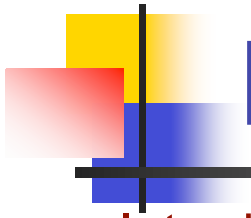
Des systèmes de TA homogènes aux systèmes de TAO hétérogènes

Hong-Thai NGUYEN

GETALP - LIG, UJF

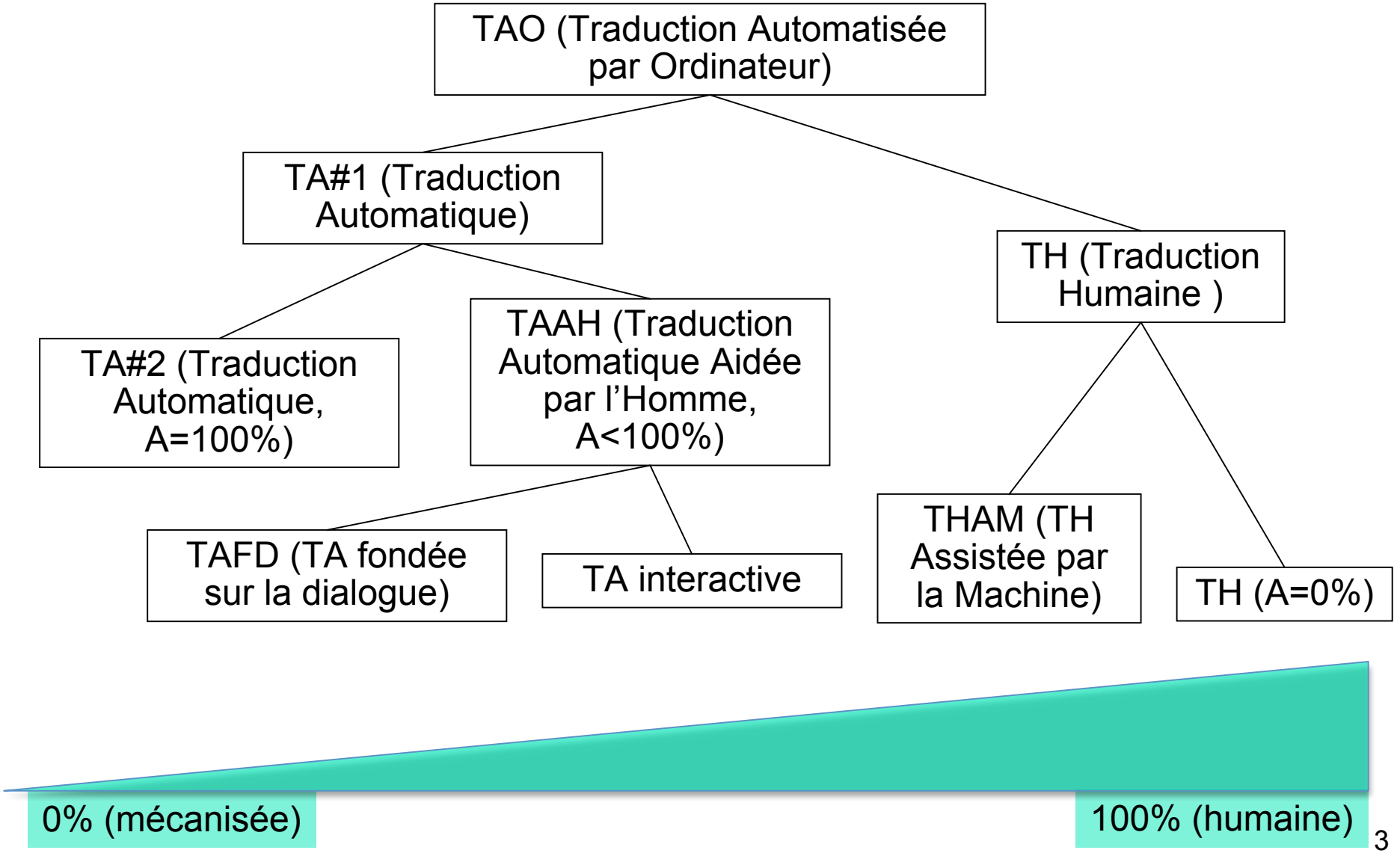
18 décembre 2009



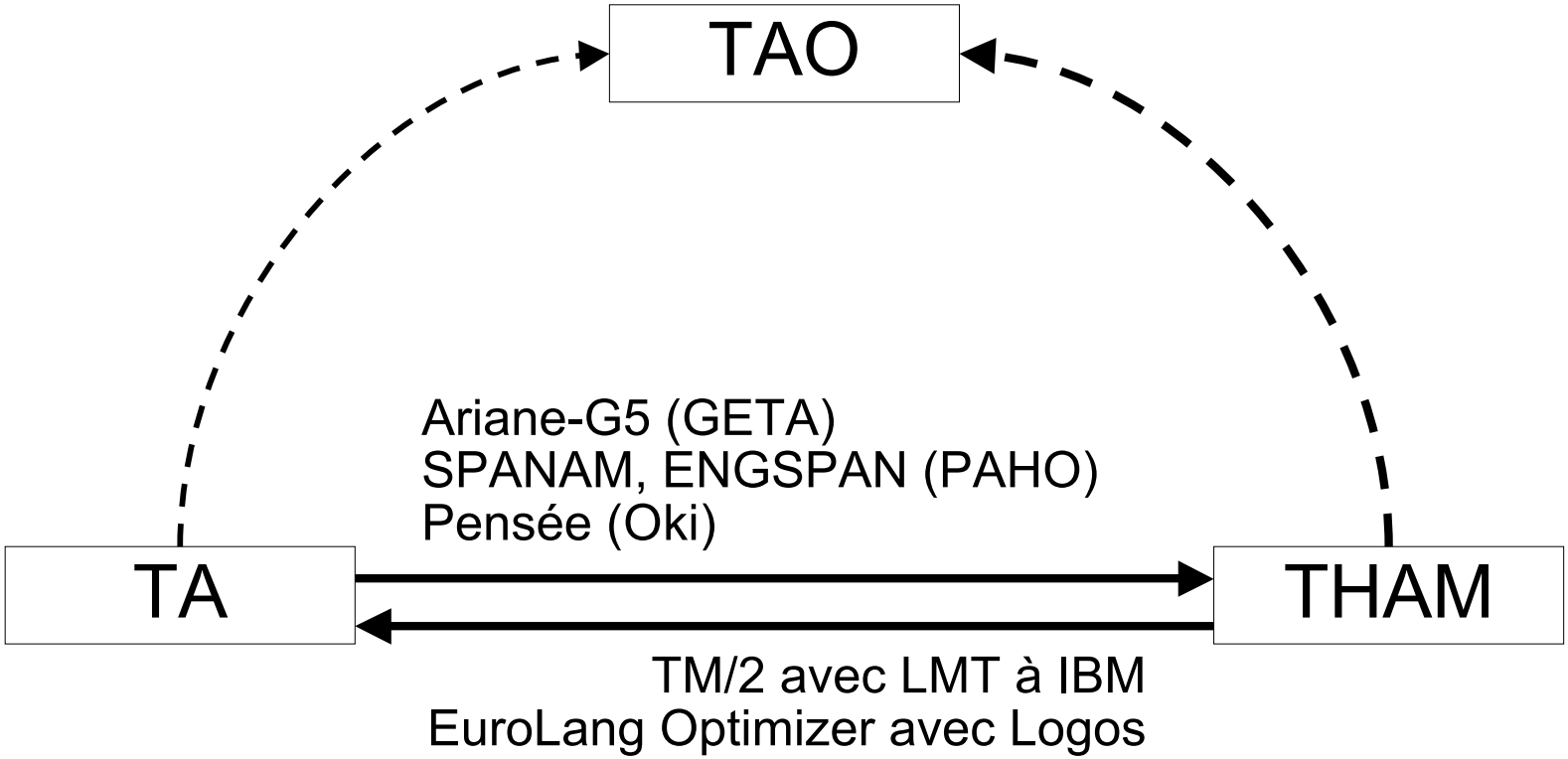
Plan

- Introduction: TA, THAM et TAO hétérogène
- BDLex pour la TAO hétérogène: PIVAX
  - Motivations
  - Architecture linguistique
  - Contrôle sur le partage des données
  - Expérimentation et validation pour le projet U++C et EOLSS
- Méta-EDL & EDL générique: WICALE & EMEU_w
  - Motivations
  - Exemple du méta-langage
  - Application pour le moniteur EMEU_w dans le projet EOLSS
  - Vers un EDL universel
- Réingénierie de LSPL: les systèmes-Q
  - Motivations
  - Réalisation
  - Application pour le projet OMNIA
  - Extensions
- Conclusion & perspectives: généricité, simplicité et flexibilité

Termes


TAO (Traduction Automatisée par Ordinateur)
TA#1 (Traduction Automatique)
TH (Traduction Humaine)
TA#2 (Traduction Automatique, A=100%)
TAAH (Traduction Automatique Aidée par l'Homme, A<100%)
TAFD (TA fondée sur la dialogue)
TA interactive
THAM (TH Assistée par la Machine)
TH (A=0%)
0% (mécanisée)
100% (humaine)

