées. Dans LICOD, nous avons choisi d'utiliser l'algorithme issu de la théorie de choix social<sup>1</sup>. Plus précisément nous appliquons la méthode de Borda [53].

#### 4.3.4.1 Méthode de Borda

Borda était en faveur d'une procédure par score. Chaque nœud classe les  $k$  communautés selon sa préférence. On attribue alors le score  $k - 1$  à la communauté classée en premier, le score  $k - 2$  à la communauté classée deuxième ...etc. et le score 0 à la dernière communauté. On additionne alors le nombre de points obtenus par les communautés dans les classements des nœuds.

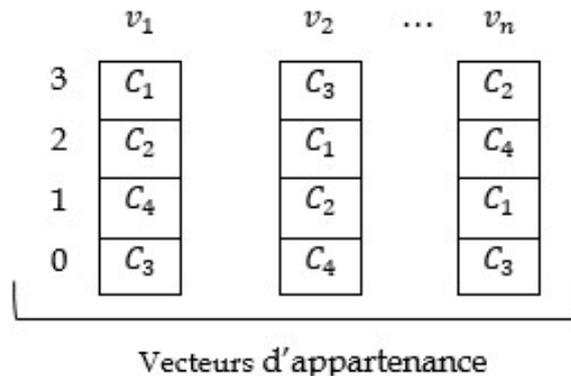


FIGURE 4.2 – Vecteurs d'appartenance des nœuds avec un exemple de 4 communautés

#### 4.3.4.2 Affectation des nœuds aux communautés

A la fin, chaque nœud  $v_i$  est associé à la communauté ayant le meilleur score dans son vecteur d'appartenance trié. Le seuil  $\epsilon$  contrôle le degré de chevauchement souhaité dans les communautés identifiées.

1. Le choix social a pour objet la sélection d'options par un groupe d'individus.

### 4.3.5 Paramètres

Notre approche a quatre paramètres :

- La mesure de centralité = [Centralité spectrale, Centralité d’intermédiarité, Notre mesure de centralité];
- Le seuil de leader : prend une valeur entre 0 et 1, une petite valeur de ce seuil permet de générer beaucoup de communautés.
- Le seuil de similitude : prend une valeur entre 0 et 1, permet de regrouper les leaders qui partagent le plus de voisins en communs, si ce seuil est fixé à 1 alors les leaders ne partagent pas des voisins en communs;
- Le degré de chevauchement  $\epsilon$  prend une valeur entre 0 et 1.

## 4.4 Implémentation

Pour implémenter notre approche, nous utilisons le logiciel open source *Gephi*. Est un logiciel d’analyse et de visualisation de réseaux sociaux (Voir Annexe D), développé en Java et basé sur la plateforme *NetBeans*. Il a été initialement développé par des étudiants de l’Université de technologie de Compiègne (UTC) en France. La première version née en 31/07/2008.



FIGURE 4.3 – Gephi logo (<https://gephi.org/>)

Comme on a mentionné précédemment, *Gephi* est open source d’une architecture logicielle modulaire<sup>2</sup>. Cette architecture permet d’ajouter des nouvelles fonctionnalités à base des plug-ins<sup>3</sup>. Pour *Gephi*, il existe plusieurs catégories de plug-ins comme : *Exports*, *Filter*, *Generator*, *Imports*, *Layouts* et *Metrics*.

Nous avons développés une nouvelle extension de type *Metrics*, appelée *Detection community*.

2. Reprend l’idée de fabriquer un logiciel à partir de composants.

3. Le terme plug-in vient du verbe anglais to plug in qui signifie brancher.

## 4.5 Benchmark utilisé

A cause de manque d'un corpus de domaine des services web sociaux, nous avons choisi de travailler sur le corpus de référence "*Neural network of the nematode C. Elegans*<sup>4</sup>".

*Caenorhabditis elegans* est un petit ver qui, entre autre du fait de sa totale transparence, a été activement étudié. *C. elegans* dispose d'un petit réseau de 297 neurones dont on connaît complètement la connectivité . L'information neuronale étant, au niveau synaptique, transférée d'un neurone à l'autre de manière unidirectionnelle, ce réseau est orienté et pondéré.

