Avancement de l'implémentation de SECTra-v3 et évolution d'outils associés (TRADOH, SANDOH)

Projet Traouiéro, document L2.2.c

Ce document est un rapport d'avancement partiel de la sous-tâche 2.2 (il s'agit de sa phase c) et est joint au rapport d'avancement des tâches T2, T3 et T4 (L234.3). Il contient aussi les informations utiles sur la phase b, réalisée depuis T0+12, et sur les logiciels TRADOH et SANDOH, qui ont un peu évolué depuis leur dépôt.

Contenu

Avancement de l'implémentation de SECTra-v3 et évolution d'outils associés (TRADOH, SANDOH)

Achille FALAISE, Valérie BELLYNCK, Christian BOITET, Lingxiao WANG

RÉSUMÉ

Ce document est un rapport d'avancement partiel de la sous-tâche 2.2. Il est joint au rapport d'avancement L234.3 des tâches 2 à 4 à T0+18 (15/7/12). Il s'agit de réaliser les développements nécessaires pour le déploiement de SECTra-v3.

Ce document est le troisième document concernant cette sous-tâche.

La sous-tâche ST2.2 participe à la tâche dédiée à la tâche dédiée à l'opérationnalisation des logiciels ARIANE. Cette sous-tâche consiste en l'intégration de SECTra_w au système Ariane-Y en construction comme « serveur de corpus de TA », sans perdre bien sûr son utilisation de base pour l'évaluation de traductions (automatiques ou non, d'ailleurs), et pour la post-édition interactive et contributive de pages Web (et dans le futur de tous types de documents).

La réingénierie de SECTra-v3 a conduit à la modularisation en différents logiciels indépendants dédiés à des fonctions spécialisées, par exemple pour gérer des données (corpus, bases lexicales) ou réaliser des traitements ou transmettre des informations entre les services…

Ce document est centré sur la réingénierie du cœur de métier de SECTra_w lui-même, indépendamment des services qu'il peut rendre à ses exploitations, tant pour les iMAG que pour organiser des campagnes d'évaluation, ou que pour supporter des tests unitaires pour des systèmes de TA ou que pour fournir des vues en pseudo-documents post-éditables, annotables, évaluables, etc., à partir d'opérations effectués sur les corpus auxquels ils donne accès.

La stratégie choisie pour son développement est de procéder itérativement, en spirale, en commençant par des versions minimalistes de tous les programmes qu'il intègre.

Ce document présente, pour la première itération de cette boucle,

- l'architecture de SECTra-v3,
- la structure de la nouvelle base de données choisie,
- les spécifications, et l'implémentation de l'API Java de SECTra-v3,
- les fonctions prévues, issues de SECTra-v2 ou l'étendant en réalisant des fonctions préalablement spécifiées, ou de nouvelles fonctions.

1. INTRODUCTION

1.1 OBJET DE LA SOUS-TACHE ST2.2 : SECTRA++

La sous-tâche ST2.2 participe à la tâche 2 (ARIANE++), est dédiée à l'intégration de SECTra_w comme serveur de corpus de TA. Elle était présentée comme suit dans la proposition de projet.

T2 : ARIANE-Y++ — Nouveaux outils pour la TAO hétérogène, exécutables sur smartphones	
ST2.2 : t3-t12 — SECTra-Y	
Objectifs	Intégrer SECTra_w au système Ariane-Y comme serveur de corpus de TA
Critères d'évaluation	Utilisabilité comme composant d'Ariane-Y, avec le moniteur de base dans son état courant.

Responsable	GETALP
Partenaires	GETALP, Floralis (mise au courant pour valorisation)
Activité	La version actuelle de SECTra_w a été construite pour être utilisable aussi bien par des programmes que par des utilisateurs humains. Pour l'intégrer à Ariane-Y, il faut: ▪ définir et implémenter certaines fonctions sur les corpus, existant en Ariane-G5, mais pas encore intégrées à SECTra_w ; ▪ définir une syntaxe appropriée pour échanger avec Ariane-Y des segments, des documents, des corpus et des mémoires de traductions (formats d'import-export) ▪ définir une structure interne dans Ariane-Y pour des données "corporales".
Livrables	▪ Sources des programmes et documentation (sur la forge du LIG). ▪ Site d'essai (et de comparaison avec Ariane-G5).
Gestion du risque	Pas de risque technique.

1.2 ORGANISATION DU TRAVAIL

Ce document présente ce qui a été effectué ainsi que l'organisation prévue pour la fin de la réalisation de cette sous-tâche qui a été décomposée en 3 phases :

Phase a :	Consolidation de SECTra_w et rétro-ingénierie
Phase b :	Séparation fonctionnelle de SECTra_w en 2 outils : - dédié à la gestion de corpus - exploitation fonctionnelle par iMAG (objet de la sous-tâche 3.1)
Phase c :	Spécification de la future version faisant office de serveur de corpus pour la TA utilisable par Ariane-Y

1.2.1 ÉTATS DES PHASES A ET B

La phase A est terminée et a été l'objet d'un livrable précédent [4].

La phase B a été l'objet d'un livrable précédent [3].

Le tableau ci-dessous récapitule l'état d'avancement de SECTra et des modules associés à la fin de la phase b (T0+12, rapport T7o-L2.2.b).

Module	Version	Cahier des charges	Spécif. externes	Spécif. internes	Implémentation		
					Version α Développement, tests unitaires	Version β Intégration, validation, débogage	Version RC[1] Tests de montée en charge
SECTra	2.0 (SECTra-v2)	100%	100%	100%	100%	80%	80% ==> échec !
	3.0 (SECTra-v3)	100%	100%	50%	20%	0%	0%
SECTra-Edit	1.0	100%	100%	0%	0%	0%	0%
iMAG	3.0	100%	100%	0%	0%	0%	0%
TRADOH	2.0	100%	100%	100%	100%	50%	0%
SANDOH	1.0 (texte brut multilingue)	100%	100%	100%	100%	0%	0%
	2.0 (XML monolingue)	100%	0%	0%	0%	0%	0%
	2.1 (XML multilingue)	100%	0%	0%	0%	0%	0%
SegDoc	1.0 (segmentation en paragraphes)	100%	100%	100%	100%	0%	0%
	1.1 (segmentation en phrases)	100%	100%	0%	0%	0%	0%
Proxy	1.0 (pages publiques, contenu statique)	100%	100%	100%	50%	0%	0%

1.2.2 Buts de la phase C en cours

Cette phase consiste à progresser sur l'implémentation, le débogage et la validation des modules qui avaient été spécifiés dans la phase b, et leur intégration au sein de SECTra-v3 et iMAG-v3.

1.2.3 Buts de la phase C à venir dans le dernier livrable (T+24)

Cette phase consiste à spécifier la future version faisant office de serveur de corpus pour la TA, utilisable par Ariane-Y.

1.2.4 L'équipe

L'équipe qui se consacre à cette sous-tâche est formée d'Achille FALAISE (postdoc, ATER à la rentrée 2012), avec l'aide de Valérie BELLYNCK, de Lingxiao WANG et de Christian BOITET.

1.2.5 Méthode de travail

Pour le développement collaboratif, les forges mutualisées du LIG sont utilisées.

https://forge.imag.fr/projects/sectra/ ← SECTra, SECTra-*
http://ligforge.imag.fr/projects/imag/ ← iMAG
https://forge.imag.fr/projects/tradoh/ ← TRADOH
https://forge.imag.fr/projects/sandoh/ ← SANDOH

[1] RC : *Release Candidate*.

Quelques liens sont à retenir :

- http://www-clips.imag.fr/getalp/Services/ est le lien de base sur les services.
- http://www-clips.imag.fr/getalp/Documents/Sectra3/ contient les documents de génie logiciel. La liste des documents est :

  SECTra3-CCH/ Cahier des charges
  SECTra3-DSI/ Document de spécification externe

1.3 Objet du document

Dans ce document, il s'agit de décrire l'avancement des travaux de la phase c. Dans notre processus de développement en spirale, on précisera le cahier des charges qu'on s'est fixé pour cette phase, avec l'état attendu à la date du livrable, et on présentera l'état courant des spécifications et de l'implémentation des fonctionnalités attendues.

2. Cahier des charges

Le module SECTra-v3 a pour objectif de gérer des corpus linguistiques en général. Dans un premier temps, il se concentrera sur un type de corpus en particulier : les mémoires de traduction. L'ouverture à d'autres types de corpus doit néanmoins être envisageable.

L'ancienne version SECTra-v2 est un prototype qui pose des problèmes importants dans la perspective d'une mise en service publique. Cela est largement dû à l'histoire de ce logiciel, qui a été développé au fil d'un travail de recherche, par empilement successif de fonctionnalités. Il n'est donc pas étonnant que le code obtenu soit très complexe, entremêlant de nombreuses fonctionnalités. La base "wiki" sur laquelle se fonde SECTra-v2, qui semblait prometteuse au début du projet, s'est révélée totalement inadaptée au développement d'une application lourde. Cette base rend par exemple nécessaire d'utiliser un *plugin* assez instable pour accéder aux bases de données, et d'utiliser (en partie) un langage de programmation obsolète et mal documenté. La structure des bases de données, élaborée empiriquement, utilise une pléthore de tables, ce qui complique la maintenance et a un effet rédhibitoire sur la montée en charge. Ce travail fondateur a permis de dégager des notions essentielles, et de valider empiriquement des spécifications externes et des interfaces ; mais il doit maintenant être repensé de manière globale, en suivant des principes de génie logiciel, dans la perspective d'une mise en production et d'une montée en charge.

La version 3 de SECTra est recentrée sur les fonctions de base de SECTra-v2 ; les fonctions « secondaires » donnant lieu à la création de nouveaux modules, *à côté* du nouveau SECTra, qui sont décrits plus bas. En particulier, la nouvelle version n'effectue pas de traductions (c'est le rôle de TRADOH), et ne propose pas d'interface graphique (c'est le rôle de SECTra-Edit).

2.1 Scénarios

Nous suivons le scénario décrit dans le rapport précédent (T0+12, T7o-L2.2.b).

Un premier exemple concret a été mis en place et est exploité dans le cadre d'un service iMAG dédié pour le laboratoire franco-portugais LICIA. Le site Web de ce laboratoire (http://licia-lab.org/pt-BR/) est conçu en portugais, et l'on souhaite que celui-ci soit disponible en français et en anglais. Les traductions sont vérifiées et validées par les rédacteurs du site.

Un deuxième exemple est en cours de mise en place avec le site Web de la société Acxys (http://acxys.com), qui est intéressé par le même type de fonctionnalité. S'agissant d'un site commercial, les il y a des contraintes spécifiques qui ne peuvent pas être prises en compte par SECTra-v2, notamment la transparence de l'outil de traduction (qui ne doit pas être visible pour les visiteurs), l'hébergement ssous un nom de domaine « national » (par exemple http://acxys.de pour le site en allemand), et en matière de référencement (traductions des mots-clés dans les URL).

2.2 Processus de développement

2.2.1 Approche

Nous suivons l'approche en spirale décrite dans le rapport précédent (T0+12, T7o-L2.2.b).

2.2.2 Dépendances et priorités de développement

Certains modules sont basés sur d'autres. C'est pourquoi nous concentrons nos efforts sur les modules les plus fondamentaux : TRADOH et SegDoc. Ensuite, dans le cadre de notre scénario, viennent en priorité Proxy et iMAG.

La figure 1 récapitule ces dépendances, et indique l'état initial (à T0+12) des modules.