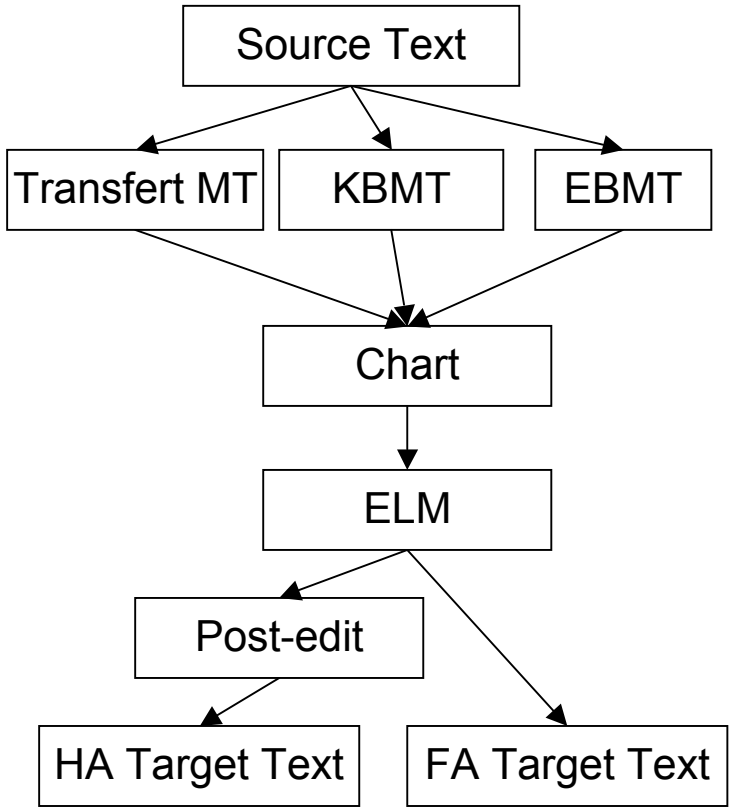
Fonctionnalités hétérogènes



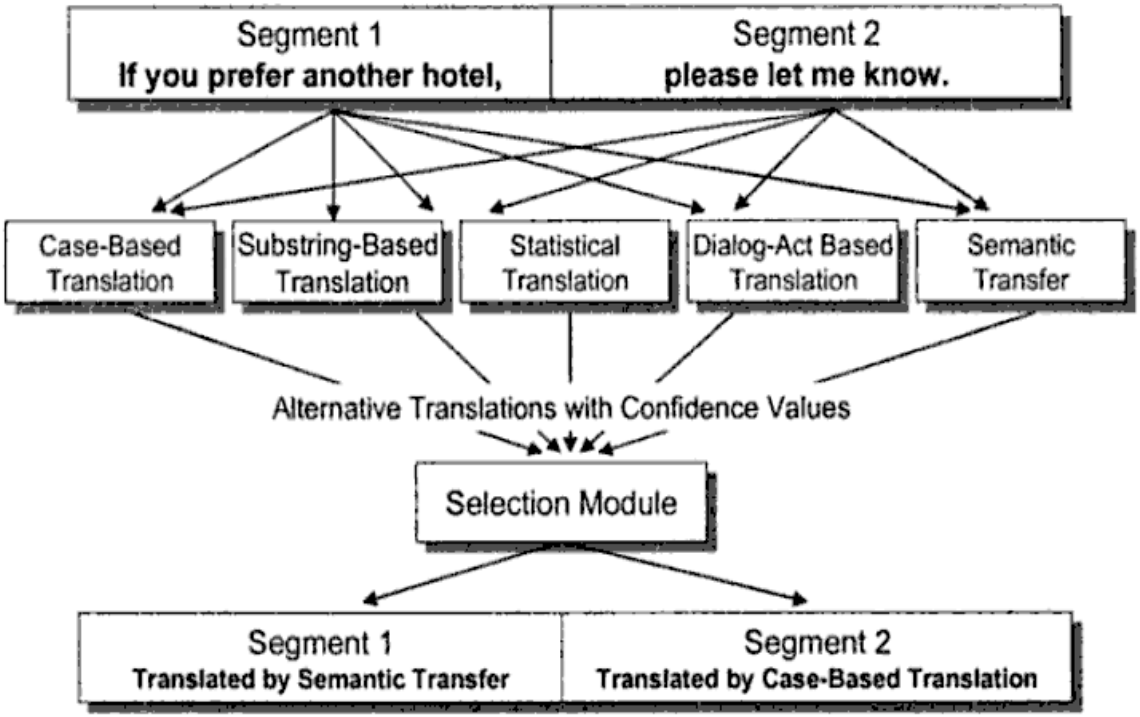
Composants hétérogènes

- Multiplicité (Pangloss, VERMOBIL, ALTFLASH…)
- Hétérogénéité (UNL)

Pangloss [CNU, 1994]

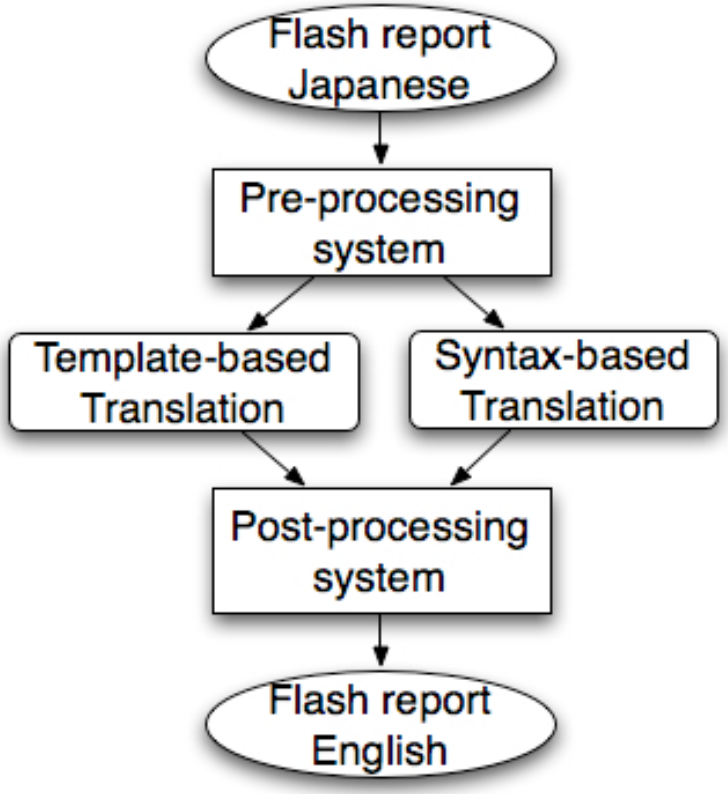


VERMOBIL [Wahlster, 96-2000]

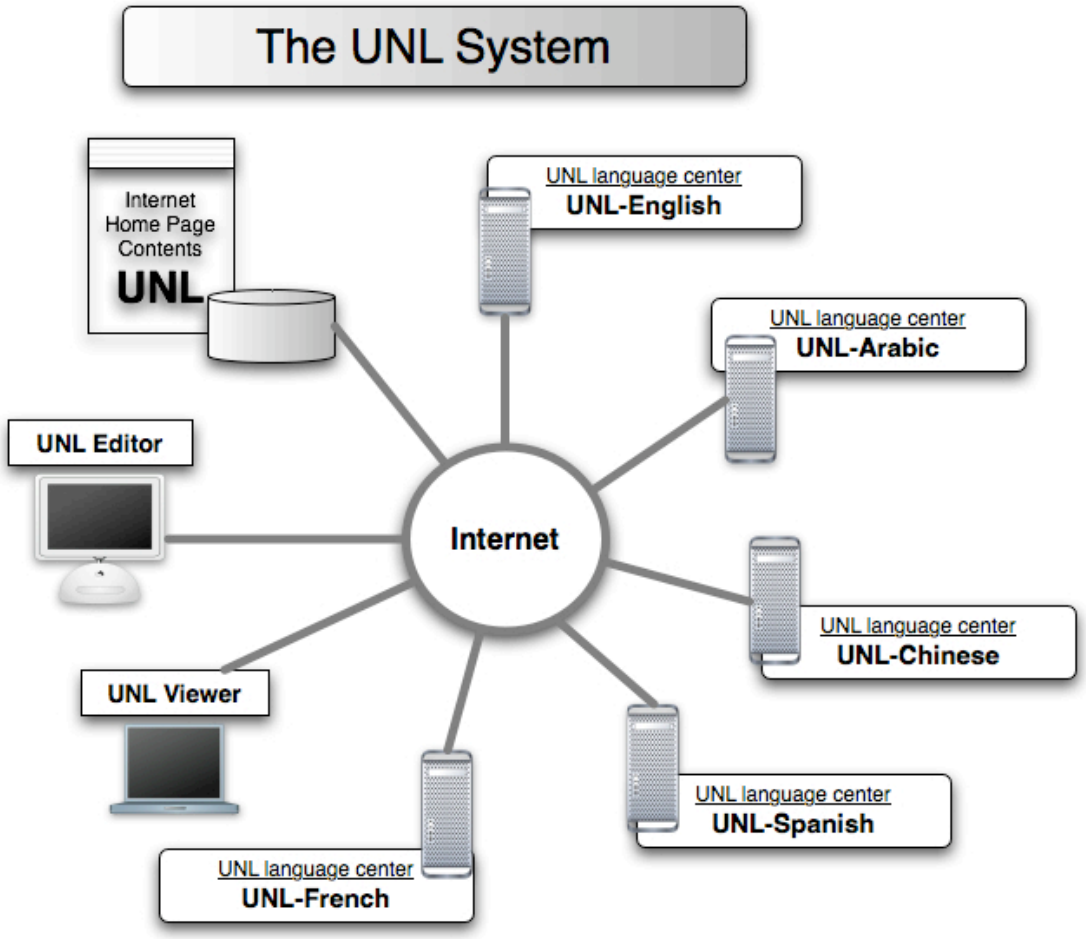


Exemple des systèmes de TAO hétérogènes

ALTFLASH [Uchino et al., 1999]

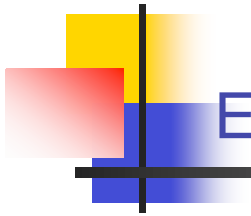


UNL [Uchida, 1996]