Diamètre	Densité	Coefficient de clustering	Nombre de nœuds	Nombre de liens	Type de graphe
14	0,027	0,169	297	2345	Orienté et pondéré

TABLE 4.1 – Les caractéristiques topologiques de réseau Celegans neural

4. Disponible pour le téléchargement dans ce lien <http://www-personal.umich.edu/mejn/netdata/>

---

## Conclusion générale

Dans ce mémoire, nous avons essayé de proposer une solution pour le problème le plus populaire dans le domaine des services web qui est de trouver une meilleure approche pour la découverte des services les plus pertinents pour une éventuelle requête en se focalisant sur les relations et la trace des interactions qui marque le comportement social entre les services eux-mêmes ou leurs utilisateurs. Nous avons parlé des travaux de Zakaria Maamar qui ont donné naissance à un nouveau concept appelé Service web sociaux. ce concept est né à l'intersection de deux domaines : l'approche social et les services web.

Dans le cadre de ce travail, nous avons pu développer une solution implémentant une approche de découverte des communautés en se basant sur la méthode LICOD (Leaders Identification for COmmunity Detection in complex networks) , cette méthode nous a permis de se rapprocher de notre but premier et réduire la complexité du problème qui a été connu et défini comme un problème de segmentation des nœuds que grâce à cette méthode, nous avons pu réduire sa complexité pour le transformer en un problème de classification et ce selon le profil des nœuds leaders. Nous avons justifié notre choix pour cette méthode, vu ses performances élevées et le principe de chevauchement préservé et qui est un concept essentiel du comportement social des services web.

Comme perspectives, nous projetons de valider notre approche et de comparer les résultats offerts par notre solution avec ceux qui existe déjà, donc de l'appliquer sur un corpus réel que nous n'avons pas pu le procurer. Ces résultats vont être issus de plusieurs mesures de centralité et d'en déduire la meilleure. Ce travail ne constitue qu'un début, vu la complexité du problème et sa popularité, après avoir validé cette approche, la prochaine étape va porter sur comment faire aiguiller la requête et la diffuser la communauté choisie.

---

## .1 Annexe A : Mesures locales

Il existe une multitude de mesures pour caractériser localement les réseaux. Ce type de mesures a pour objectif d'apporter des informations sur le voisinage d'un nœud ou de mettre en évidence certaines propriétés structurelles. Nous présentons les principales mesures utilisées localement, et pour chacune d'entre elles nous montrons son intérêt potentiel pour l'analyse des réseaux.

### - Le degré

Le degré d'un nœud  $v_i$  dans un réseau  $G = (V, E)$  est le nombre de liaisons connectées à  $v_i$ . Il est noté  $k_{v_i}$ , défini par :

$$k_{v_i} = \sum_{j=1}^n a_{ij} \text{ avec } a_{ij} \begin{cases} 1 & \text{si } v_i, v_j \in E \\ 0 & \text{sinon} \end{cases}$$

Dans un réseau orienté, on distingue généralement le degré entrant  $k^{in}$  du degré sortant  $k^{out}$ . Le degré d'un nœud apporte essentiellement une information sur la connectivité d'un nœud dans le réseau et permet de déterminer son rôle.

### - La centralité locale

La centralité local  $C^{deg}(v_i)$  d'un nœud  $v_i$  est le degré  $k_{v_i}$  du nœud  $v_i$ , normalisé par le degré maximal potentiel, c.à.d.  $(|V| - 1)$ . Autrement dit, c'est le pourcentage de nœuds avec lesquels le nœud  $v_i$  est connecté.

$$C^{deg}(v_i) = \frac{k_{v_i}}{|V|-1}$$

Elle s'appelle aussi centralité de degré, représente la forme la plus simple et la plus intuitive de la notion de centralité.

### - La distance

La distance  $d(v_i, v_j)$  d'un nœud  $v_i$  à un nœud  $v_j$  est la taille du plus court chemin connectant le nœud  $v_i$  au nœud  $v_j$ . Autrement dit, c'est le plus petit nombre de liaisons nécessaires pour joindre ces deux nœuds.  $d(v_i, v_j)$  est également appelée distance géodésique. Cette mesure fournit une indication locale entre deux nœuds et est utilisée dans le calcul d'autres mesures.