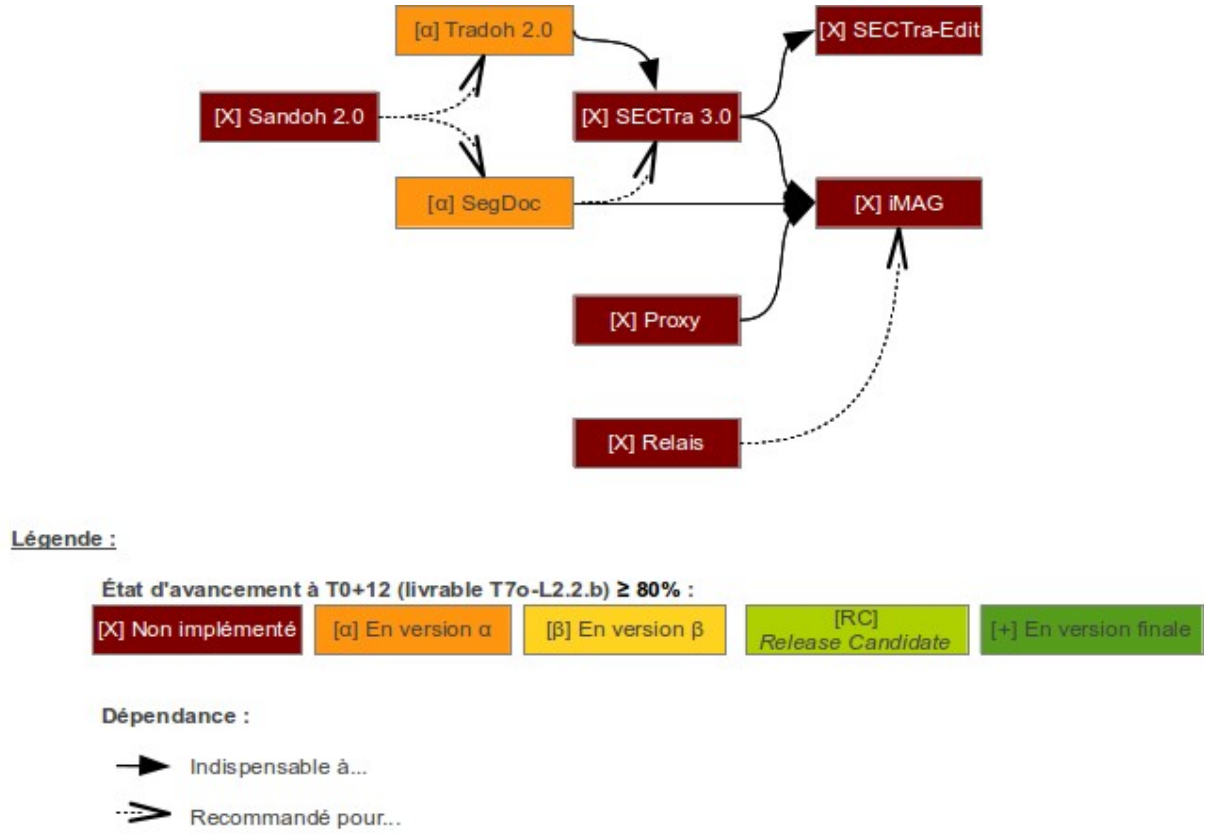

Figure 1 : dépendances de SECTra et des modules associés à T0+12

2.3 Architecture

Nous suivons l'architecture décrite dans le rapport précédent (T0+12, T7o-L2.2.b).

Quatre modules ont été ajoutés, basés sur SECTra, sur le modèle de SECTra-Edit :

- SECTra-Eval : est une interface d'évaluation de corpus de traductions.
- SECTra-Test : sert de serveur corporal de TAO (c'était l'objet initial de la sous-tâche).
- SECTra-Export : réalise une exportation de mémoires de traductions.
- SECTra-Import : réalise une importation de mémoires de traductions.

2.4 Fonctionnalités réalisées ou visées

RAPPEL L2.2.b

Les fonctions de base que fournit le nouveau SECTra sont les suivantes :

- ***createTranslationMemory(name)*** : crée un corpus de type mémoire de traduction, ayant pour nom *name*.
- ***exportTranslationMemory(name)*** : exporte une mémoire de traduction (format TMX 1.4b).

- ***importTranslationMemory(name)*** : importe une mémoire de traduction (format TMX 1.4b).
- ***removeCorpus(name)*** : supprime le corpus *name*, quel que soit son type.
- ***addOccurrence(corpusName, lang, text, context)*** : ajoute une occurrence de segment dans le corpus *corpusName*.
- ***removeOccurrence(corpusName, lang, text, context)*** : supprime une occurrence de segment.
- ***removeSegment(corpusName, lang, text)*** : supprime le segment dans le corpus *corpusName*, indépendamment de tout contexte.
- ***addTranslation(corpusName, sourceLang, sourceText, targetLang, targetText, context, author)*** : ajoute une relation de traduction entre deux occurrences de segment. L'auteur *author* peut être soit un utilisateur de SECTra, soit un système de TA. Ajoute les occurrences dans la mémoire de traduction si elles n'existent pas.
- ***getTranslation(corpusName, sourceLang, sourceText, targetLang, context)*** : retourne, s'il existe, le segment qui est la traduction de l'occurrence demandée.
- ***removeTranslation(corpusName, sourceLang, sourceText, targetLang, targetText, context)*** : supprime une relation de traduction.

Ne sont décrites ici que les fonctions exposées par l'interface (API) REST, destinées à être commodes d'utilisation. Pour de meilleures performances, d'autres fonctions doivent être prévues, permettant notamment de spécifier chaque segment par son identifiant unique dans la mémoire de traduction, plutôt que par son contenu (langue, texte et contexte).

3. Spécifications externes

3.1 Redéfinitions de notions par rapport à SECTra-v2

Un certain nombre de notions on été raffinées par rapport à la précédente version de SECTra. Ces redéfinitions ont déjà été présentées dans le rapport précédent (T0+12, T7o-L2.2.b), mais étant donné l'importance de ces notions pour la compréhension du présent rapport, elles sont rappelées ici.

3.1.1 Segment

RAPPEL L2.2.b

La notion de segment est centrale, puisque c'est l'unité sur laquelle on va travailler. Un *segment* est à la base un objet, textuel ou plus complexe (par exemple un graphe de chaînes), monolingue, dont la langue est identifiée. De nombreux types de segments dérivent de cette définition de base, et son présentés dans la thèse de HUYNH Cong Phap (voir notamment p. 169 et suivantes), notamment la notion de *multisegment* (collection de segments dont la relation entre eux n'est pas explicite). Plusieurs de ces dérivations font appel de façon impérative à la notion de *langue source,* ce qui pose problème dans la mesure où la pratique montre que cette dernière n'est pas toujours connue.

C'est pourquoi, en se basant sur les définitions de la thèse, on établit pour la nouvelle version la notion plus générique de *multisegment contextualisé* : c'est à dire une collection de segments qui, dans un contexte donné, entretiennent une relation. Cette relation pourra être plus ou moins spécifique selon les cas : lien de traduction, lorsque la *langue source* est connue (cela correspond alors au *segment multilingualisé contextualisé* de la thèse), mais aussi lien d'équivalence sémantique entre segments de langues différentes, voire entre reformulations jugées équivalentes dans une même langue. Dans tous les cas, le *contexte* de la relation est présent, et on appelle *occurrence* l'appariement d'un segment (isolé ou membre d'un multisegment) et d'un contexte donné.

3.1.2 CONTEXTE

RAPPEL L2.2.b

La notion de contexte mérite d'être encore affinée. Actuellement, il s'agit, comme dans la version précédente, du document dont est issu le segment. On ajoute à cette définition le *chemin du segment*, qui correspond dans le cas d'un document XML ou HTML au chemin d'accès du segment dans le DOM du document. On espère ainsi pouvoir distinguer plus efficacement, par exemple une occurrence d'un segment qui apparaît toujours dans le menu d'une page Web d'un site donné, de ce même segment lorsqu'il apparaît dans le corps du texte du même site.

3.1.3 LANGUE

RAPPEL L2.2.b

La notion de langue est aussi à préciser par rapport à la thèse. Il faut en effet distinguer, en plus de la langue proprement dite, ses variantes linguistiques (dialecte, spécialité, etc.) et scripturales (système d'écriture, transcription). Le codage ISO-639-2 utilisé ne permet pas, par exemple, de distinguer clairement le mandarin du cantonais. Pour le mandarin, il ne permet pas de savoir si ce dernier est écrit en chinois simplifié (majeure partie de la république populaire de Chine, mais avec d'importantes exceptions) ou en chinois traditionnel (Taïwan, diaspora chinoise, mais aussi Macao et Hong Kong), sans parler des transcriptions en alphabet latin comme le pinyin. C'est pourquoi nous suivons pour la nouvelle version les recommandations du W3C sur le codage de la langue [9].

3.2 CONTRÔLES D'ACCÈS

RAPPEL L2.2.b

Des identifiants d'utilisateur peuvent être fournis. En cas de tentative d'accès à un corpus, segments, relations, etc. avec un utilisateur ne convenant pas, une erreur HTTP 403 est retournée. Les permissions sont définies en cascade au niveau des corpus, des documents, des occurrences et enfin des relations (de traduction). Des fonctions doivent permettre de définir ces permissions, mais ne sont pas encore définies exactement. Elles devraient s'inspirer des ACL du système de fichier NTFS[2].

3.3 FORMATS D'ENTRÉE/SORTIE

Les retours se font au format XML par défaut, conformément à la philosophie REST. Du JSON, plus compact, peut être obtenu à la demande.

3.4 NOYAU : SECTRA

Pas de changement par rapport au rapport précédent (T0+12, T7o-L2.2.b).

RAPPEL L2.2.b

Le module SECTra-v3 a pour objectif de gérer des corpus linguistiques en général. Dans un premier temps, il se concentrera sur un type de corpus en particulier : les mémoires de traduction. L'ouverture à d'autres types de corpus doit néanmoins être envisageable.

Cette version de SECTra est recentrée sur les fonctions de base de SECTra_w ; les fonctions « secondaires » donnant lieu à la création de nouveaux modules, *à côté* du nouveau SECTra, qui sont décrits plus bas. En particulier, la nouvelle version n'effectue pas de traductions (c'est le rôle de Tradoh), et ne propose pas d'interface graphique (c'est le rôle de SECTra-Edit).

[2] https://fr.wikipedia.org/wiki/Access_Control_List

3.5 VUES

3.5.1 SECTRA-EDIT

Pas de changement par rapport au rapport précédent (T0+12, T7o-L2.2.b).

RAPPEL L2.2.b	Ce module est un frontal d'interaction avec l'utilisateur. Il reprend exactement l'interface graphique de l'ancien SECTra-v2, mais en l'interfaçant avec le nouveau SECTra-v3 et les nouveaux modules. Ses fonctionnalités étant assez riches, de nouveaux modules devront probablement être ajoutés. D'un point de vue externe, sa spécification ne diffère pas de celle de SECTra-v2.

3.5.2 SECTRA-EVAL

Ce module est un nouveau venu. Il doit permettre l'évaluation de corpus de traductions, avec les mêmes fonctionnalités que la version de SECTra_w développée autour des corpus EOLSS et ERIM.

3.5.3 SECTRA-TEST

Ce module sert à :

- la définition de suites de tests ;
- la gestion de corpus pour la traduction (pseudo-corpus) ;
- la mémorisation d'états après chaque phase de traitement.

3.5.4 IMAG

Pas de changement par rapport au rapport précédent (T0+12, T7o-L2.2.b).

RAPPEL L2.2.b	Il s'agit essentiellement d'un agrégateur de services, manipulant des pages Web et des segments fournis par d'autres services, et les présentant à l'utilisateur via une interface graphique. Cette dernière reprend les spécifications de l'interface existante. IMAG doit en outre mémoriser les paramètres des sites élus, de sorte à pouvoir les retrouver dans les mémoires de traduction de SECTra.

3.5.5 SECTRA-EXPORT ET SECTRA-IMPORT

Ces deux modules font leur apparition dans cette version du rapport.

3.5.5.1 Formats d'échange

La spécification des formats d'échange (importation/exportation dans SECTra) a progressé depuis le dernier rapport. Nous avons remarqué l'intérêt que présentait le standard OAXAL[3] [9], validé par l'OASIS[4]. Ce standard comprend nombre de formats utiles :

- **TMX (Translation Memory eXchange)** : mémoire de traductions ;
- **XML:TM (XML-based text memory)** : historiques de modifications portant sur des segments ;
- **XLIFF (XML Localisation Interchange File Format)** : messages de localisation ;
- **GMX (Global Information Management Metrics Exchange)** : mesure de la complexité de segments et de la qualité de traductions.
- **SRX (Segmentation Rules eXchange)** : règles de segmentation.

[3] Open Architecture for XML Authoring and Localization

[4] Organization for the Advancement of Structured Information Standards

Ces formats sont plus pauvres que les représentations dont nous avons besoin dans SECTra, mais ils ont le mérite d'exister, et d'être des formats d'échange reconnus ; nous devons donc les supporter. Cela n'interdit pas de développer par ailleurs d'autres formats correspondant mieux à nos besoins, qui idéalement pourraient être des extensions des formats OAXAL.

3.5.5.2 Exportation

Il faut dans un premier temps au moins pouvoir exporter dans les formats TMX, XML:TM, et dans un format plus riche préservant toutes nos informations. Ce dernier format n'est pas encore spécifié, mais il devrait se baser sur XML:TM.