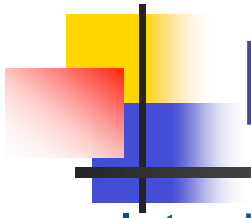
Problèmes liés aux 2 types d'hétérogénéité

- Hétérogénéité des fonctions
  - Pas de synergie lexicale entre TA et THAM
- Hétérogénéité des composants et des ressources
  - Développement distribué
  - Données propriétaires
  - Incompatibilité de forme et de fond entre les représentations et les processus linguistiques



Essai de réponse à ces problèmes: études et réalisations

- Synergie lexicale entre la TA et la THAM
  - PIVAX, base lexicale universelle pour
    - TA hétérogène à pivot lexical
    - TA + TH (TAO)
- Intégration des composants logiciels
  - Vers un système de TAO hétérogène générique
    - Méta-EDL générique (pour le développement): WICALE
    - Moniteur Web générique (pour l'exploitation):EMEU_w
    - Intégration ou réingénierie des LSPL dans un système de TAO hétérogène générique: les systèmes-Q



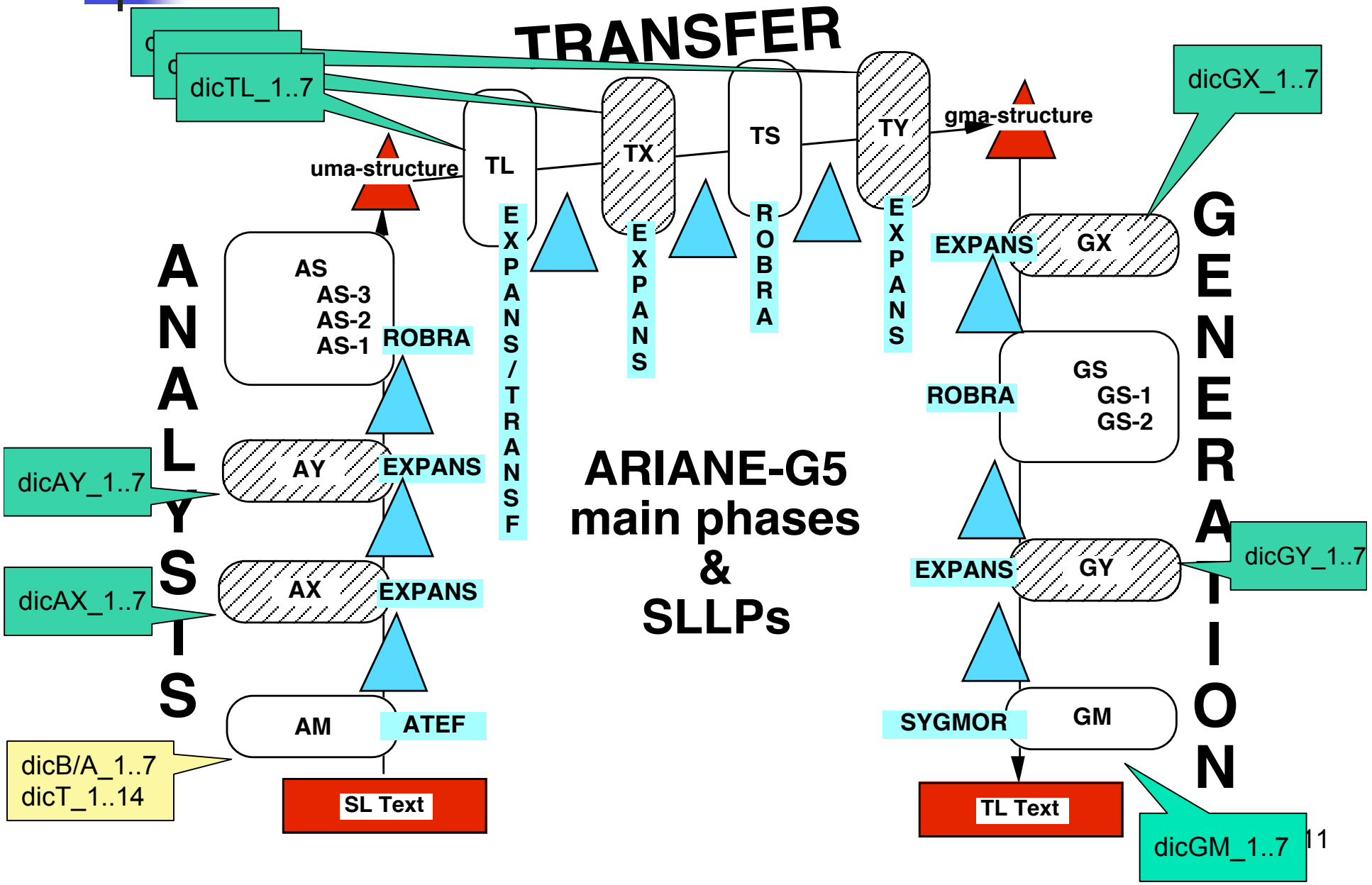
Plan

- Introduction: TA, THAM et TAO hétérogène
- BDLex pour la TAO hétérogène: PIVAX
  - Architecture linguistique
  - Contrôle sur le partage des données
  - Expérimentation et validation pour le projet U++C et EOLSS
- Méta-EDL & EDL générique: WICALE & EMEU_w
  - Motivations
  - Exemple du méta-langage
  - Application pour le moniteur EMEU_w dans le projet EOLSS
  - Vers un EDL universel
- Réingénierie de LSPL: les systèmes-Q
  - Motivations
  - Réalisation
  - Application pour le projet OMNIA
  - Extensions
- Conclusion & perspectives

Traiter “tous les dictionnaires de TA” ?
Trop ambitieux

- Nombreux (chaque "phase" peut en avoir plusieurs)
  - de 14 à 70 dans un système écrit en Ariane-G5
  - 60 fichiers dictionnaire principaux pour 1 couple de langue de Systran
  - plusieurs dictionnaires spécialisés pour un système de TA du commerce
- Problème de consistance & complétude
  - Possible pour un système de TA homogène (LMT, METAL, BDTAO d’Ariane)
    - Gérer tous ses dictionnaires dans une BDL
    - & compiler le code spécifique depuis ces fichiers
- Impossible pour un ensemble arbitraire de systèmes
  - Problème 1: différences dans l’information lexicale et dans la structure
  - Problème 2: différences dans le modèle lexical

Ariane-G5 (1978-): structure



Exemple d'un dictionnaire

```
(Flexions verbales)
UL          Condition          Affectation          Chaîne)
FLEX2       ==                 /          /.
            ==TPRSS                       /         /ES,
            ==SVINGS                      /         /ING,
            ==SVINGSG                     /         /ING'S,
            ==SVINGPG                     /         /INGS,
            ==SVINGP                      /         /INGS',
            ==SVING            /          /ING,
            ==SVVENG                      /         /ED'S,
            ==SVVENP                      /         /EDS,
            ==PRTPAST                     /         /ED,
```

GM de FVX-ENX (français-anglais) en SYGMOR

```
'ALL'  == ==        /  /     /''TOUT'   ,$INT,$KADDEICT.
'ALLOF'  ==         /        /'TOUT'    ,$INT,$KADDEICT.
```

TL de ENX-FVX (anglais-français) en EXPANS/TRANSF

Systran (EN-AR)

Monolingual (EN):
EN
- Alignment
  - alignment-enar.xml
  - TableCodage.ENSQ
  - TableCodage.ENSH
  - TableCodage.ENUR
  - TableCodage.ENRU
  - TableCodage.ENPT
  - TableCodage.ENPL
  - TableCodage.ENSV
  - TableCodage.ENNL
  - TableCodage.ENJA
  - …
  - TableCodage.ENKO
  - TableCodage.ENDE
  - TableCodage.ENES
  - TableCodage.ENIT
  - TableCodage.ENHU
  - TableCodage.ENFA
  - TableCodage.ENEL
  - TableCodage.ENDA
- Guess

Homography
- EN_Table.HM
- Inflection
  - EN_Table.DETPRO
  - EN_Table.NP
  - EN_Table.V
  - EN_Table.N
  - EN_Table.A
- Mono
  - lemma-en.lst
  - te_stopwords-en.lst
- Postprocess
  - postprocess_XX-en.lst
- Translation
  - compent-en.lst
  - vk-en.lst
  - seg-en.lst
  - norm_ja-en.lst
  - norm_cjk_cisco-en.lst
  - norm_cjk-en.lst
  - norm-en.lst
  - lookup-en.lst
  - loca-en.lst

….