---

## - Le coefficient de Clustering

Le coefficient de Clustering<sup>5</sup> d'un nœud  $v_i$ , noté  $C_{v_i}$ , est la probabilité que deux voisins  $v_j$  et  $v_k$  du nœud  $v_i$  soit eux-mêmes voisins (le postulat : *l'ami de mon ami est mon ami*). Considérons une disposition géométrique des nœuds. On pose  $t_{v_i}$  le nombre de triangles dont le nœud  $v_i$  fait partie. Le coefficient de Clustering est donné par la formule :

$$C_{v_i} = \frac{2xt_{v_i}}{k_{v_i}x(t_{v_i}-1)}$$

D'une certaine façon, cette mesure peut être vue comme la densité locale d'un nœud. Elle permet par exemple de déterminer si un nœud est central ou périphérique. Elle est par exemple utilisée par de nombreux algorithmes d'affichage de réseaux.

---

5. Appelé aussi coefficient d'agglomération, de connexion, de regroupement, d'agrégation ou de transitivité.

---

## .2 Annexe B : Mesures globales

Contrairement aux mesures précédentes, les mesures globales décrivent l'ensemble de la structure en mettant en évidence certaines propriétés statistiques.

### - La densité

La densité  $p$  d'un réseau  $G = (V, E)$  est l'une des premières mesures utilisées pour caractériser la structure. Elle est égale au nombre de liaisons  $|E|$ , divisé par le nombre de liaisons possibles  $E_{max}$ . Pour un réseau non-orienté contenant  $|V|$  nœuds, le nombre de liaisons possibles est donné par  $E_{max} = \frac{1}{2} \times |V| \times (|V| - 1)$ .

$$p = \frac{|E|}{E_{max}} = \frac{2X|E|}{|V|X(|V|-1)}$$

La densité apporte une information sur la connectivité globale à l'intérieur du réseau.

### - Le degré moyen

Le degré moyen  $K$  d'un réseau  $G = (V, E)$ , correspond à la moyenne des degrés individuels  $C_{v_i}$  de chaque nœud  $v_i$ . Le degré moyen est souvent comparé aux degrés individuels, pour déterminer comment un nœud donné est connecté par rapport à la moyenne. Il apporte également une information globale sur la connectivité des nœuds.

### - La distance moyenne

La distance moyenne est une mesure individuelle de la distance qui sépare en moyenne un nœud  $v_i$  des autres ( $|V| - 1$ ) nœuds du réseau. Ainsi, soit  $d(v_i, v_j)$  la distance séparant le nœud  $v_i$  du nœud  $v_j$ , la distance moyenne  $L_{v_i}$  du nœud  $v_i$  est obtenue par :

$$L_{v_i} = \frac{1}{|V|-1} = \sum_{j=1, j \neq i}^{|V|} d(v_i, v_j)$$

Cette mesure fournit une information sur la proximité des nœuds dans le réseau ainsi que leur facilité à communiquer et échanger. Noter que la centralité de proximité (Closness centrality) définit par :

$$C_c = \frac{1}{L_{v_i}} = \frac{|V|-1}{\sum_{j=1, j \neq i}^{|V|} d(v_i, v_j)}$$

### - Le diamètre

Le diamètre  $Q$  d'un réseau est la plus grande des distances pouvant séparer deux nœuds au sein du réseau.

$$Q = \max d(v_i, v_j) \text{ avec } (v_i, v_j) \in VXV$$

Tout comme la distance moyenne, elle apporte une information sur la facilité qu'ont les nœuds du réseau à communiquer.

---

## - La centralité d'intermédiation

La centralité d'intermédiation (*Betweenness centrality*) mesure l'utilité d'un nœud dans la transmission de l'information au sein d'un réseau. Un nœud est d'autant plus central qu'il est situé sur beaucoup de plus courts chemins entre d'autres paires de nœuds. Elle est définie formellement par :

$$L_b(v_i) = \sum_{v_j, v_k \in V} \frac{\sigma(v_j, v_k | v_i)}{\sigma(v_j, v_k)}$$

Où  $\sigma(v_j, v_k)$  est le nombre des plus courts chemins liants  $v_j$  à  $v_k$ , et  $\sigma(v_j, v_k | v_i)$  est le nombre des plus courts chemins passant par  $v_i$ .