3.5.5.3 Importation

Il faut dans un premier temps au moins pouvoir importer les formats TMX, XML:TM, et dans le format plus riche préservant toutes nos informations. Pour l'import dans les formats TMX, XML:TM, il faut prévoir des valeurs par défaut (ou des sous-spécifications) compatibles avec SECTra, ou à défaut modifier SECTra pour qu'il fonctionne efficacement avec ces valeurs par défaut ou sous-spécifications.

3.6 UTILITAIRES

3.6.1 TRADOH

3.6.1.1 TRADOH-service

Pas de changement par rapport au rapport précédent (T0+12, T7o-L2.2.b) pour TRADOH proprement dit.

RAPPEL L2.2.b

Tradoh est un service de traduction automatique multi-systèmes.

Le nouveau Tradoh présente les mêmes spécifications externes que l'ancien, notamment la possibilité de passer par une langue pivot. Cette version y ajoute :

- une API orientée service REST ;
- une gestion par *plugin* des modules de traduction automatique ;
- le choix des systèmes à utiliser et de leur ordonnancement ;
- la possibilité de demander une seule traduction (la première trouvée par Tradoh), ou bien toutes les traductions possibles ;
- la possibilité de demander une sortie simple (texte) ou détaillée (XML : erreurs, chemins de traduction, etc.) ;
- une sortie toujours en UTF-8, y compris si le système de TA appelé par Tradoh retourne en réalité autre chose (dans ce cas on convertit en UTF-8).

En cas d'erreur, si TRADOH a le choix entre essayer un autre système de TA ou passer par une langue « pivot », il commence par essayer un autre système de TA.

Tradoh peut se baser sur Sandoh pour identifier la langue et le codage lorsque ceux-ci ne sont pas spécifiés.

3.6.1.2 TRADOH-batch

Il est nécessaire d'ajouter une version utilisable en ligne de commande, pour traiter des corpus entiers (TRADOH-batch). Ce travail s'inscrit comme une étude préliminaire dans le cadre de la sous-tâche 3.2 (tableau blanc).

3.6.2 SANDOH

Pas de changement par rapport au rapport précédent (T0+12, T7o-L2.2.b).

RAPPEL L2.2.b	Sandoh permet d'identifier la langue et le codage d'un texte, et le cas échéant de découper un texte multilingue en fragments homogènes du point de vue de la langue. Ou souhaite ajouter plusieurs fonctionnalités : • identifier dans un document XML la langue de chaque nœud textuel ; • identification plus précise de la langue (conforme aux recommandations du W3C). On souhaite aussi améliorer les performances du Sandoh actuel.

3.6.3 SEGDOC

SegDoc fait l'objet d'un rapport dédié (T0+18, T7o-L3.4a).

3.6.4 PROXY

Pas de changement par rapport au rapport précédent (T0+12, T7o-L2.2.b), hormis que la gestion des cookies et de l'authentification sont reportés à une version 1.1.

RAPPEL L2.2.b	Le proxy s'applique aux documents HTML. Il remplit deux rôles : • normaliser les documents HTML pour produire des documents XML *bien formés*, mais pas nécessairement du XHTML *valide* ; • aspiration de page Web : il s'agit de remplacer toutes les URLs du document par des liens permettant de naviguer dans le document *via* un proxy quelconque. Il doit notamment pouvoir gérer les cookies et l'authentification sur l'Intranet des sites Web.

3.6.5 XMLISE

Il s'agit d'un module minimaliste, qui convertit un document HTML pas nécessairement bien formé et pas nécessairement en UTF-8, en HTML bien formé et UTF-8.

4. SPÉCIFICATIONS INTERNES

Les points les plus significatifs en termes de choix d'algorithmes, de structures de données et de formats sont indiqués ici.

4.1 NOYAU : SECTRA

4.1.1 STRUCTURE DES DONNÉES

Pour la version 3.0, on distingue 5 entités principales.

- **Méta-document** : il s'agit d'une ressource documentaire identifiée exclusivement par son *origine*, c'est à dire concrètement par une unique URI[5]. Par définition (RFC 3986), une URI est soit une URL, c'est à dire une ressource identifiée par un chemin d'accès, soit une URN, c'est à dire une ressource identifiée par un nom. Un méta-document peut s'instancier dans plusieurs versions ou variantes : des *documents*.
- **Document** : une instance de méta-document qui se caractérise par son *contenu* et par son *type* (actuellement, uniquement HTML). Si par exemple deux fichiers

[5] http://fr.wikipedia.org/wiki/Uniform_Resource_Identifier

correspondent au même méta-document et ont le même contenu, alors on considère qu'il s'agit de deux instances d'un même document.

- **Segment** : un segment représente une unité textuelle de base pour la traduction, dans une langue donnée. Un segment peut être récursif (phrases à trous par exemple), et est d'un type donné. Un segment contient du texte brut, et éventuellement des versions normalisées de ce texte, qui peuvent être utilisées pour rechercher des similarités entre segments.
- **Contexte** : il s'agit d'une notion complexe. Pour l'instant, cette notion est définie pour un segment comme l'intersection d'un document et des segments voisins (précédent et suivant) dans ce document. Un segment peut apparaître dans plusieurs contextes, et un contexte peut incorporer plusieurs segments (par exemple phrases à trous où le « trou » peut correspondre à plusieurs segments).
- **Traduction** : il s'agit d'une relation entre deux segments, dans un certain contexte, et avec un certain auteur. Cette relation peut être notée. Des segments liés entre eux par des relations de traduction forment un *segment multilingualisé contextualisé*. Cette sorte de « super-segment » peut avoir des une structure complexe, qui reflète son historique.

Limitations et notes pour de futures versions :

- Il faudra approfondir la notion d'*utilisateur*, ne serait-ce que pour gérer les droits, ce qui amènera à ajouter au moins une entité supplémentaire.
- La vue SECTra-Eval implique d'intégrer des notions de temps passé pour la postédition d'un segment.
- Les fichiers satellites et compagnons ne sont pas encore gérés. Il faudra introduire la notion de *relation entre documents*.
- Il pourra s'avérer nécessaire de suivre un méta-document qui migre ou bien est dupliqué, et donc concrètement de pouvoir lui associer plusieurs URI.
- La notion de contexte devra être enrichie. Dans le cas de documents arborescents (XML), il peut s'avérer intéressant d'ajouter le chemin complet (y compris attributs des balises XML) depuis la racine du document, ce qui permet de tenir compte de la proximité hiérarchique dans le document.
- Au besoin, en fonction du corpus, d'autres relations peuvent être crées entre segments, sur le modèle de la relation de traduction. Cela peut permettre de représenter certains corpus structurés lorsque l'utilisation d'un squelette n'est pas pertinente.

4.1.2 Bases de données

On utilise une base de données globale, et une base de données spécifique pour chaque corpus. Pour des raisons de cohérence et de performances, la structure des tables est complètement réorganisée par rapport à la version précédente, et a la structure suivante :

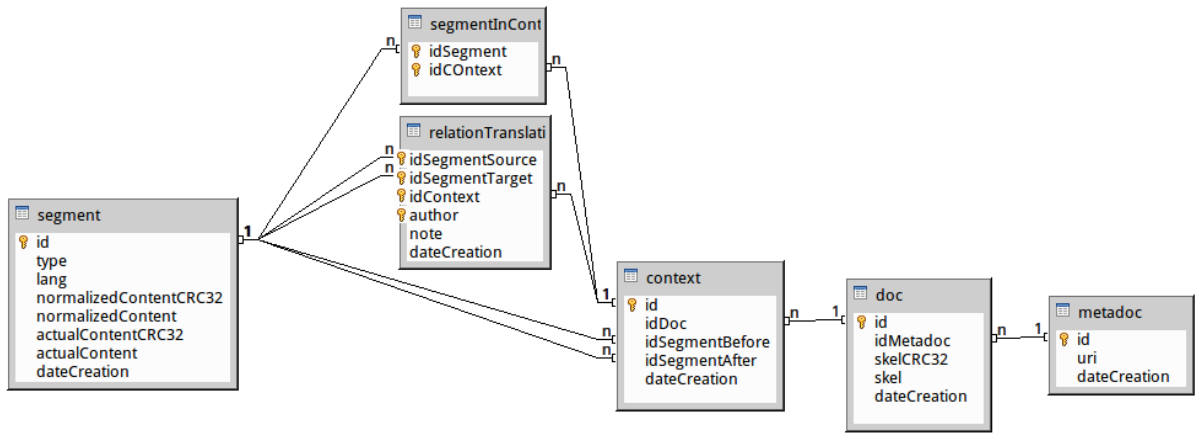

Figure 2 : structure de la base de données de SECTra-v3

Cette structure de base de données contient les principaux champs définis plus haut dans la structure de données, mais il en manque encore quelques-uns, comme par exemple le *type* de document, qui n'étaient pas indispensables pour les premiers tests ; ils seront ajoutés ultérieurement.

4.1.3 Architecture

Cette base de données n'est jamais accédée en dehors de SECTra lui-même. SECTra expose une couche d'abstraction objet, l'*API Java SECTra*, pour manipuler les mémoires de traductions. Cette couche présente en termes d'objets et d'héritage les concepts décrits dans la spécification fonctionnelle, et peut être intégrée à tout programme écrit en Java.

La *API Java SECTra* est utilisée pour créer une troisième couche, *Service SECTra*, qui permet d'utiliser SECTra en ligne de commande, et de la déployer en tant que service Web.

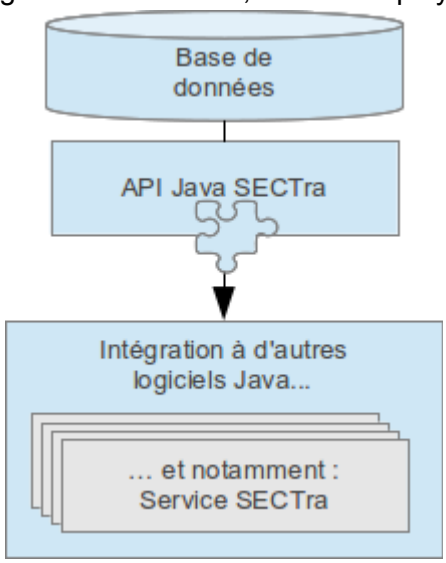

Figure 3 : couches de SECTra

4.1.4 Structure de l'API Java SECTra

L'API est structurée suivant le même modèle conceptuel que la base de données.

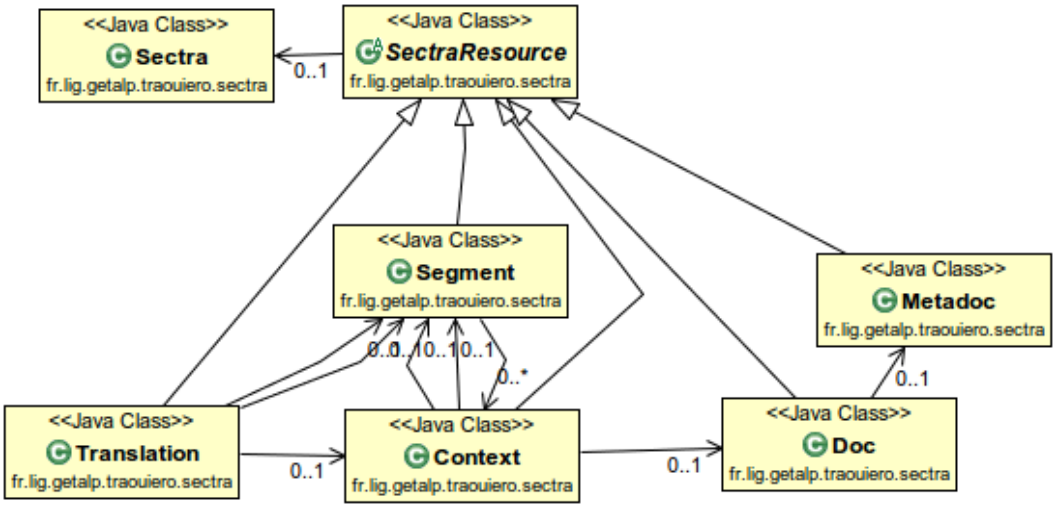

Figure 4 : diagramme UML de l'API Java

Toutes les classes correspondant à quelque chose en base de données (Segment, Translation, Context, Doc, Metadoc) dérivent de la super-classe *SectraResource*. Un corpus (classe *Sectra*) contient essentiellement un ensemble de *SectraResource*.