Bilingue (ENglish-ARabic)
Transfer
- ENAR
  - compounds-enar.txt
  - possessive_pronouns-enar.lst
  - personal_pronouns-enar.lst
  - ordinal_numbers-enar.txt
  - transfer-enar.txt
  - dates-enar.lst
  - negation-enar.lst
  - frozen_compounds-enar.txt

Exemples de dictionnaires ENAR (SYSTRAN)

```
#$Id: lemma-en.lst,v 1.9 2005/10/27 15:20:34 rebollo Exp $
## id     lemma     cat     info     morpho
27869     circlip     N     +CT+CON+DEVICE+ATTACH+CHAR     N1
27870     circuit     N     +ABS+CON+CT+CIRC+RTES+CHAR+ABR=(ckt)     N1
27871     circuitry N     +CON+CT+MS     N5
27874     circularity     N     +GRDPRP+ABS+MS+PROPTY     N5
#27875   circulate N     +C-NCO N1
27876     circulation     N     +PROGEN+ABS  N1
```

lemma-en.lst

```
        # ENAR Txt Transfer Dictionary
# SRCLEMMA   SRCPOS   TGTLEMMA    TGTPOS   MODIFICATIONS    NOTES
# MODIFICATIONS = (SRCLEMMA|SCRPOS|TGTLEMMA|TGTPOS)(\+SRCLEMMA|\+SCRPOS|\+TGTLEM
...
10947    circlip          noun        حلقة            noun:common
10948    circuit          noun        دارة            noun:common
10949    *circuit         verb        دار             verb:plain
10950    circuital        adj         لولبي           adj:base
10951    circuitry        noun        دارات كهربائية   noun:common
10952    circular         adj         دائري           adj:base
10953    circular         noun:common دائري           noun:common
10954    circularisation noun        تثبيت           noun:common
10955    circularity      noun        استدارة         noun:common
10956    circularize      verb        دوّر             verb:plain
...
```

transfert-enar.lst

Fichiers lexicaux dans les systèmes de TA

- Diversité
  - Macrostructure
    - ARIANE : Machine → disque → paire L1-L2 → étape → phase → DICT (texte)
    - SYSTRAN : paire L1-L2 → Dict* (≠ types) → DICT (texte)
    - CICC/EDR : Dict → paire L1-L2 → DICT (texte)
    - CAT2 : Phase → paire L1-L2 → DICT (texte)
  - Microstructure UL: unité lexicale
    - Ariane-G5 : morphe, lemme, UL, UL + sens [numéro]
    - LMT, KANT : lemme, sens
    - UNL : sens
    - ETAP-3: : morphe, lemme, UL (FLS), sens
    - MU : forme, lemme, sens
    - SYSTRAN : forme, lemme (terme, idiome), UL (restrict)

Différences entre ”espaces lexicaux"
organisation, code

- Différentes architectures linguistiques, théories lexicales (unité lexicale est ≠)
  - Plusieurs systèmes écrits en Ariane-G5 :
    - Lemme, et UL (famille dérivationnelle) ("prolexeme")
      - REPAIR_V = {repair_V, repair_N, reparation_Nact, reparable_PPA, reparably_PPAdj, repairer_Nagt}
  - ETAP-3 :
    - Lemme, ou UL enrichie (avec les fonctions lexico-sémantique)
  - UNL (pour les centres de langue utilisés outils DeCo et EnCo) :
    - LN : forme, lemme
    - UNL : UW = [un ensemble de] « *lexème interlingue* »
- Différents types de syntaxe pour les fichiers dictionnaires
  - ARIANE : 3 syntaxes
    - ATEF, TRANSF/EXPANS, SYGMOR
  - SYSTRAN : format textuel pour chaque dictionnaire+ X-Dict
  - LMT, PT : syntaxe basée sur Prolog/VM
  - MU : CommonLisp de Kyodai (Kyoto University)
  - Etc. : METAL, Reverso, ATLAS, AS-Transac…

Donc: construire une BDLex pour un système de TAO hétérogène est

possible pour des cas particuliers

- Systèmes homogènes, de TA + THAM

(quasi) impossible pour N systèmes

- S'ils utilisent des architectures linguistiques différentes

Bases de données lexicales (BDLex)

- Pour la TA
  - Peu de DBL pour les systèmes de TA
    - longue histoire, problème difficile
  - Autour d'Ariane-G5 (Grenoble)
    - Visulex (1982) consolidation d'info lexicale d'une PL d'Ariane-G5
    - BDTAO (1992) B'Vital firm (French National MT projet "Esope")
  - projet MU (Kyoto)
    - vraie BDLex en Adabase (Lisp)
  - SYSTRAN & autres éditeurs des systèmes de TA
    - Solutions partielles
- Pour la TH
  - La plate-forme Jibiki [Sérasset, 2005]
  - Certaines BDLex créées: Papillon, LexAlp, GDEF, …

Idées principales dernière la construction de PIVAX

- PIVAX: base de données lexicales pour les systèmes de TA partageant un même pivot lexical
- Limitation aux systèmes de TA utilisant un même pivot lexical
  - Première application à UNL/U++,
    - construire un BDLex générique
  - Accepte les autres pivots lexicaux, comme IF (Nespole!)
- Possibilité de ne pas partager les informations "propriétaires"
  - Manipuler seulement la partie "publique" de l'info:
    - "Vocable", Lemme (avec POS), sens # (lexie id)
  - Considérer l'information "propriétaire" ou "privé" comme commentaire
- Développement en Jibiki [https://ligforge.imag.fr/projects/jibiki/]
  - Service Web
  - Modélisation de la macro et microstructure
  - Fonctions implémentées
    - gestion d'utilisateurs, gestion de version, génération de l'interface d'édition, gestion de la représentation des entrées par XSLT…