## - La centralité de vecteur propre

<sup>6</sup> L'idée que la centralité d'un nœud soit déterminée par la centralité des nœuds auxquels il est connecté. Dans un réseau social, cela correspond à l'idée qu'un acteur est d'autant plus important qu'il soit connecté à des acteurs qui sont eux même importants. Il s'agit en fait d'une extension de la centralité du degré dans laquelle on ne donne pas le même poids aux nœuds voisins. Pour la mise en œuvre de ce principe, *Bonacich* propose de considérer la centralité d'un nœud comme étant dépendante de la combinaison linéaire des centralités de ses nœuds voisins.

## - Le coefficient de Clustering moyen

Le coefficient de *Clustering* moyen  $C$  correspond à la moyenne des coefficients de *Clustering*  $C_{v_i}$  obtenus en chaque nœud  $v_i$ . Il apporte une information sur la tendance qu'ont les nœuds à former des amas densément connectés.

## - La distribution des degrés

La distribution des degrés  $P(k)$  est définie comme la probabilité qu'un nœud, choisi aléatoirement, ait un degré de  $k$ . Autrement dit,  $P(k)$  représente le pourcentage de nœuds dans  $G$  ayant  $k$  connexions.

$$P(k) = \frac{|\{v_i \in V; k_{v_i} = k\}|}{|V|}$$

Cette information est utilisée pour caractériser les réseaux et comprendre comment se répartit la connectivité au sein de la structure.

---

6. Appelée aussi centralité spectrale ou Eigenvector.

---

### .3 Annexe C : Modularités

Une structure de communautés dans un graphe  $G = (V, E)$  d'ordre  $n$  et de  $m$  de taille est un ensemble  $S = \{C_1, C_2, \dots, C_k\}$  tel que :  $\cup_{i=1}^k C_i = V$  et chaque  $C_i$  vérifie la définition de communauté considérée. Dans le cas où les ensembles  $C_i$  sont disjoints (i.e  $\forall i = 1, \dots, k, \forall j = 1, \dots, k, i \neq j \Rightarrow C_i \cap C_j = \emptyset$ ), la structure de communauté est aussi appelée partition. Si par contre ils ne sont pas disjoints, on dit qu'il y a recouvrement ou chevauchement des communautés de la structure

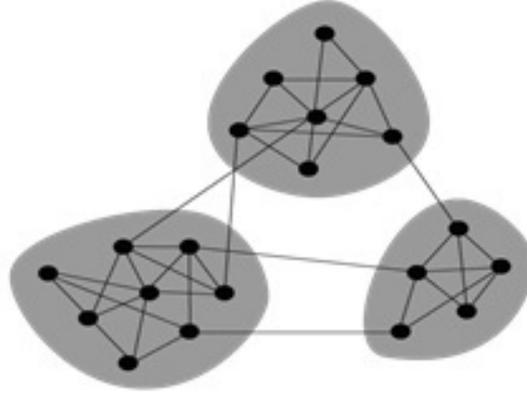


FIGURE 4 – Structure de communautés

La modularité d'un sous-graphe  $C_i = (V_{c_i}, E_{C_i})$  de  $G$  est définie par :

$$Q(c_i) = \frac{|E_{c_i}|}{m} - \left(\frac{d_{c_i}}{2m}\right)^2$$

Où  $d_{c_i}$  est égale à la somme des degrés des nœuds appartenant à  $c_i$ .

La modularité a été initialement proposée comme solution au problème suivant : « Étant données  $K$  partitions des sommets d'un graphe où chaque partition contient un nombre différent de communautés, quelle est la meilleure de ces  $K$  partitions ? ». Dans cette optique, la modularité associée à une partition  $P = (C_1, C_2, \dots, C_k)$  est égale à la somme de la modularité des communautés qui la composent :

$$Q(p) = \sum_{i=1}^k Q(c_i)$$

Cette formule peut également être réécrite de la manière suivante :

$$Q(p) = \frac{1}{2m} \sum_{i=1}^n \sum_{j=1}^n [a_{ij} - \frac{k_i k_j}{2m}] \delta(c_j, c_i)$$

Où  $c_i$  est la communauté à laquelle appartient le sommet  $i$ , la fonction delta de *Kronecker*  $\delta(c_j, c_i)$  est égale à 1 si  $c_i = c_j$  et à 0 sinon. Cette métrique est définie comme différence entre la proportion d'arêtes internes aux classes et cette même quantité pour une partition