4.1.5 Les classes de l'API Java SECTra

Les classes exposent uniquement des méthodes « métier ». Les méthodes affectant directement la base de données sont privées, elles ne sont pas exposées.

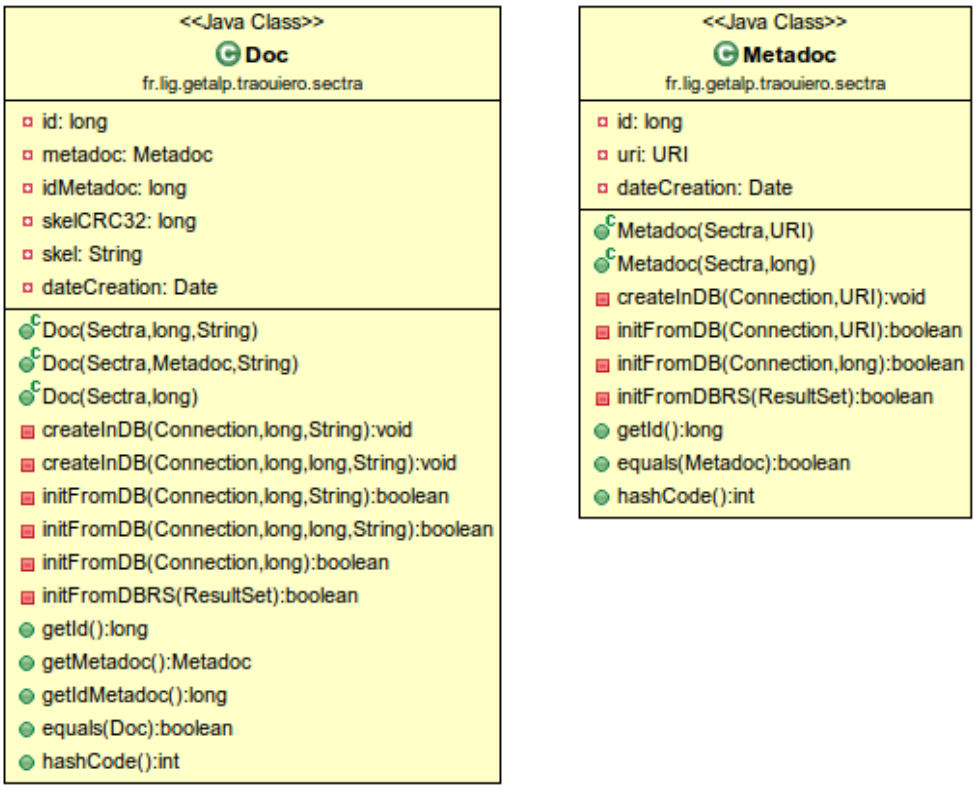

Figure 5 : diagramme UML des classes Doc et Metadoc
(attributs et méthodes privés précédés d'un carré rouge)

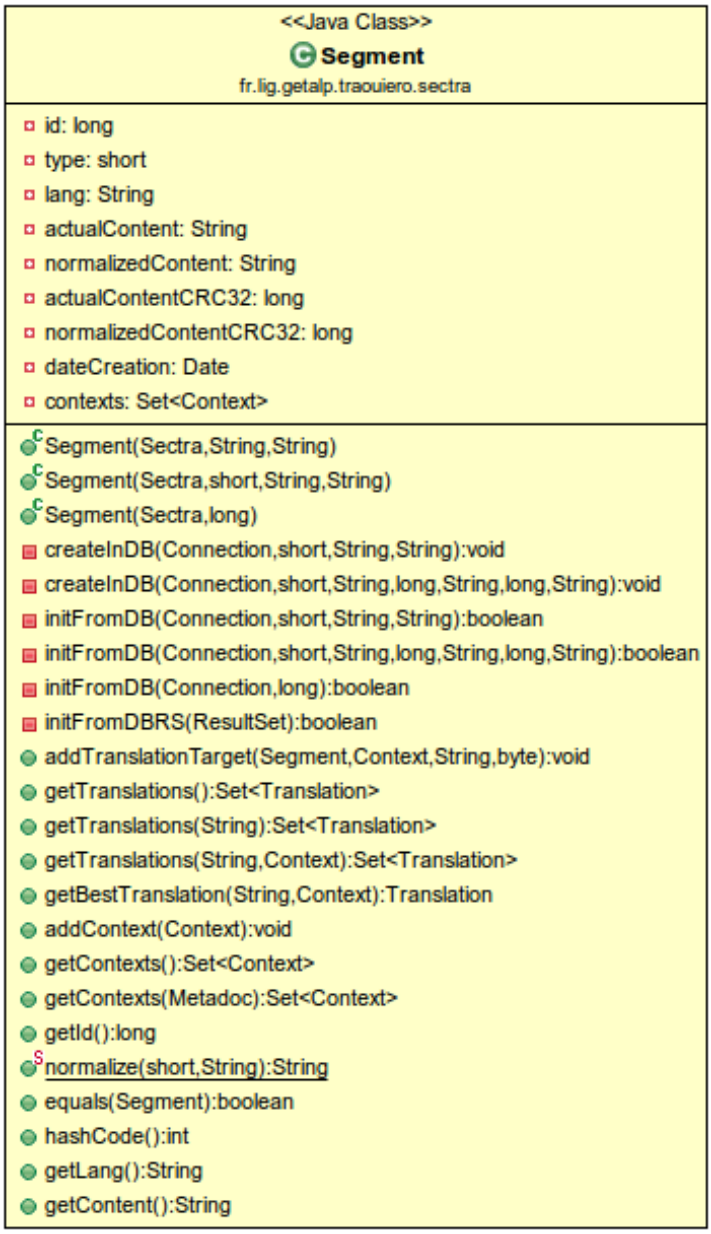

Figure 6 : diagramme UML de la classe Segment (attributs et méthodes privés précédés d'un carré rouge)

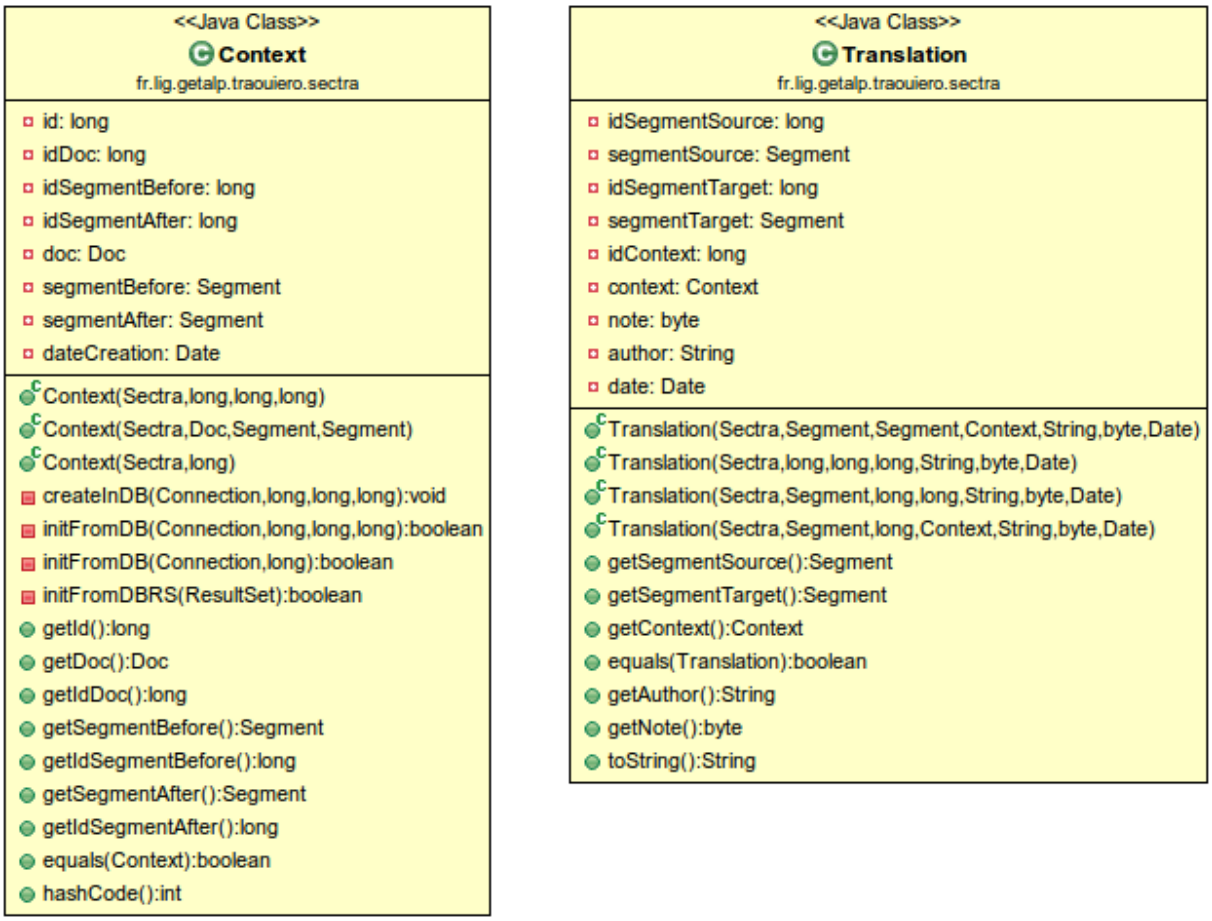

Figure 7 : diagramme UML des classes Context, et Translation (attributs et méthodes privés précédés d'un carré rouge)

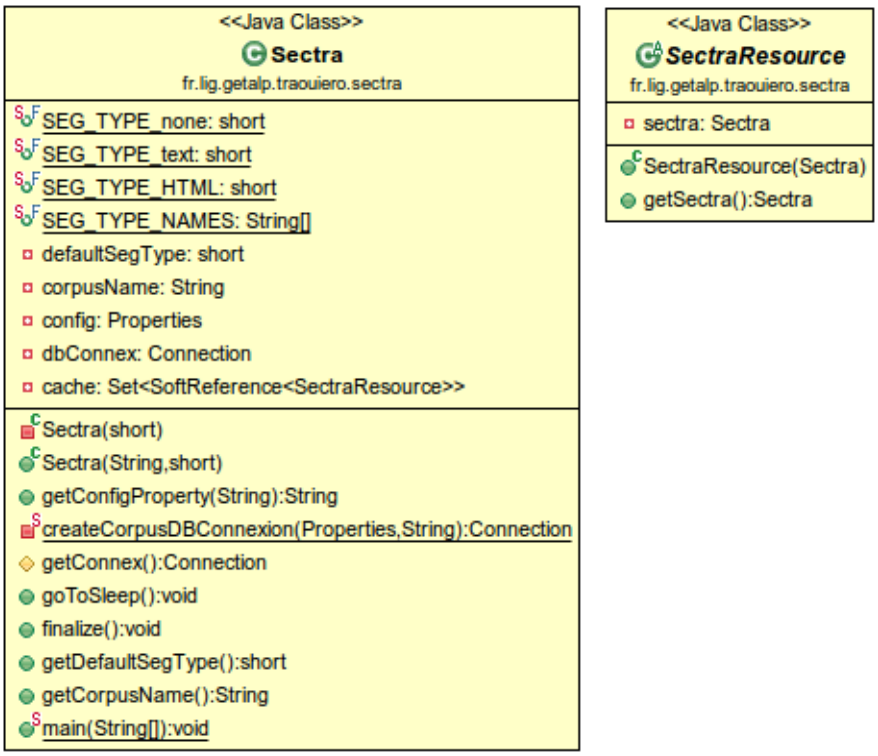

Figure 8 : diagramme UML des classes Sectra et SectraResource (attributs et méthodes privés précédés d'un carré rouge)

4.1.6 Algorithme de recherche de la meilleure traduction

Comme on peut stocker plusieurs traductions pour un même segment, il est nécessaire de prévoir un algorithme pour sélectionner la « meilleure » traduction. Cette question soulève plusieurs problèmes scientifiques concernant les critères à utiliser et la manière de les pondérer, en fonction de l'origine de la traduction, de son éventuelle notation par des utilisateurs, de son contexte, et de l'utilisation de segments plus ou moins similaires et déjà traduits.

L'algorithme présenté ici est n'est qu'une première approche intuitive. Il est basé sur un système de points ; la traduction ayant le plus de points est considérée comme la meilleure. Le barème est le suivant :

- Contexte :
  - 1 point si la traduction vient du même méta-document (même URI) ;
  - 1 point si la traduction vient du même document (même URI et même contenu) ;
  - 1 point si le segment précédent est identique ;
  - 1 point si le segment suivant est identique ;
- Note de la traduction :
  - le score est multiplié par la note de la traduction.