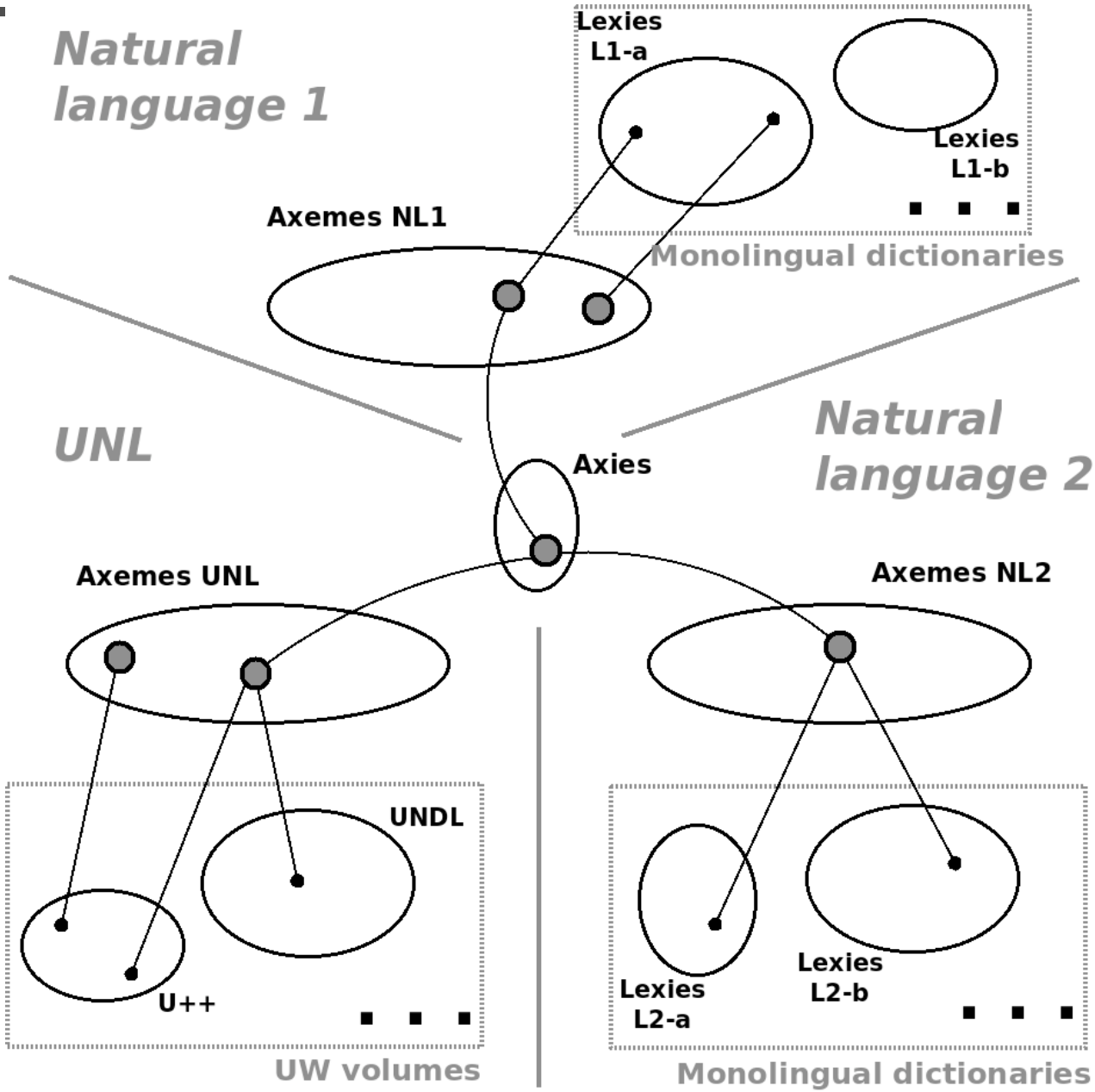
Quelques détails

Macrostructure et microstructures

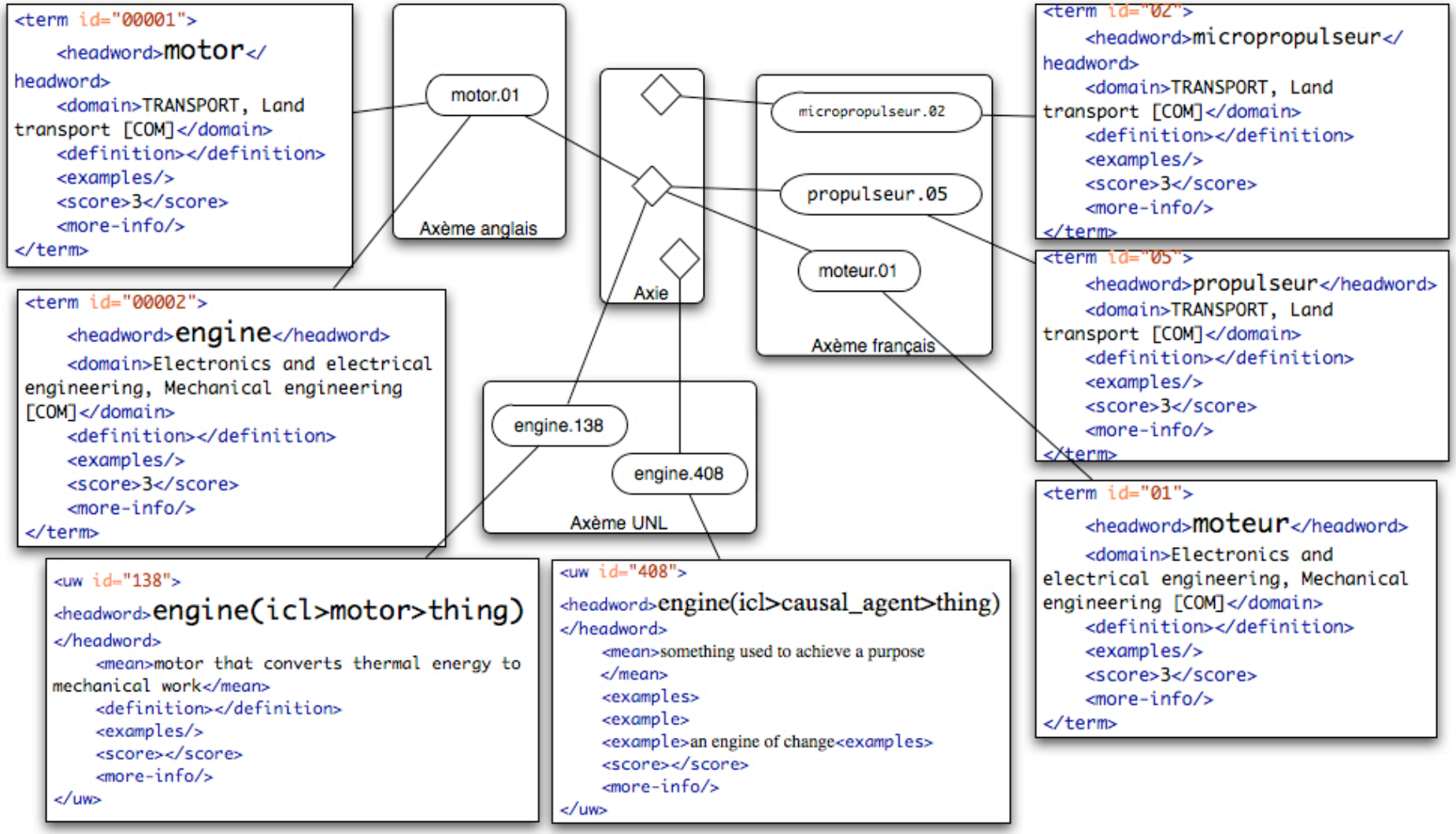
Exemple de données

Validation dans le projet U++C

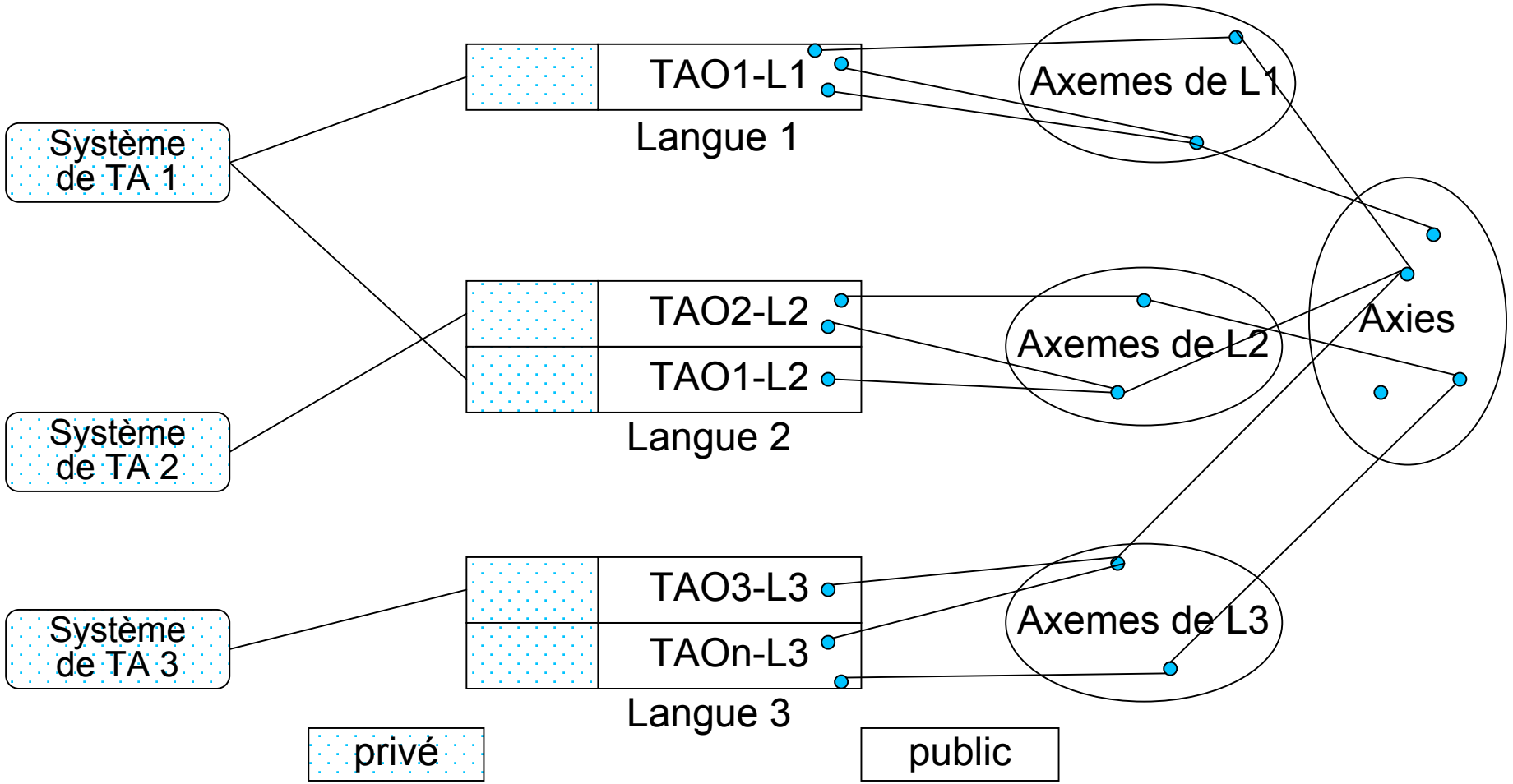
Macrostructure d'une base PIVAX



Exemple tiré des entrées



Données lexicales publiques et privées

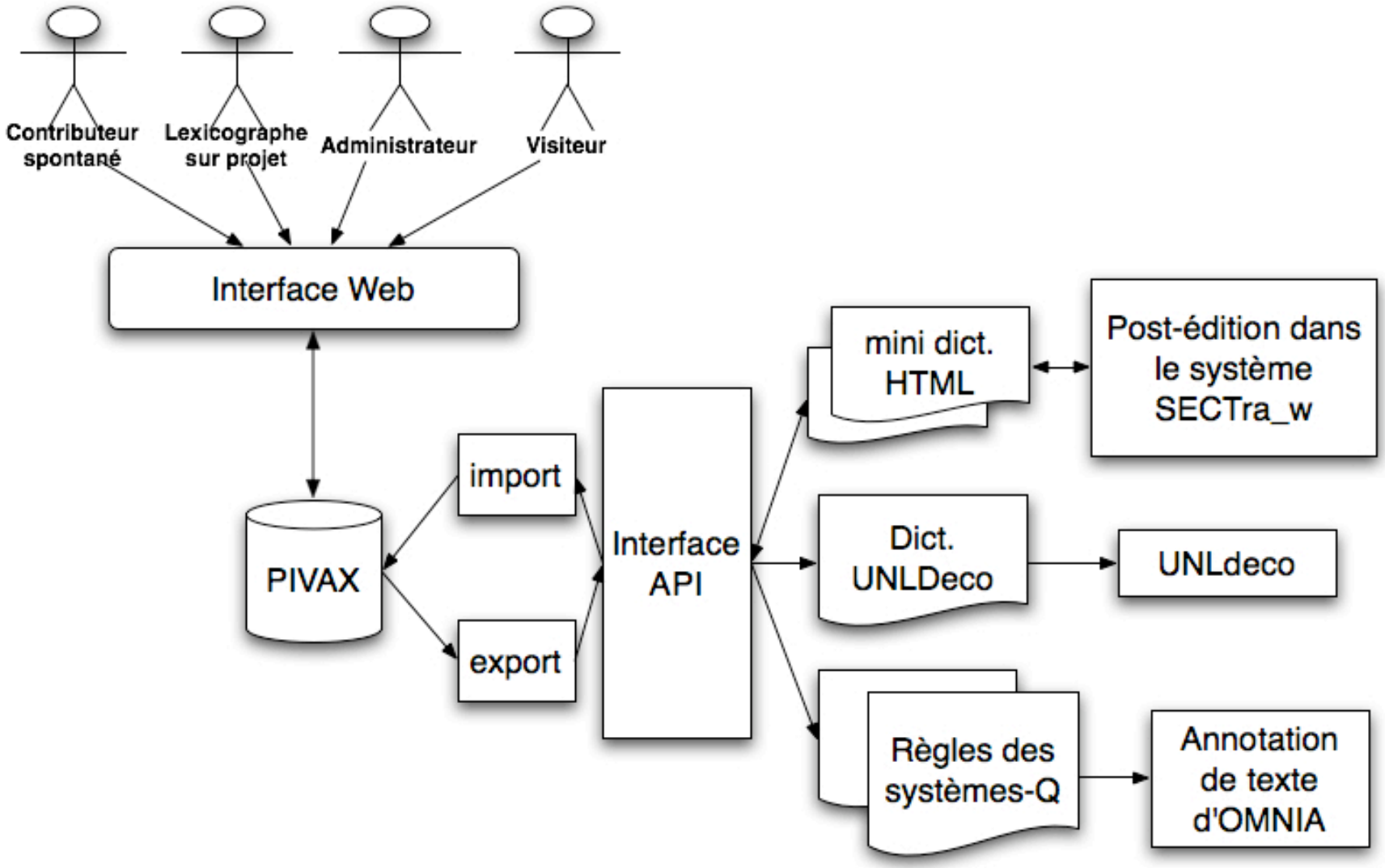


Une entrée PIVAX

dans un volume français de Systran

```
<d:data>
    <p:lexie p:id="lexie.systran.fier.1200
  p:process_status="UNPROCESSED" p:status="UNKNOWN"
  p:owner="Systran" p:score="0.00003">
        <p:lemma p:access="public">
            fier
        </p:lemma>
        <p:class p:access="public">
            Adj
        </p:class>
        <p:comment p:access="public" >
            "fier" can also be a V ("se fier à")
        </p:comment>
        <!--  ..... partie propriéaire  .....  -->
        <p:proper_information p:access="hidden">
            <!-- Systran proprietary codes -->
         </p:proper_information>
    </p:lexie>
</d:data>
```