---

aléatoire du graphe ayant des classes de même cardinalité; c'est l'écart entre ce qui est observé et ce qui est dû au hasard [54]. Les valeurs possibles pour la modularité sont dans l'intervalle  $]-1,1[$ . En cas d'absence de structure de communautés, la valeur de la modularité est négative ou nulle, alors qu'une valeur supérieure à 0.3 indique la présence d'une structure de communautés [44]. Considérons le graphe jouet de la Figure suivante et supposons que l'on souhaite comparer la qualité des partitions  $P1 = \{\{A, B, C, D\}, \{E, F, G, H\}\}$  et  $P2 = \{\{A, B\}, \{C, D, E, F, G, H\}\}$ . Le calcul des modularités de  $P1$  et  $P2$  indique que  $Q(P1) = 0.42$  et  $Q(P2) = 0$ . Ce résultat signifie que la partition  $P1$  est meilleure que  $P2$  et que cette dernière ne correspond pas à une structure de communautés.

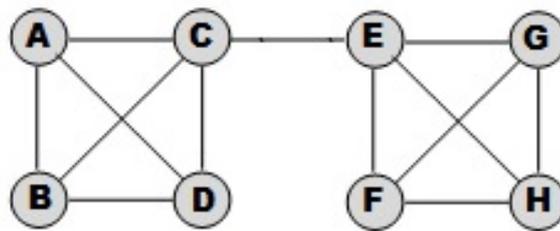


FIGURE 5 – Exemple d'un graphe jouet

La modularité semblait alors être une réponse très prometteuse à problème de sélection de la meilleure partition jusqu'à ce que *Fortunato* et *Barthelemy* [55] remettent en cause ce principe de sélection en se basant sur la modularité. Ils montrent en effet que la modularité favorise les partitions en communautés de grande taille. Autrement dit, cela signifie que si le graphe contient plusieurs communautés et que certaines de ces communautés sont de petite taille, la modularité tend à favoriser les partitions où les petites communautés ont été regroupées. *Fortunato* et *Barthelemy* montrent plus précisément que pour un graphe contenant  $m$  liens, la modularité favorise les partitions composées de communautés ayant au moins  $\sqrt{m/2}$  liens, même si le graphe contient en réalité des communautés ayant un nombre de liens inférieur à ça.

## 4 Annexe D : Fiche technique sur Gephi

Gephi est organisé en trois onglets (en haut, sous la barre de menus) :

- l'onglet laboratoire de données permet de voir et de modifier les nœuds et les arcs du graphe ainsi que leurs propriétés ;