4.1.7 La couche Service SECTra

La couche Service SECTra offre trois méthodes d'accès :

- en ligne de commande (méthode *main*) ;
- en tant que service Web,

- *via* la méthode HTTP GET (méthode doGET) ;
- *via* la méthode HTTP POST (méthode doPOST) ;

Ces trois méthodes se chargent de récupérer les paramètres, et appellent la méthode *work*, qui passe les instructions à l'API Java SECTra.

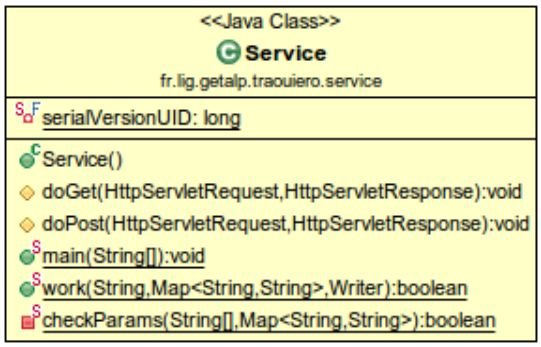

Figure 9 : diagramme UML de la classe Service
(attributs et méthodes privés précédés d'un carré rouge)

Cette couche doit permettre l'appel à toutes les fonctionnalités de SECTra, ainsi qu'aux modules associés.

4.2 Vues

4.2.1 SECTra-Edit

Pour cette version, pas de changement par rapport au rapport précédent (T0+12, T7o-L2.2.b).

RAPPEL L2.2.b

Cette interface graphique sera basée sur un principe de *templates*, proposant une vue basée sur les données des autres modules. Le premier de ces *templates* sera construit à partir de l'interface de l'ancienne version de SECTra.

Pour une version ultérieure, qui dépasserait SECTra-v2, il faudrait prévoir la possibilité de corriger la segmentation effectuée par SegDoc, car on sait qu'elle ne sera pas toujours exacte. Il faut pouvoir fusionner des segments adjacents ou au contre scinder un segment.

4.2.2 SECTra-Eval

Comme SECTra-Edit, cette interface graphique sera basée sur un principe de *templates*, proposant une vue basée sur les données des autres modules. Le premier de ces *templates* sera construit à partir de l'interface de l'ancienne version de SECTra.

4.2.3 SECTra-Test

La spécification n'est pas encore faite.

4.2.4 iMAG

iMAG consiste en l'application successive de 4 services :

- xmlise
- proxy
- segedoc/seg
- segdoc/wear

4.2.4.1 Spécifications du service

URL : http://<*racine*>/services/imag/

Paramètre	Description	Valeurs	Valeur par défaut
enc	**Facultatif.** Codage de l'URL.	**[rien]** : les URL/URI sont encodées en urlencode. **b64** : les URL/URI sont encodées en base64. **b64u** : les URL/URI sont encodées en base64url_encode, voir cette fonction.	
url	**Obligatoire.** URL où télécharger le document à segmenter.	Une URL.	
content	**Facultatif.** Contenu du document à segmenter.	Des caractères UTF-8.	
corpus	**Obligatoire.** Nom du corpus.	Un nom de corpus existant, où seront rangés, le cas échéant, les segments créés.	
docURI	**Obligatoire.** URI du document à segmenter.	L'URI identifie un document. Ce n'est pas nécessairement une URL réelle.	
il	**Facultatif.** Langue source.	Un code ISO-639-3 ou W3C. Utilisé pour ajouter des métadonnées au document.	
ol	**Facultatif.** Langue cible.	Un code ISO-639-3 ou W3C. Utilisé pour ajouter des métadonnées au document.	

4.2.4.2 Interface de test

Une interface a été développée pour faciliter le test d'iMAG. Voir figure 20 dans l'annexe « copies d'écran ».

4.2.5 SECTra-Export

L'exportation vers des formats connus est une opération triviale. L'élaboration d'un format plus riche ne devrait pas non plus poser de difficulté.

4.2.6 SECTra-Import

L'importation pose plus de problème, en particulier quand certaines informations utiles à SECTra ne sont pas prises en charge par le format d'importation. Il peut être nécessaire de modifier SECTra pour lui permettre de supporter la sous-spécification (par exemple champs ayant la valeur NULL en base de données).

4.3 Utilitaires

4.3.1 TRADOH

Quelques ajouts s'avèrent nécessaires.

4.3.1.1 TRADOH-service

Il s'appuie sur des *plugins* pour chaque système de TA.

4.3.1.1.1 Spécifications de l'interface REST de TRADOH-service

URL: http://<*racine*>/services/tradoh2/

Paramètre	Description	Valeurs	Valeur par défaut
MTmode	**Facultatif.** Mode de traduction.	**All**: utiliser tous les systèmes. **Once**: s'arrêter au 1er qui marche.	*Once*
text	**Obligatoire.** Texte à traduire.	Le texte à traduire, codé en UTF-8.	
source	**Obligatoire.** Langue source.	Un code ISO-639-3 ou W3C.	
target	**Obligatoire.** Langue cible.	Un code ISO-639-3 ou W3C.	
MT	**Facultatif.** Liste des systèmes de traduction à utiliser.	Une liste de noms de systèmes, séparés par des barres verticales. La liste des noms de systèmes est disponible dans l'interface de test.	*Etap3*|*Sway*|*Sistec*|*Systran6*|*Google* **(sujet à modification)**
piv	**Facultatif.** Liste des langues pivot.	Une liste de code ISO-639-3 ou W3C, séparés par des barres verticales. L'absence de pivot est obtenu par le caractère [espace] (#x20;).	[espace]
outFormat	**Facultatif.** Format de sortie.	*/xml* ou *text*. La valeur *text* force MTmode=*Once*.	*/xml*
verbose	**Facultatif.** Retourne des logs.	*0* ou *1*.	*0*
nocache	**Facultatif.** Désactive le cache.	*0* ou *1*.	*0*

4.3.1.1.2 Sorties

- si une traduction est possible : code HTTP 200, et :
  - format *text* : la traduction.
  - format */xml* : la trace dans un format XML :

```
- <tradoh version="2.0.13">
  - <translation>
      <source lang="eng">Hello world</source>
      <target lang="fra" mt="Google">Bonjour tout le monde</target>
    </translation>
  </tradoh>
```

Figure 10 : exemple de sortie XML de TRADOH
(traduction réussie, un seul système de TA, pas de pivot)

```
- <tradoh version="2.0.13">
  - <error>
      <source lang="fra">Bonjour tout le monde.</source>
      <target lang="zho" mt="Etap3">Direction Language pair not available: .</target>
    </error>
  - <translation>
      <source lang="fra">Bonjour tout le monde.</source>
      <target lang="zho" mt="Google">大家好。</target>
    </translation>
  </tradoh>
```

Figure 11 : exemple de sortie XML de TRADOH (deux systèmes de TA, l'un avec une traduction

échouée, l'autre avec une traduction réussie)

```
- <tradoh version="2.0.13">
  - <translation>
      <source lang="fra">Bonjour tout le monde.</source>
      <target lang="zho" mt="Google">大家好。</target>
    </translation>
  - <translation>
    - <source lang="eng">
      - <translation>
          <source lang="fra">Bonjour tout le monde.</source>
          <target lang="eng" mt="Google">Hello everyone.</target>
        </translation>
      </source>
      <target lang="zho" mt="Google">大家好。</target>
    </translation>
  </tradoh>
```

Figure 12 : exemple de sortie XML récursive de TRADOH (un seul système de TA, paramètres MTmode=All et piv= |eng, traduction réussie à la fois directement et par pivot)

- si aucune traduction n'est possible : code HTTP 501, et une sortie XML (ne comportant que des traces d'erreurs) si c'est le mode actif.

4.3.1.2 Méthode Java

En plus de ce service Web, une méthode Java permet l'appel à TRADOH, avec au moins les paramètres suivants : texte à traduire, langue source et langue cible, afin que SECTra puisse communiquer avec TRADOH. Les autres paramètres potentiels bénéficient d'une valeur par défaut dans TRADOH.

4.3.1.3 Interface de test

Une interface a été développée pour faciliter le test de TRADOH. Voir figure 19 dans l'annexe « copies d'écran ».

4.3.1.4 TRADOH-batch

Il s'agit d'un script, qui traite des dossiers contenant des fichiers textes. Il y a un dossier par langue, contenant un fichier par texte.

4.3.1.4.1 Dépendances

Le script requiert *perl*, *curl* et *iconv*.

4.3.1.4.2 Paramètres

Les paramètres du script sont les suivants :

- l'URL du service TRADOH (par exemple *http://getalp.imag.fr/tradoh/*) ;
- la liste des langues et des codages à traiter, en indiquant le nom du dossier, le nom du codage (nom iconv, voir la commande *iconv -l* pour la liste des codages supportés), et le code TRADOH : par exemple *de-Macintosh-deu*|*en-Macintosh-eng*| *fr-Macintosh-fra*, si les dossiers à traiter s'appellent *de*, *en* et *fr*, et contiennent des fichiers encodés au format Macintosh ;
- la liste des langues cibles (code TRADOH), par exemple deu|eng|fra.

4.3.1.4.3 Sortie

Le script produit des dossiers intermédiaires (pour la conversion d'encodage) et des dossiers de traductions, de la forme <*langueSource*>2<*langueCible*>_<*systèmeTA*>, où <*langueSource*> est la langue source, <*langueCible*> la langue cible, et <*systèmeTA*> le

système de TA utilisé. Les fichiers ont le même nom que les fichiers textes en source, et il s'agit de texte codé en UTF-8.

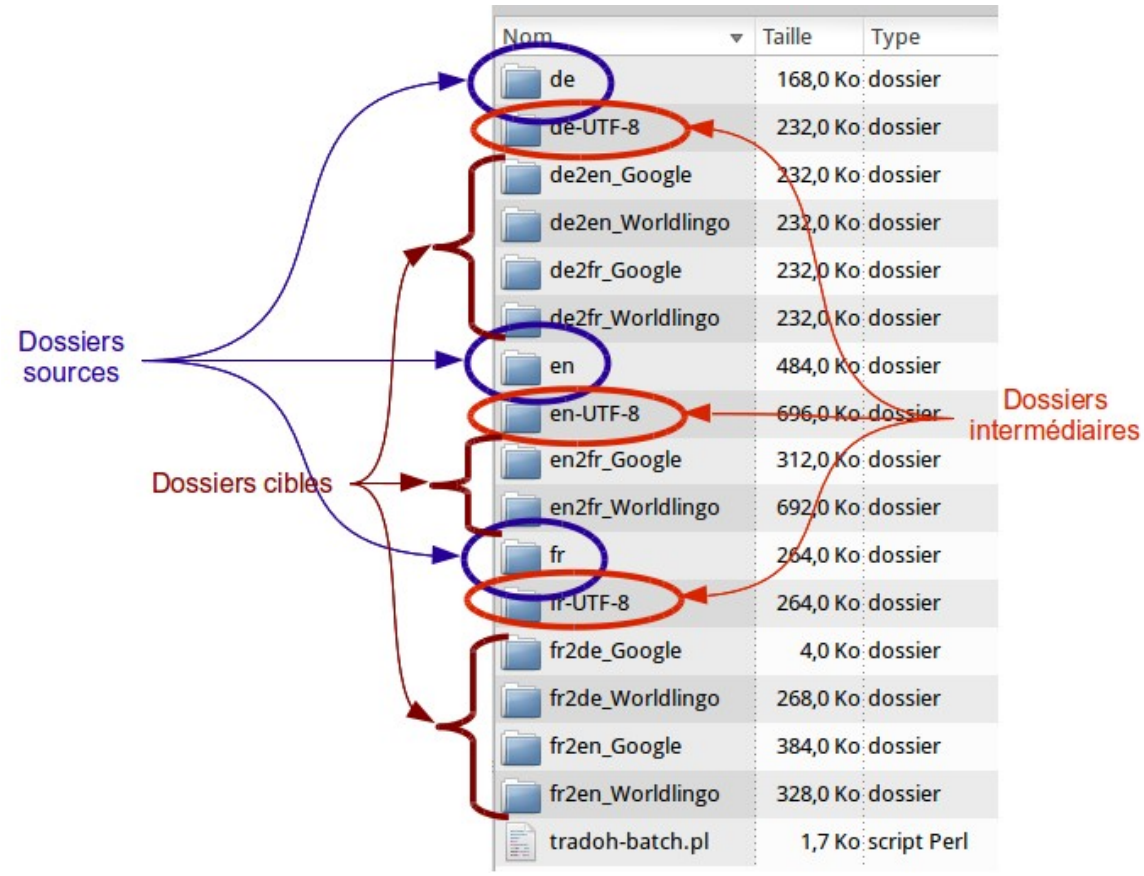

Figure 13 : organisation des fichiers pour TRADOH-batch

4.3.2 SANDOH

Les spécifications sont celles du SANDOH initial.