PIVAX, un serveur lexical



Application à U++C

Ressource	Type/Langue	Type	Nombre des entrées
UNL-Deco	UNL-FRA	bilingue	**39 389**
PARAX	UNL-FRA	bilingue	**18 978**
PARAX	UNL-CHN (données par Pr. ShiX en 2001)	bilingue	**9 315**
PARAX	UNL-RUS	bilingue	**13 817**
PARAX	UNL-ESP	bilingue	**3 833**
UNLKB	UNL	monolingue	**21 618**
UPM UW++	UW avec définition et exemples	monolingue	**207 009**
UNDL	Projet EOLSS-UNL-FR/UW	monolingue	**21 354**
IATE	Projet EOLSS/ENG, termes correspondant aux 21K UW récupérés par Robodico	bilingue	**255 305**
IATE	Projet EOLSS/FRA, termes correspondant aux 21K UW récupérés par Robodico	bilingue	**258 175**

Utilisation d'un serveur de PIVAX pour la THAM

Navigation inspirée de PARAX

Structure, interface (1 PARAX example)

Utilisation directe (interface Web)

Intégration dans le projet de traduction EOLSS

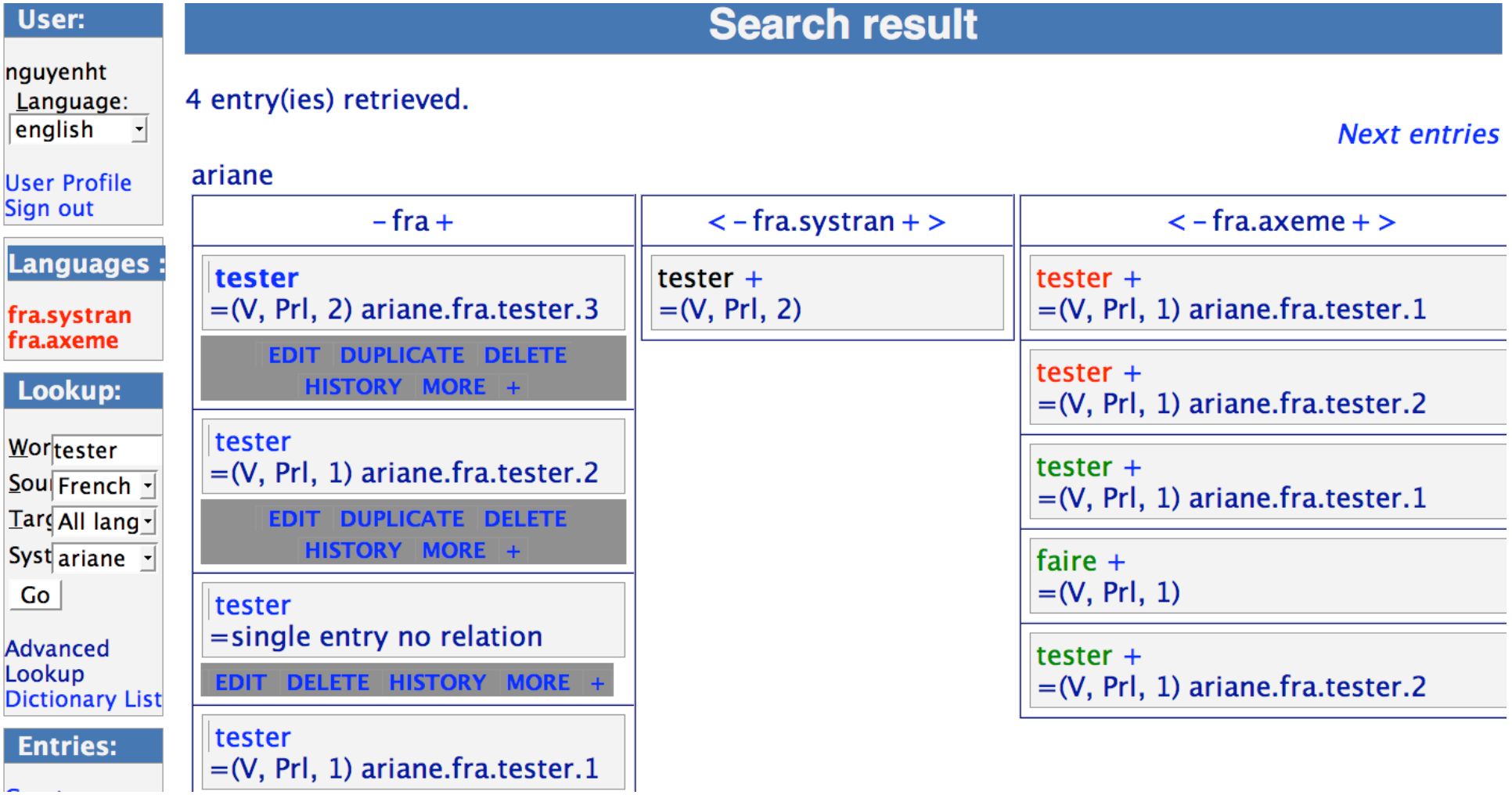
Interface et fonctions

- Navigation à la PARAX mais dans l'environnement Web
  - Simplicité
    - PIVAX crée la page HTML/Form
    - Représentation de liens par couleurs
      - Trop difficile / lourde pour manipulation directe de "graphe lexical"
  - Flexibilité
    - contrôle le largeur et la disposition de colonne
    - Caché / affiché colonnes
- Implémentation en Jibiki
  - Gestion d'utilisateurs
  - Édition en mode à la Wiki
  - Génération d'interface
    - Présentation de résultat de recherche
    - Recherche multicritères
    - Édition
    - Import & export
  - Support à l'interface multilingue (anglais, français …)

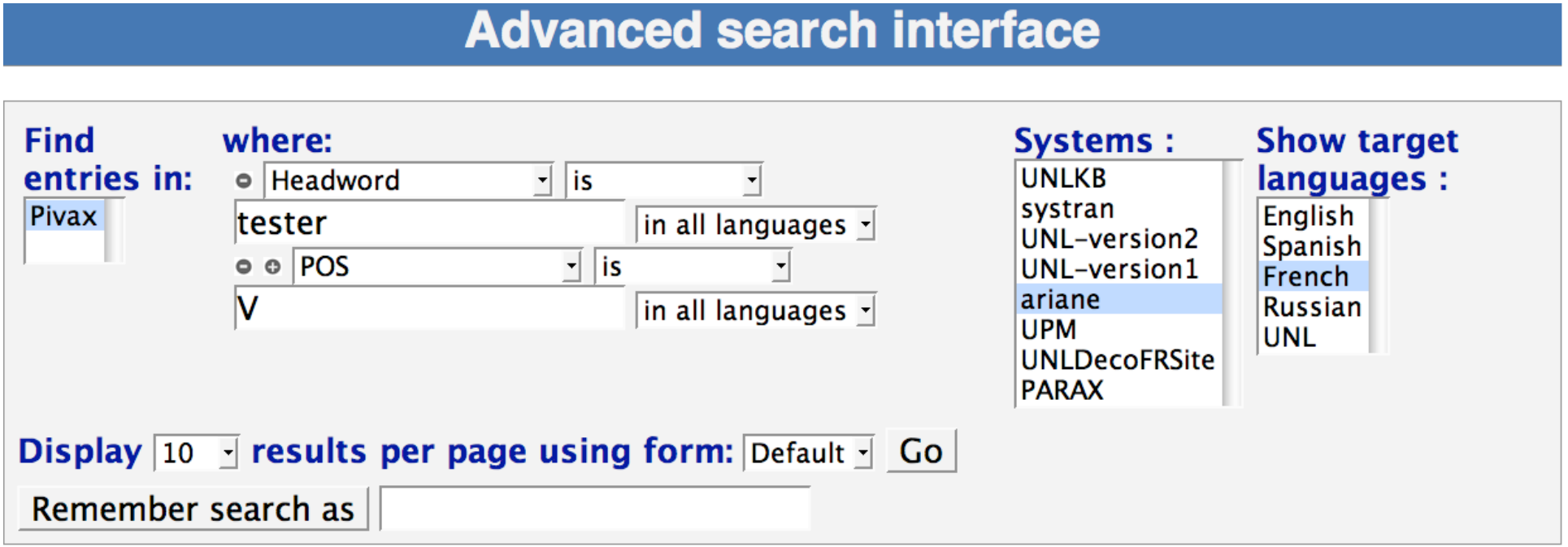
Interface de consultation de PARAX

(exemple avec 3 volumes: JAP, RUS, FRA)