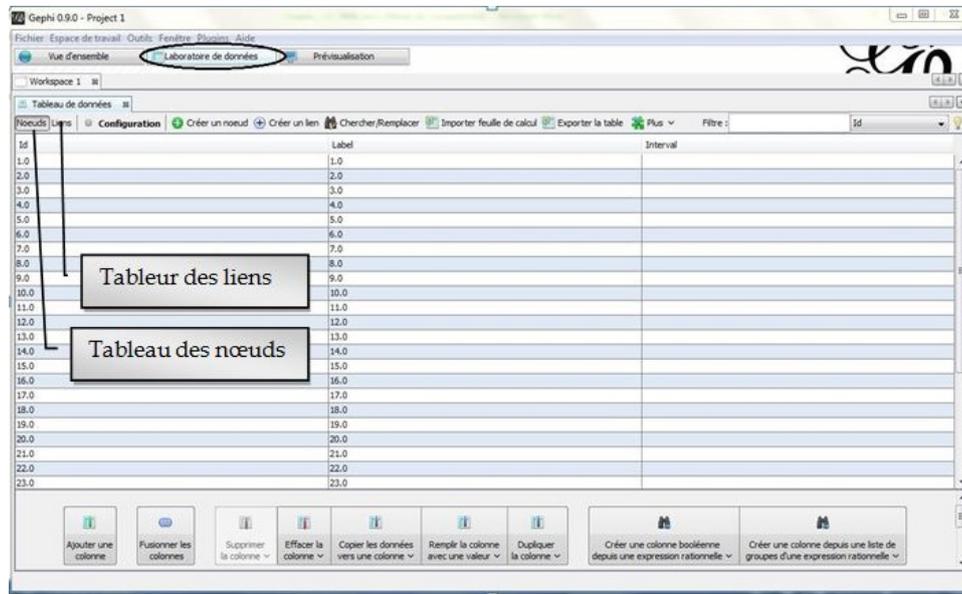


FIGURE 6 – L'onglet laboratoire de donnée

- l'onglet Vue d'ensemble permet d'effectuer l'analyse et la plus grande partie de la mise en forme du graphe (application d'un algorithme de positionnement, couleur et taille des nœuds et des arcs...);

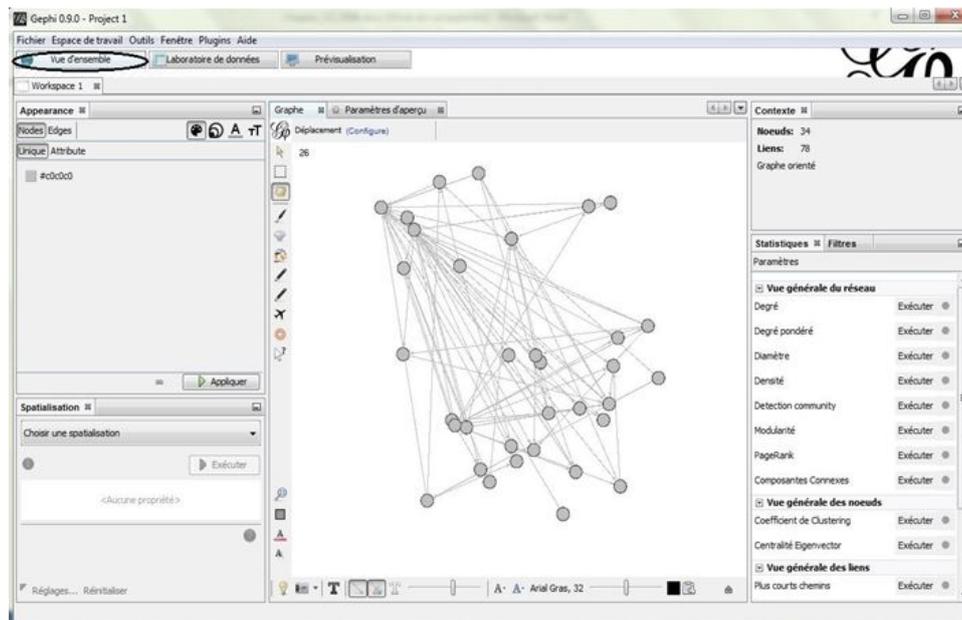


FIGURE 7 – L'onglet Vue d'ensemble

- l'onglet prévisualisation permet de peaufiner la visualisation et de générer une belle image.