4.3.3 SEGDOC

Voir le rapport T7o-L3.4a.

4.3.3.1 Segmentation

URL : http://<*racine*>/services/segdoc/seg/

Paramètre	Description	Valeurs	Valeur par défaut
enc	**Facultatif.** Codage de l'URL.	**[rien]** : les URL/URI sont encodées en urlencode. **b64** : les URL/URI sont encodées en base64. **b64u** : les URL/URI sont encodées en base64url_encode, voir cette fonction.	
url	**Facultatif.** URL où télécharger le document à segmenter.	Une URL.	
content	**Facultatif.** Contenu du document à segmenter.	Des caractères UTF-8.	
corpus	**Obligatoire.** Nom du corpus.	Un nom de corpus existant, où seront rangés, le cas échéant, les segments créés.	
docURI	**Obligatoire.** URI du document à segmenter.	L'URI identifie un document. Ce n'est pas nécessairement une URL réelle.	
type	**Obligatoire.** Liste des langues pivot.	**HTML**	
defaultLang	**Obligatoire.** Langue par défaut du document.	Un code ISO-639-3 ou W3C.	

4.3.3.2 Habillage

URL : http://<*racine*>/services/segdoc/wear/

Paramètre	Description	Valeurs	Valeur par défaut
enc	**Facultatif.** Codage de l'URL.	**[rien]** : les URL/URI sont encodées en urlencode. **b64** : les URL/URI sont encodées en base64. **b64u** : les URL/URI sont encodées en base64url_encode, voir cette fonction.	
url	**Facultatif.** URL où télécharger le document à segmenter.	Une URL.	
content	**Facultatif.** Contenu du document à segmenter.	Des caractères UTF-8.	
corpus	**Obligatoire.** Nom du corpus.	Un nom de corpus existant, où seront rangés, le cas échéant, les segments créés.	
docURI	**Obligatoire.** URI du document à segmenter.	L'URI identifie un document. Ce n'est pas nécessairement une URL réelle.	
type	**Obligatoire.** Liste des langues pivot.	*HTML*	
ol	**Obligatoire.** Langue cible.	Un code ISO-639-3 ou W3C.	

4.3.4 PROXY

URL : http://<*racine*>/services/proxy/

Paramètre	Description	Valeurs	Valeur par défaut
enc	**Facultatif.** Codage de l'URL.	**[rien]** : les URL/URI sont encodées en urlencode. **b64** : les URL/URI sont encodées en base64. **b64u** : les URL/URI sont encodées en base64url_encode, voir cette fonction.	
url	**Obligatoire.** URL où télécharger le document à segmenter.	Une URL.	
docURI	**Obligatoire.** URI du document.	L'URI identifie un document. Ce n'est pas nécessairement une URL réelle.	
proxy	**Obligatoire.** URL du proxy.	Une URL.	
corpus	**Obligatoire.** Nom du corpus.	Un nom de corpus existant, où seront rangés, le cas échéant, les segments créés.	
docURI	**Obligatoire.** URI du document à segmenter.	L'URI identifie un document. Ce n'est pas nécessairement une URL réelle.	
il	**Facultatif.** Langue source.	Un code ISO-639-3 ou W3C. Utilisé pour ajouter des métadonnées au document.	
ol	**Facultatif.** Langue cible.	Un code ISO-639-3 ou W3C. Utilisé pour ajouter des métadonnées au document.	

4.3.5 XMLISE

Xmlise se base sur la librairie Tidy pour reformer le HTML, et sur SANDOH pour détecter l'encodage et produire de l'UTF-8.

URL : http://<*racine*>/services/xmlise/

Paramètre	Description	Valeurs	Valeur par défaut
enc	**Facultatif.** Codage de l'URL.	**[rien]** : les URL/URI sont encodées en urlencode. **b64** : les URL/URI sont encodées en base64. **b64u** : les URL/URI sont encodées en base64url_encode, voir cette fonction.	
url	**Obligatoire.** URL où télécharger le document à segmenter.	Une URL.	

5. IMPLÉMENTATION

L'état d'avancement de l'implémentation est indiqué ici. À l'exception de SECTra, SECTra-Edit et iMAG, tous les modules sont fonctionnels, au moins à un stade préliminaire.

Les implémentations sont prévues et testées pour une exécution sur plateforme Unix (Linux ou MacOS). S'agissant de Java et de langages de script, l'exécution devrait être possible sur plateforme Windows, mais cette possibilité n'est pas testée pour l'instant. Ces aspects de machine cible peuvent se révéler importants pour le cas où un client voudrait effectuer une installation sur site.

5.1 NOYAU : SECTRA

SECTra est implémenté en Java. Il s'appuie sur une base de données MySQL. Les spécifications et l'implémentation de cette base sont stables sur le plan de la structure, mais des champs et des tables seront vraisemblablement ajoutées par la suite.

5.1.1 BASE DE DONNÉES

La base de données est implémentée en MySQL conformément aux spécifications.

MySQL n'indexe pas les chaînes de caractères de manière optimale (indexation basée sur les *n* premiers caractères), c'est pourquoi, pour toute chaîne qui peut faire l'objet d'une recherche, on ajoute sa clé de hachage CRC32.

5.1.2 API JAVA SECTRA

L'API est implémentée conformément aux spécifications.

Les corpus pouvant être de taille importante, il n'est pas envisageable de les conserver entièrement en mémoire de travail. C'est pourquoi l'API charge dynamiquement les éléments (segments, contextes, documents, etc.) dont elle a besoin, en créant des instances de classes à la volée, qui sont libérées après usage.

Après libération, ces instances sont toutefois conservées temporairement dans un cache, ce qui permet d'accélérer les séquences de requêtes concernant les mêmes objets, et de réduire la charge de la base de données, qui est souvent le goulot d'étranglement sur ce genre d'applications. Le cache est constitué d'un *Set* de *WeakReference* pointant vers les instances d'éléments de corpus. L'intérêt d'utiliser une *WeakReference* comme pointeur vient du fait qu'elle peut être détruite à tout moment par le ramasse-miettes de Java, ce qui permet d'utiliser au maximum la mémoire disponible, mais sans risquer de la saturer. Ainsi SECTra tire toujours profit de toute la mémoire (de plus en plus conséquente sur les machines modernes) que l'on peut lui allouer.

5.1.3 COUCHE SERVICE SECTRA

Actuellement, la couche Service n'implémente que 2 fonctionnalités, liées au module SegDoc (voir le rapport T7o-L3.4.a concernant ce module) :

- *seg* : segmentation de documents (actuellement uniquement de documents HTML) ;
- *wear* : habillage de squelettes de documents.

Conformément aux spécifications, cette couche peut être utilisée :

- en ligne de commande ;
- en tant que service Web ; elle peut être déployée directement :
  - dans un serveur Tomcat/Java via une archive war fournie ;
  - dans un serveur Apache/PHP à l'aide d'une archive exécutable jar et d'une capsule PHP fournie.

5.2 Vues

5.2.1 SECTra-Edit

L'implémentation n'a pas commencé.

5.2.2 SECTra-Eval

L'implémentation n'a pas commencé.

5.2.3 SECTra-Test

L'implémentation n'a pas commencé.

5.2.4 iMAG

IMAG est implémenté en PHP, mais sans interface de postédition. Voir la figure 20 dans l'annexe « copie d'écrans ». Il s'agit de l'objectif de la tâche 3.1 (logiciel iMAG et relais iMAG).

5.2.5 SECTra-Export

L'implémentation de l'exportation vers le format TMX a débuté.

5.2.6 SECTra-Import

L'implémentation n'a pas commencé.

5.3 Outillage

5.3.1 TRADOH 2.0

TRADOH 2.0 est pleinement fonctionnel et déjà utilisé pour d'autres projets.

Si TRADOH lui-même semble exempt de bug à ce jour, il peut rester des bugs liés aux 7 plugins, qui sont corrigés au fur et à mesure. La dernière versions est la 2.0.14. Il arrive aussi que les systèmes de TA appelés ne fonctionnent pas au moment de l'appel, en particulier parce que certains d'entre eux sont des systèmes expérimentaux. Dans ce cas, TRADOH peut mettre longtemps à répondre (si il attend une réponse d'un système qui ne répond jamais). Pour éviter cela, une fonction de *blacklistage* temporaire de systèmes de TA défaillants a été mise en place, et est en cours d'évaluation.

5.3.1.1 Plugins

Actuellement, TRADOH supporte les couples service/langue suivants :

1. Etap3 : anglais, russe, UNL (code *unl), arbre syntaxique (code *sts) ;
2. Google : italien, français, anglais, allemand, suédois, vietnamien, espagnol, russe, ukrainien et chinois simplifié ;
3. Sistec : anglais, malais ;
4. Sway : UNL (code *unl), français ;
5. Systran6 : italien, français, anglais, allemand, suédois, russe, chinois simplifié ;
6. Systran7 : anglais, chinois ;
7. Worldlingo : portugais, anglais, français, arabe, japonais, bulgare, chinois simplifié, chinois traditionnel, tchèque, danois, néerlandais, farsi, allemand, grec, haoussa, hébreux, hindi, hongrois, italien, coréen, norvégien, polonais, russe, espagnol, suédois, thaï, turc, ourdou.

5.3.1.2 Support du texte balisé

Le texte comportant des balises HTML est souvent incorrectement traité par les systèmes de traduction. Pour le plugin Worldlingo, qui gère correctement les balises, mais pas leurs attributs, TRADOH remplace les balises par des identifiants, et reconstitue les balises et leurs attributs en sortie.

5.3.1.3 TRADOH-batch

TRADOH-batch est implémenté en Perl et a déjà été testé utilisé avec succès. Le passage de paramètres reste à améliorer.

5.3.1.4 Évaluation

TRADOH et TRADOH-batch ont été testés sur des dizaines de milliers de segments. Le traitement est parfois lent, du fait de la lenteur des systèmes de traduction que nous utilisons. Cela peut s'améliorer de deux manières :

- en utilisant des systèmes de TA locaux ;
- en envoyant aux systèmes de TA des listes de segments plutôt que des segments individuels.

Ce sera l'objet de la version 2.1 de TRADOH.

5.3.2 SANDOH

L'implémentation est en cours.

5.3.3 SegDoc

Voir le rapport T7o-L3.4a.

5.3.4 Proxy

Une version complète est implémentée, et est en cours de validation.

5.3.5 Xmlise

L'appel à SANDOH n'est pas encore implémenté. Le reste fonctionne, et est en cours de validation.

5.4 Déploiement

Les modules sont actuellement déployés sur deux plateformes de test : Ubuntu 12.04 et Debian 6.0. Ils devraient fonctionner sur tout système Unix. Ils nécessitent Apache, PHP (+module Tidy), MySQL et une JVM Java. Les composants écrits en Java (SECTra, SegDoc, SANDOH) étant encapsulés dans des archives jar contrôlés par des scripts PHP, il n'est pas nécessaire d'utiliser un serveur d'applications Java tel que Tomcat, même si cela reste possible (à privilégier pour les performances).

Pour installer les modules dans un tel environnement, il faut :

- copier les modules PHP dans le dossier *www* d'Apache ;
- vérifier les permissions sur les dossiers ./services/<*module*>/temp des modules (le cas échéant) ; Apache doit pouvoir écrire dedans ;
- créer une base de données pour le corpus ;
- importer un template de corpus dans cette base ;
- vérifier la configuration des modules (fichier ./services/<*module*>/config.php si il existe, sinon premières lignes des fichiers ./services/<*module*>/index.php).