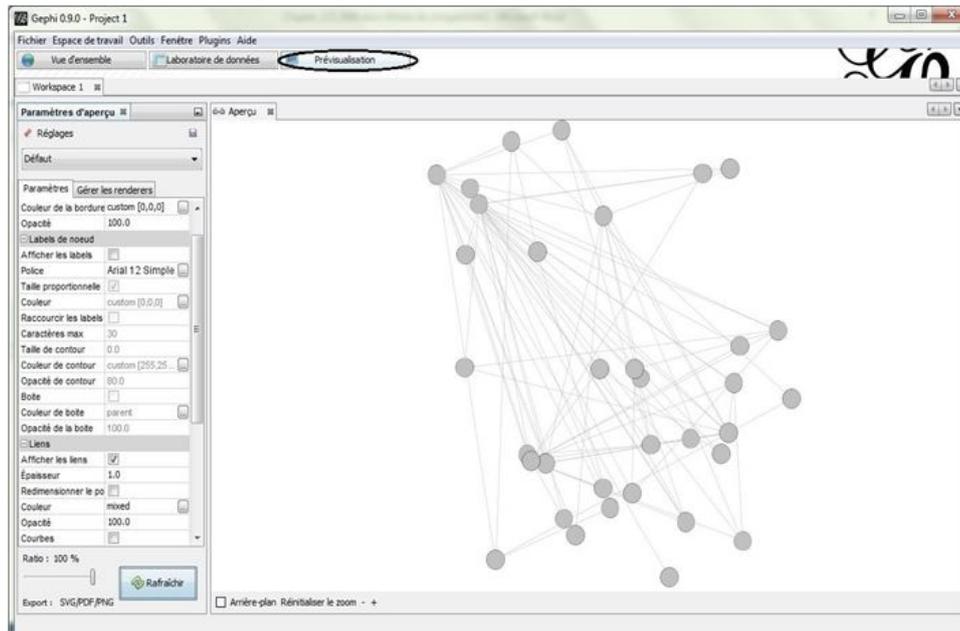


FIGURE 8 – L'onglet prévisualisation

Dans la partie Vue d'ensemble, le tiers droit de l'écran fournit :

- en haut, dans la partie Contexte : un rappel des propriétés du graphe (dirigé ou non, nombre de nœuds, nombre d'arcs) ;
- en bas, dans la partie Statistiques : plein de mesures classiques : le degré moyen, la densité, l'excentricité,... Cliquez sur Exécuter pour lancer ces mesures, puis sur le petit icône Voir le rapport si vous avez besoin d'y revenir.

# Bibliographie

- [1] L. Maesano C. Bernard et X. Le Galles. Service web avec j2ee et .net .conception et implémentation' groupe eyrolles. 2003.
- [2] M. P. Papazoglou. Service -oriented computing : Concepts characteristics and directions proceedings of the fourth international conference on web information systems engineering. 2003.
- [3] F. Casati A. Gustavo and Al. Web services : Concepts architectures and applications. 2004.
- [4] IBM. Présentation des services web. Dernière consultation 2016.
- [5] Daniel Austin Abbie Barbir Sharad Garg and Christopher Ferris. Web services architecture requirements, w3c working group note 11 february 2004 [http ://www.w3.org/tr/wsa-reqs/](http://www.w3.org/tr/wsa-reqs/)., Dernière consultation 2016.
- [6] Biplav Srivastava and J Koehler. Web service composition current solutions and open problems. 2003.
- [7] W. Nagy N. Mukhi F Curbera M Duftler R. Khalaf and S Weerawarana. Unraveling the web services web an introduction to soap wsdl and uddi. 2002.
- [8] T.Newling M. Endrei J. Ang A. Arsanjani S Chua P Comte P Krogdahl M. Luo. Patterns : Service-oriented architecture and web services' ibm international technical support organization. 2004.
- [9] T. Berners-Lee J. Hendler and O. Lassila. The semantic web' scientific am.,vol. 284. 2001.
- [10] P. Laublet C. Reynaud and J. Charlet. Le web sémantique. 2005.
- [11] A. Polleres J. Bruijn M. Stollberg D. Roman D. Fensel H. Lausen and J. Domingue. Enabling semantic web services : The web service modeling ontology' springer-verlag berlin heidelberg. 2007.
- [12] T R Gruber. A translation approach to portable ontology specifications knowledge acquisition. 1993.