6. Récapitulatif de l'état d'avancement

Module	Version	Cahier des charges	Spécif. externes	Spécif. internes	Implémentation		
					Version α Développement, tests unitaires	Version β Intégration, validation, débogage	Version RC[6] Tests de montée en charge
SECTra	2.0 (SECTra_w)	100%	100%	100%	100%	80%	80% ==> échec !
	3.0 - noyau	100%	100%	100%	100%	80%	0%
	3.0 - API Java	100%	100%	100%	100%	80%	0%
	3.0 - service	100%	100%	100%	50%	0%	0%
SECTra-Edit	1.0	100%	100%	0%	0%	0%	0%
SECTra-Eval	1.0	100%	100%	0%	0%	0%	0%
SECTra-Test	1.0	100%	0%	0%	0%	0%	0%
iMAG	3.0	100%	100%	100%	80%	20%	0%
SECTra-Export	1.0	100%	50%	50%	20%	0%	0%
SECTra-Import	1.0	100%	20%	0%	0%	0%	0%
TRADOH	2.0 - service	100%	100%	100%	100%	90%	80% ==> OK !
	2.0 - batch	100%	100%	100%	100%	90%	100% ==> OK !
SANDOH	1.0 (texte brut multilingue)	100%	100%	100%	100%	0%	0%
	2.0 (XML monolingue)	100%	100%	100%	50%	0%	0%
	2.1 (XML multilingue)	100%	100%	100%	20%	0%	0%
SegDoc	1.0 (segmentation en paragraphes)	100%	100%	100%	100%	80%	20%
	1.1 (segmentation en phrases)	100%	100%	100%	0%	0%	0%
Proxy	1.0 (pages publiques, contenu statique)	100%	100%	100%	100%	50%	0%
	1.1 (cookies, authentification)	100%	100%	0%	0%	0%	0%
Xmlise	1.0	100%	100%	100%	80%	80%	0%

[6] RC : *Release Candidate*.

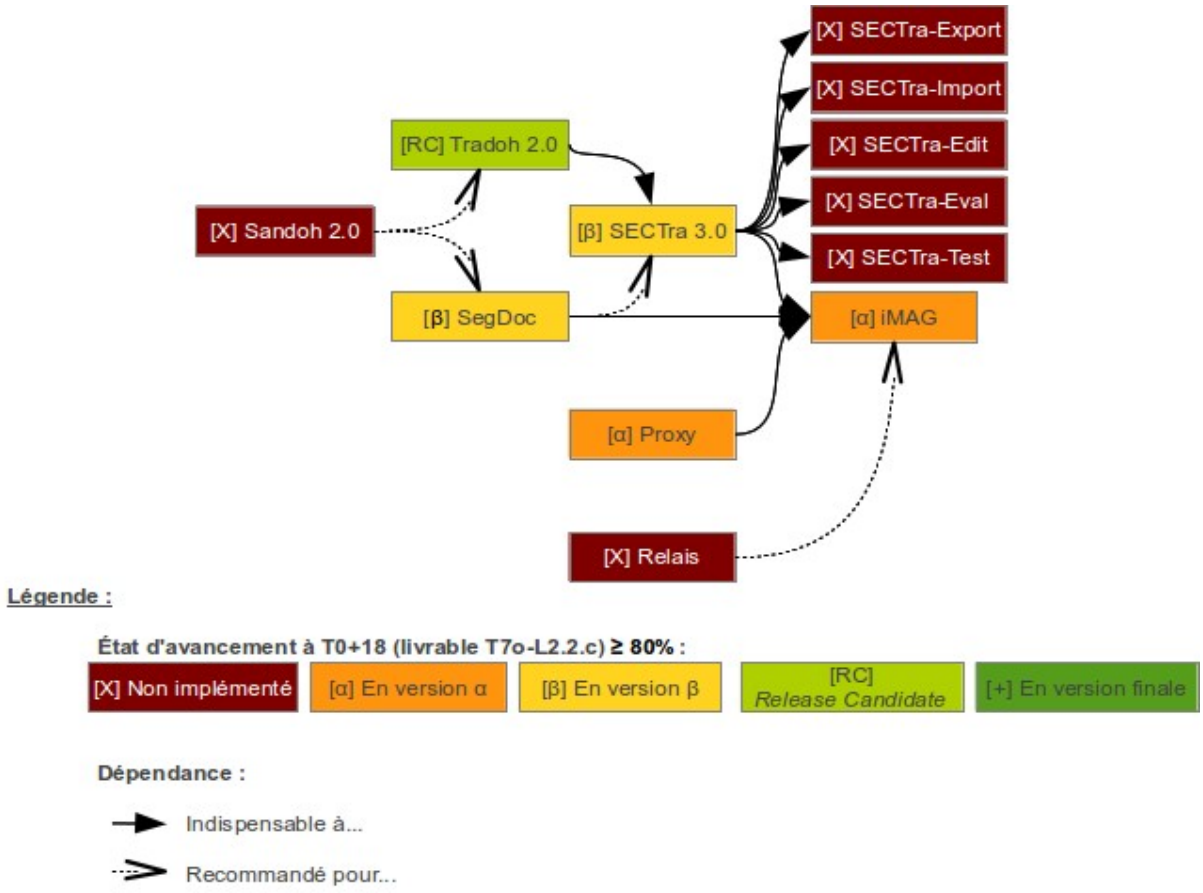

Figure 14 : dépendances de SECTra et des modules associés à T0+18

7. CONSOLIDATION SUR LA VERSION PRÉCÉDENTE XWIKI (SUITE AU DOCUMENT 2.2.A) PAR ZHANG YING, WANG LINGXIAO ET ACHILLE FALAISE

La version de SECTra3 n'étant pas opérationnelle, il a été nécessaire de poursuivre quelques travaux d'amélioration de la version 2, de SECTra_w, basée sur XWiki, dont les documents sont consultables en ligne à l'adresse http://aximag.ligforge.imag.fr/GL/. La liste des documents est :

SECTra_w-UG/	Manuel d'utilisateur de SECTra_w
iMAG-CCH/	Cahier des charges
iMAG-CG/	Conception globale
iMAG-DSE/	Document de spécification externe
iMAG-GR/	Gestion des risques
iMAG-PQ/	Plan qualité

7.1 FORGE

7.1.1 LES SPÉCIFICITÉS DU LOGICIEL SECTRA_W

Le logiciel SECTra_w a été développé par HUYNH Cong Phap comme une spécialisation du moteur de sites Web Xwiki. Cet outil a l'intérêt de permettre de réaliser rapidement un site Web intégrant des fonctionnalités avancées exploitant des bases de données. Il intègre, comme tous les CMS, une gestion des utilisateurs. Mais de plus, il permet d'écrire des

contenus (articles dynamiques) avec des langages de script interprétés par le moteur pour plus de sûreté : Velocity et Groovy.

La version de Xwiki utilisée pour le logiciel est 1.3.1, ce qui est important parce que :

- les syntaxes des langages Groovy et Velocity ont évolué et ne sont pas compatibles avec les nouvelles versions (pas de rétro-compatibilité),
- certains des codes développés pour spécialiser SECTra_w sont introduits dans les informations de la base de données de la plate-forme Xwiki classiquement comme des articles, ce qui rend impossible l'utilisation directe et uniforme d'un logiciel de journalisation de fichiers comme SVN.

Il est à noter pour finir que certains autres codes de SECTra_w sont intégrés aux codes de la plate-forme Xwiki elle-même (fichiers javascript, templates…)

Pour déposer les codes spécifiques de SECTra_w sur la forge, il a fallu trouver une stratégie qui regroupe tous les codes dans un seul dossier.

7.1.2 La forge

La forge est à l'adresse https://ligforge.imag.fr/scm/?group_id=148. Pour y accéder, il faut un compte sur cette forge et que les administrateurs du projet vous y aient autorisé.

7.1.3 Constitution du dépôt

L'illustration ci-dessous montre la liste des dossiers du dossier correspondant au dépôt de la forge du LIG appelée ligforge :

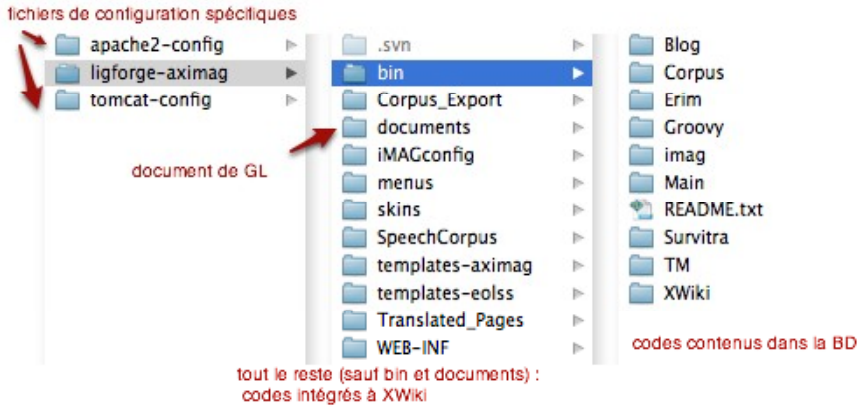

Figure 15 : dépôt du logiciel SECTra_w/iMAG sur la forge du LIG

7.2 Exportation

Un script PHP, exporttmx-all-v3.php, est installé sur le serveur *aximag.fr* et permet d'exporter des mémoires de traduction au format TMX, en vue d'une réimportation dans SECTra-v3.

Le chemin exact de ce fichier n'est pas donné ici, pour des raisons de sécurité, mais n'est pas difficile à trouver si on a un accès SSH au serveur (partir de */var/www*).

La configuration du script se fait dans le fichier :

```
$db = 'eolss';  // Indiquer le nom du corpus (base de données)
$version = 3;
// Version du script
$srcLang = 'en';
// Langue des segments source (ISO 2 caractères)
$nbSegMax = 1000 ;
// Nombre maximum de segments
```

Le script est utilisable uniquement en ligne de commande depuis le serveur (commande *php exporttmx-all-v3.php*). La sortie est dans le dossier *out* (un dossier par paire de langue).

Par exemple, avec la configuration ci-dessus, le fichier *./out/SECTra_w-eolss-en-fr.tmx* contient :

```
<tmx version="1.4b">
 <header creationtool="SECTra_w-exporttmx-all" creationtoolversion="3" o-tmf="SECTra_w-v1-Database" segtype="block" srclang="en" adminlang="en" datatype="html"/>
 <body>
  <tu id="d10_e2_25_06_txt---$$_sent_1">
   <tuv xml:lang="en">
    <seg><![CDATA[BETWEEN THE GREAT RIVERS]]></seg>
   </tuv>
   <tuv xml:lang="fr">
    <seg><![CDATA[ENTRE LES GRANDS FLEUVES]]></seg>
   </tuv>
  </tu>
  <tu id="d10_e2_25_06_txt---$$_sent_2">
   <tuv xml:lang="en">
    <seg><![CDATA[WATER IN THE MIDDLE EAST AND NORTH AFRICA]]></seg>
   </tuv>
   <tuv xml:lang="fr">
    <seg><![CDATA[L'EAU AU MOYEN-ORIENT ET EN AFRIQUE DU NORD]]></seg>
   </tuv>
  </tu>
  <tu id="d10_e2_25_06_txt---$$_sent_3">
   <tuv xml:lang="en">
    <seg><![CDATA[BETWEEN THE GREAT RIVERS]]></seg>
   </tuv>
   <tuv xml:lang="fr">
    <seg><![CDATA[ENTRE LES GRANDS FLEUVES]]></seg>
   </tuv>
  </tu>
 </body>
</tmx>
```

Figure 16 : mémoire de traduction exportée au format TMX

7.3 iMAG TRANSPARENTES AVEC PROXIMAG

7.3.1 CAHIER DES CHARGES

C'est une demande récurrente de utilisateurs de pouvoir avoir des iMAG « transparentes », c'est à dire sans message d'avertissement et formulaire imposé, sans bandeau et accessibles ailleurs que sur le domaine *aximag.fr*.

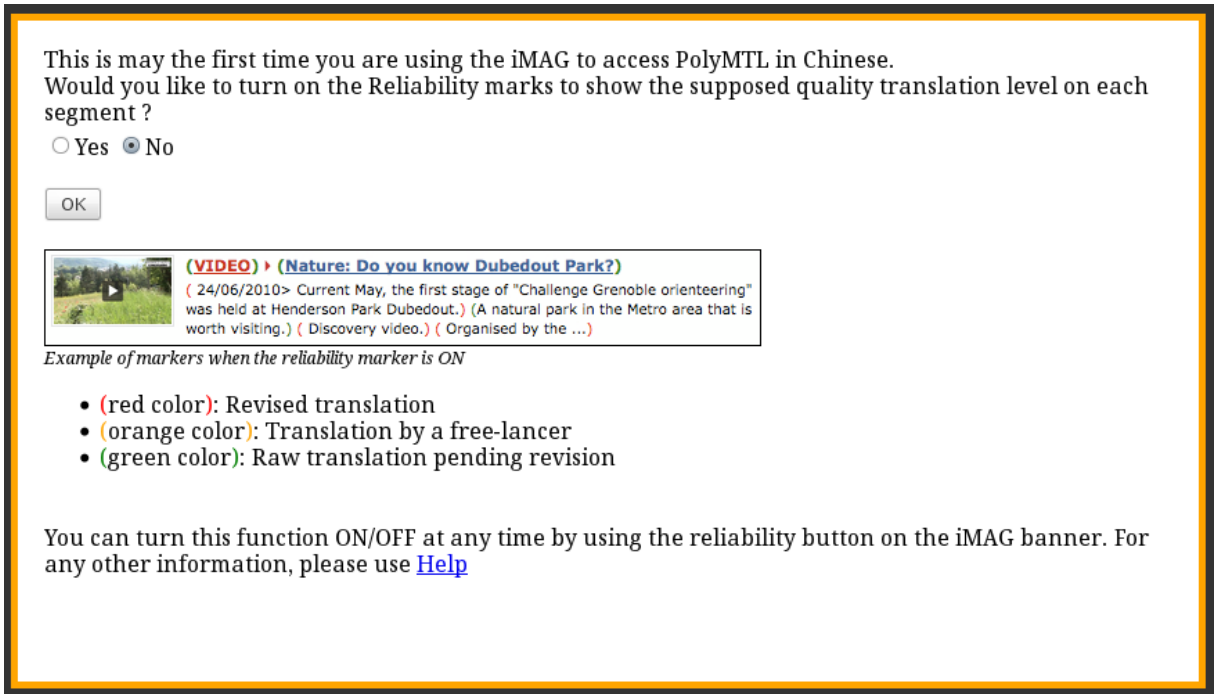

Figure 17 : le message d'avertissement et le formulaires imposés lors de tout accès à une iMAG

Le problème est que SECTra-v2 ne propose pas de moyen direct de récupérer la traduction d'une page Web d'une iMAG. Il faut obligatoirement passer par le message d'avertissement et saisir le formulaire, depuis un navigateur identifié comme un navigateur « classique » (pas un robot), et enregistrer les préférences dans un *cookie*.

7.3.2 Spécifications externes

Un script PHP, installable sur n'importe quel serveur, doit pouvoir servir de proxy vers *aximag.fr*.

7.3.3 Spécifications internes

Le script se fait passer pour un utilisateur qui accède à une iMAG depuis Firefox. Il récupère automatiquement la page d'avertissement avec le formulaire, saisit les options, soumet le formulaire, récupère le *frameset* comportant le lien vers la page traduite, puis la page traduite en question, le tout à l'insu de l'utilisateur, à qui on présent uniquement la page traduite.

Les liens de la page sont réécrits pour passer par Proximag.

Proximag est un service Web, avec les paramètres suivants :

URL : http://<*racine*>/services/sectra2/proximag/

Paramètre	Description	Valeurs	Valeur par défaut
url	**Obligatoire.** URL de la page à traduire.	Une URL. Attention, elle doit correspondre à une URL valide d'une iMAG dans SECTra-v2.	
imag	**Obligatoire.** Nom de l'iMAG.		
sl	**Obligatoire.** Langue source.	Un code ISO-639-2.	
tl	**Obligatoire.** Langue cible.	Un code ISO-639-2.	

7.3.4 Implémentation

C'est un script PHP, qui nécessite le module *curl* de PHP. Il est implémenté et fonctionnel.

7.3.5 EXEMPLE

Figure 18 : affichage transparent du contenu d'une iMAG (anglais→japonais) par Proximag. Noter que dans cet exemple Proximag est hébergé en local (cf. URL), mais en pratique peut être hébergé n'importe où.

7.3.6 BOGUES ET DÉVELOPPEMENTS FUTURS

Il y a des problèmes à la sortie de l'iMAG, ce n'est pas encore transparent. Cela fait une fenêtre d'avertissement de plus à intercepter.

Afin d'être totalement transparent, y compris sur le plan de l'indexation, les liens réécrits doivent présenter une forme « lisible », et être traduits eux aussi dans la langue cible.

8. RÉFÉRENCES BILIOGRAPHIQUES

[1] **Boitet C., Huynh C.-P., Blanchon H. & Nguyen H.-T. (2009)** *A Web-oriented System to Manage the Translation of an Online Encyclopedia Using Classical MT and Deconversion from UNL.* Proc. RIVF 2009, Da Nang, 13-17/7/09, IEEE, P. Bellot ed., 8 p. (extended & updated from eponym CI-2008 paper)

[2] **Falaise A., Kalitvianski R. (2012)** T7o — Étude de la segmentation de documents et première version de SegDoc liée à SECTra-v3. Document L3.4.a, joint au livrable L234.3, projet ANR Traouiero, 15/7/2012 .

[3] **Falaise A., Bellynck V., Boitet C. (2012)**, Progression de la spécification et de l'implémentation de SECTra-v3 et de nouveaux composants pour un module iMAG autonome . Projet Traouiéro, document L2.2.b .

[4] **Falaise A., Bellynck V. (2011)**, Consolidation de SECTra_w et rétro-ingénierie. Projet Traouiéro, document L2.2.a.

[5] **Huynh C.-P., Boitet C. & Blanchon H. (2008)** *SECTra_w : an Online Collaborative System for Evaluating, Post-editing and Presenting MT Translation Corpora.* Proc. LREC-08, Marrakech, 27-31/5/08, ELRA/ELDA ed., 8 p.

[6] **Richard Ishida (2009)**, *Language tags in HTML and XML*, World Wide Web Consortium (W3C). En ligne: http://www.w3.org/International/articles/language-tags/Overview.en.php

[8] **Nguyen H.-T. (2009)** *Des systèmes de TA homogènes aux systèmes de TAO hétérogènes.* Thèse, GETALP, LIG, UJF, Grenoble-1, 230 p.

[9] **Zydroń Andrzej, Saldana Derek** (2009) *Reference Model for Open Architecture for XML Authoring and Localization Version 1.0*, Organization for the Advancement of Structured Information Standards (OASIS). En ligne : http://docs.oasis-open.org/oaxal/V1.0/oaxal-v1.0.html

9. ANNEXES

9.1 COPIES D'ÉCRAN

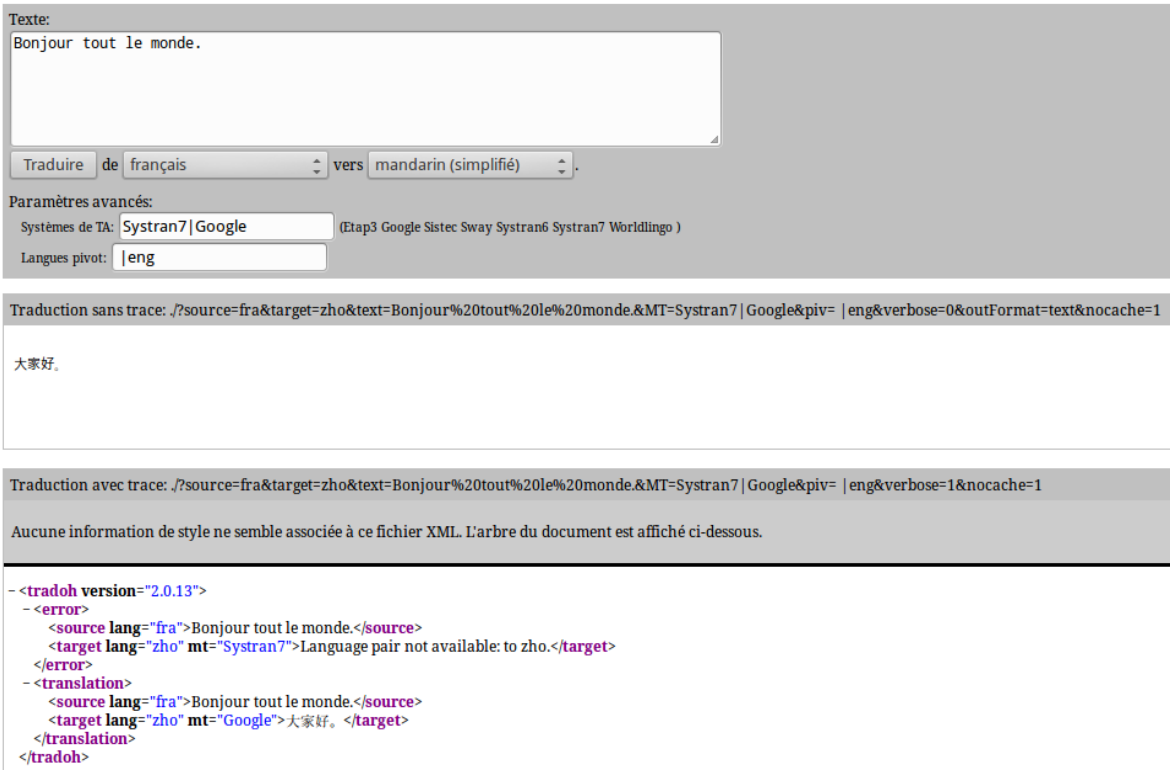

Figure 19 : copie d'écran de l'interface de test du nouveau TRADOH 2.0.14

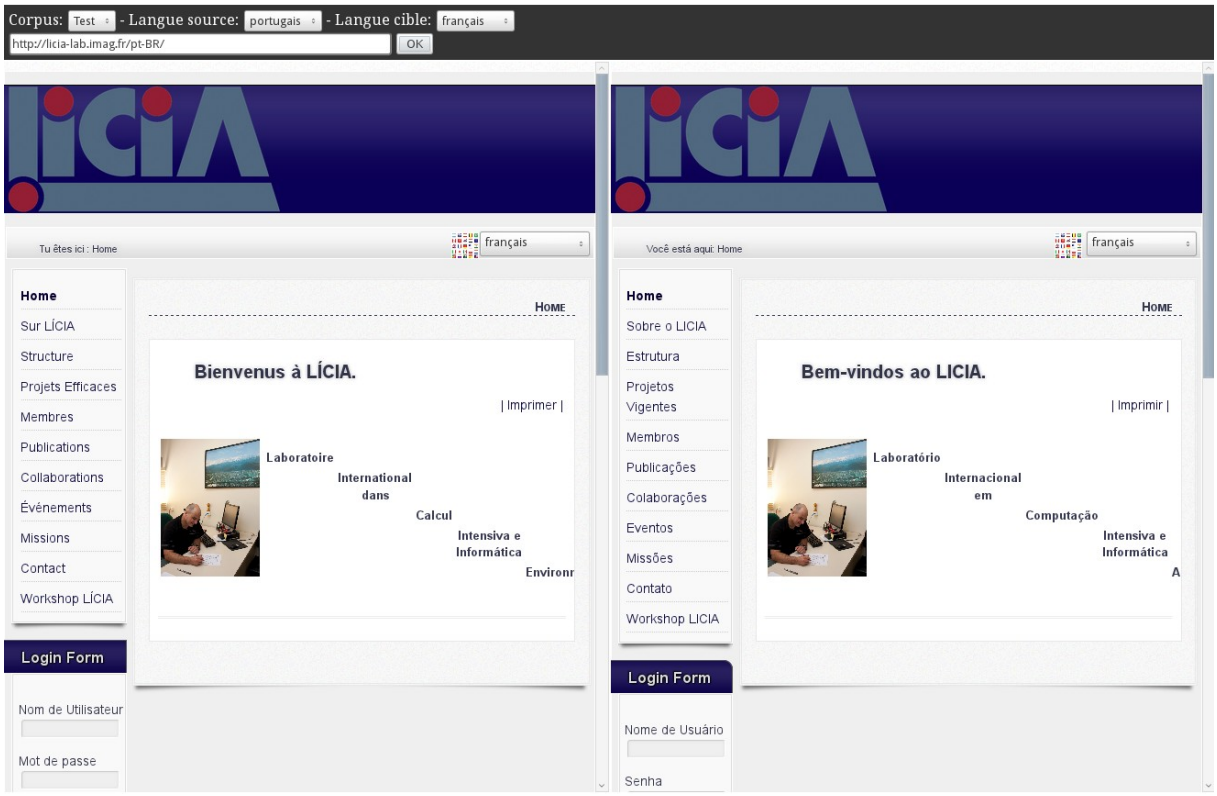

Figure 20 : copie d'écran de l'interface de test du nouveau iMAG 3.0.1