- [13] Michael Daconta Leo Obrst et Kevin Smith. Developing the semantic web : a guide to the future of xml web services and knowledge management. 2003.
  - [14] Patrick Kellert et Farouk Toumani. Les web services sémantiques. 2004.
  - [15] R Akkiraju J Farrell and Al. Web service semantics. 2005.
  - [16] Hakim Hacid Zakaria Maamar and Michael N.Huns. Why web services need social networks. *IEEE Computer Society,90-94,PDF : 05731594*.
  - [17] C. Petrie. Practical web services. *IEE Internet Computing, vol. 13, no. 6, pages 94–96, 2009*.
  - [18] Kathleen M. Carley andWenji Mao Fei-YueWang, Daniel Zeng. Social computing : From social informatics to social intelligence. *IEEE Intelligent systems, vol. 22, no. 2, pages 79–83, 2007*.
  - [19] Singh and M.N. Huhns. Service-oriented computing : Semantics, processes, agents. *John Wiley et Sons, 2005*.
  - [20] Leandro Krug Wives Hamdi Yahyaoui Zakaria Maamar, Noura Faci and Hakim Hacid. Towards a method for engineering social web services. in engineering methods in the service-oriented context. *pages 153167. Springer, 2011*.
  - [21] Philippe Thiran Zakaria Maamar and Jamal Bentahar. Web services communities,from intracommunity coepetition to intercommunity competition. *E.Business Applications for Product Development and Competitive.1*.
  - [22] T. Bui and A. Gacher. Web services for negotiation and bargaining in electronic markets. *Proc. of 38th Hawaii International Conference on System Sciences, HICSS'05, 2005*.
  - [23] B. Di Martino. Semantic web services discovery based on structural ontology matching. *International Journal of Web and grid services,2009*.
  - [24] S. Kouadri Mostéfaoui Zakaria Maamar and H. Yahyaoui. Towards an agent-based and context oriented approach for web services composition. *IEEE transactions on knowledge and data engineering 17(5), may 2005*.
  - [25] Nathalie Henry J.D.F. Matrixexplorer : Un système pour l'analyse exploratoire de réseaux sociaux. *IEEE transactions on knowledge and data engineering 17(5), may 2005*.
  - [26] Guillaume Erétéo F.G. Michel Buffa Patrick Grohan. Analyse des réseaux sociaux et web sémantique : un état de l'art. 2009.
  - [27] Malek.M. Introduction à l'analyse des réseaux sociaux. 2009.
  - [28] Mercanti Guérin Maria. Analyse des reseaux sociaux et communautes en ligne : Quelles applications en marketing ?
  - [29] Saglietto.L. Quelques points de repères dans l'étude des réseaux par la théorie des graphes. 2006 : p.195-216.
-

- [30] Stattner-E. Contributions à l'étude des réseaux sociaux : propagation, fouille, collecte de donnée. 2013.
  - [31] Thovex-C. Réseaux de compétences : de l'analyse des réseaux sociaux à l'analyse prédictive de connaissances. 2012.
  - [32] Chikhi N-F. Calcul de centralité et identification de structures de communautés dans les graphes de documents. 2011.
  - [33] Yakoubi-Z. Détection et évaluation des communautés dans les réseaux complexes. 2014.
  - [34] Freeman-L.C. Centrality in social networks conceptual clarification. 1979.
  - [35] K.T.-L. Zhukov. G. W. Flake. Web dynamics : Adapting to change in content-size- topology and use. 2004.
  - [36] Fortunato. S. Community detection in graphs. 2010.
  - [37] Tremblay. N. Réseaux et signal : des outils de traitement du signal pour l'analyse des réseaux. 2014.
  - [38] R. DUNCAN LUCE-A.D.P. A method of matrix analysis of group structure. 1949.
  - [39] LUCF. R.D. Connectivity and generalized cliques in sociometric group structure. 1950.
  - [40] ALBA. R.D. A graph-theoretic definition of a sociometric clique.
  - [41] Seidman. S.B. A graph-theoretic generalization of the clique concept.
  - [42] Gergely Palla.I.D.-Illés Farkas-Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. nature. 2005.
  - [43] Rui Xu-D.W. Clustering. 2008.
  - [44] M.G M. E. J. Newman. Finding and evaluating community structure in networks.2003. 2003.
  - [45] Chikhi.N.F. Calcul de centralité et identification de structures de communautés dans les graphes de document. 2011.
  - [46] Stanica.R. Détection des communautés dans des réseaux dynamiques. 2008.
  - [47] Pascal Pons. Computing communities in large networks using random walks. 2006.
  - [48] Gaurav Agarwal. Modularity-maximizing graph communities via mathematical programming. 2008.
  - [49] Zachary. W.W. An information flow model for conflict and fission in small groups. 1977.
  - [50] Usha Nandini Raghavan. R.A. and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. 2007.
-

- [51] Kanawati. R. Licod : Leaders identification for community detection in complex networks. 2011.
  - [52] Ulrik Brandes. A faster algorithm for betweenness centrality. 2001.
  - [53] D. Sculley. Rank aggregation for similar items. 2007.
  - [54] Laurence Reboul Jean-Baptiste Angelelli, A.G. Détection de communautés, disjointes ou chevauchantes, dans les réseaux.
  - [55] M.B Santo Fortunato. Resolution limit in community detection. 2006.
  - [56] Philippe Thiran Zakaria Maamar and Jamal Bentahar. Web services communities from intracommunity coepetition to intercommunity competition. *E.Business Applications for Product Development and Competitive1*.
-