Interface de navigation de PIVAX
inspirée de PARAX

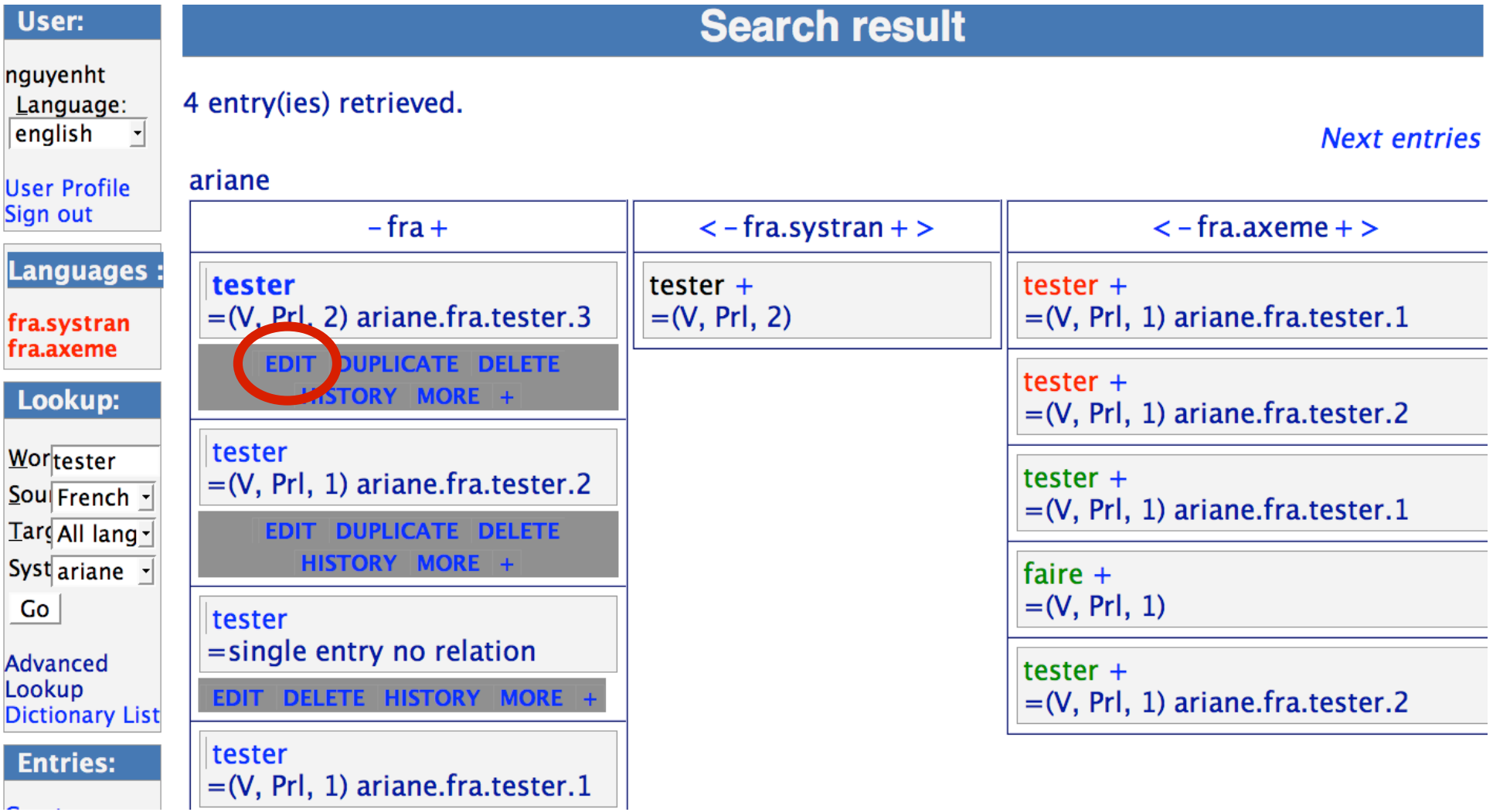


Interface de recherche avancée de PIVAX

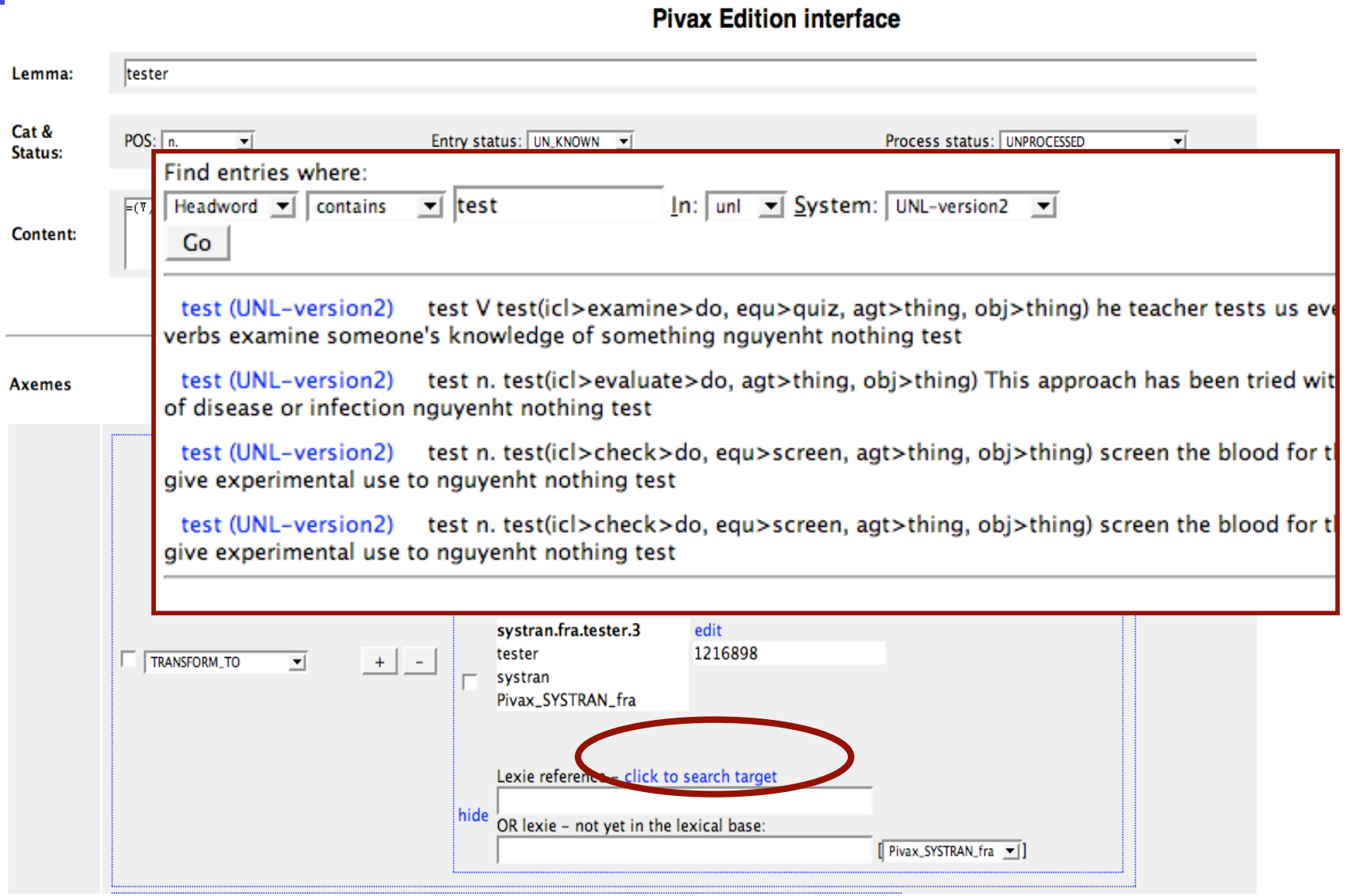


Interface de contribution

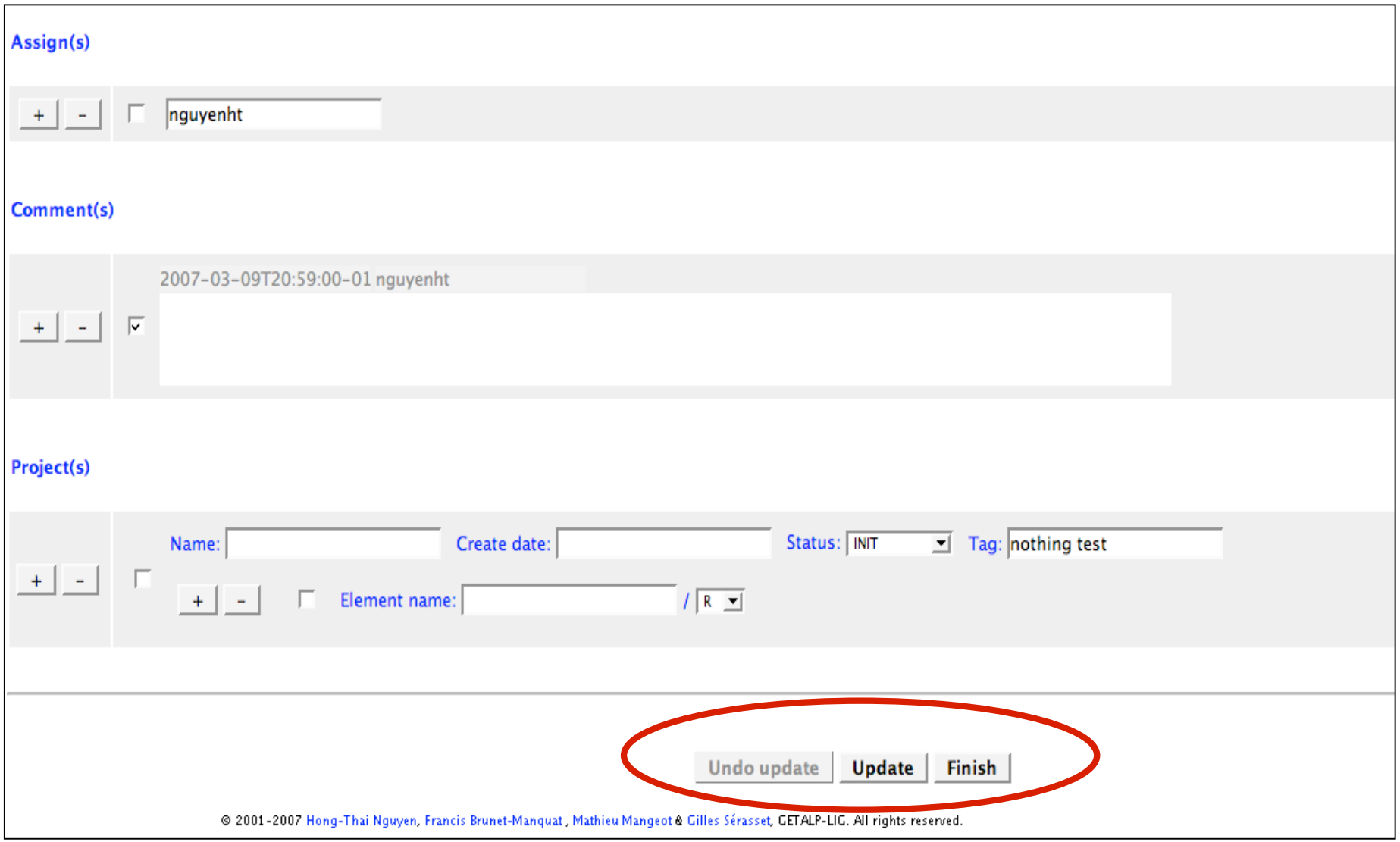
depuis la navigation



Contribution (1)

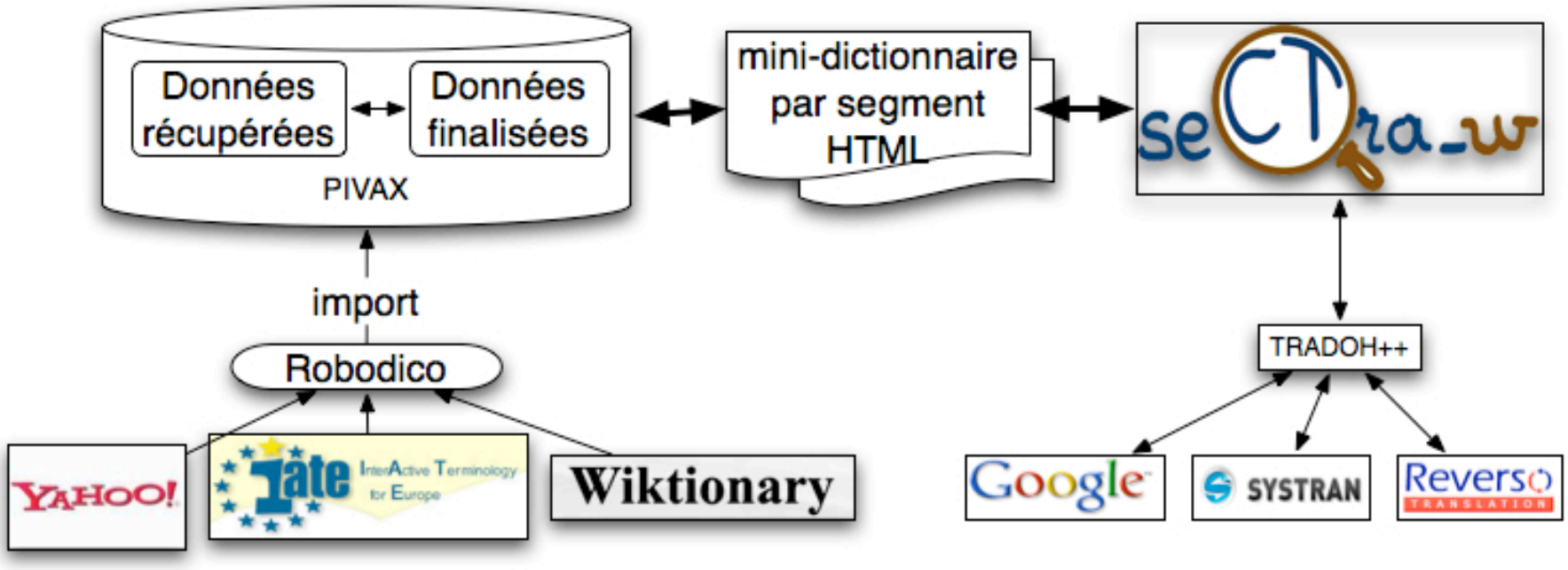


Contribution (2)

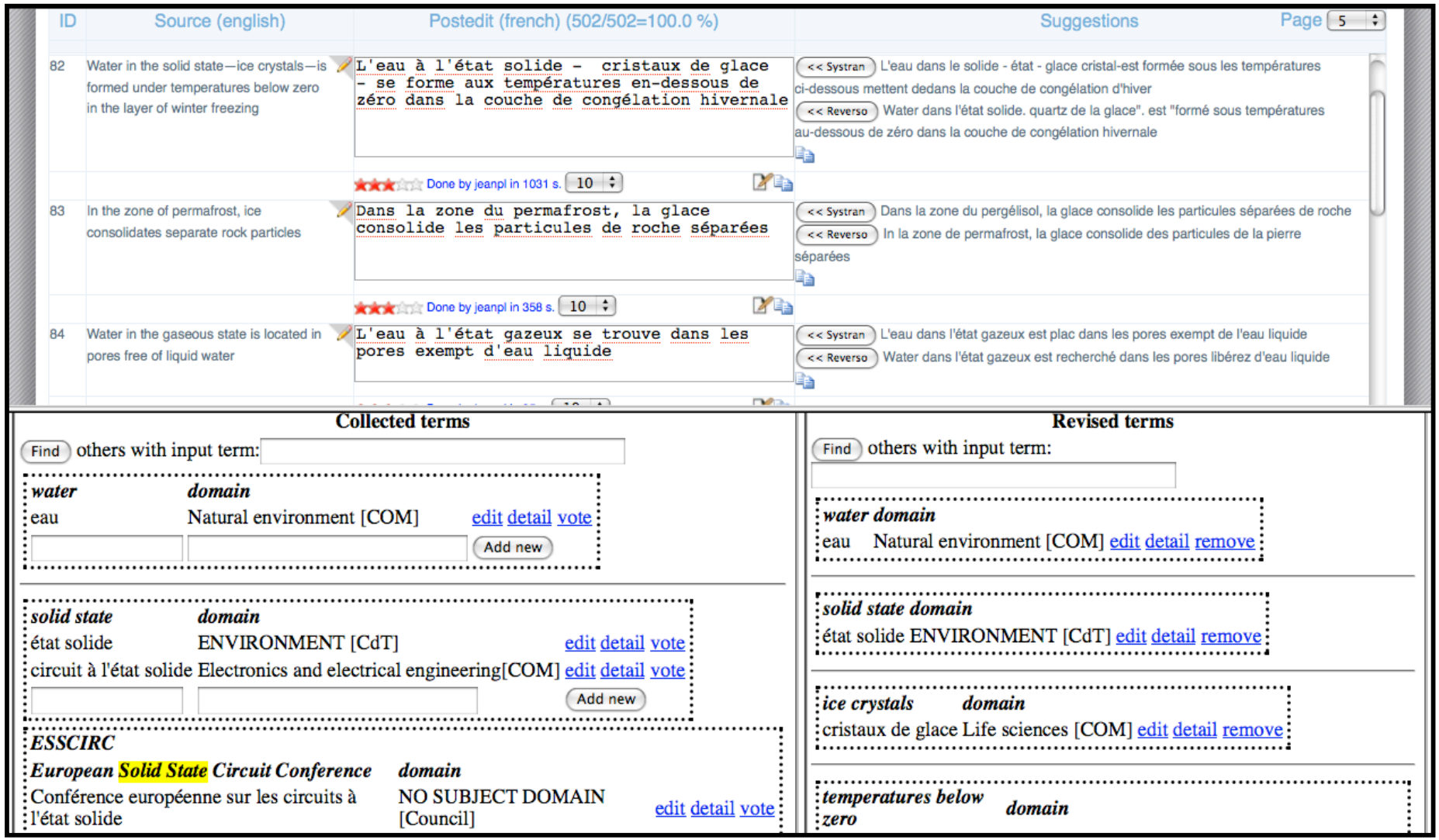


Intégration dans un système de THAM

- Projet EOLSS (Encyclopedia of Life Support Systems)
- Support lexical
  - Préparation des ressources lexicales terminologiques
    - Consultation automatique de IATE, Yahoo Term par Robodico
    - Stockage dans PIVAX
  - Intégration
    - Mini-dictionnaire pour chaque segment
    - Calcul à l'avance
    - Prise en compte des contributions lors de la post-édition

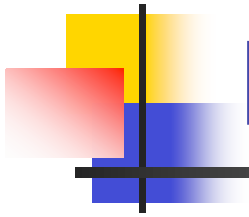


Prototype d'intégration



Conclusion pour PIVAX

- Une première BDLex collaborative pour la TA hétérogène utilisant un « pivot lexical »
  - Architecture linguistique: lexie, axème, axie
  - Partage partiel des données
- Usage multiple entre la TA et la THAM
  - Utilisation de la plate-forme Jibiki
  - Navigation à la PARAX
  - Intégration à la TH
- Application dans les projets U++C/UNL, EOLSS, OMNIA



Plan

- Introduction: TA, THAM et TAO hétérogène
- BDLex pour la TAO hétérogène: PIVAX
  - Motivations
  - Architecture linguistique
  - Contrôle sur le partage des données
  - Expérimentation et validation pour le projet U++C et EOLSS
- Méta-EDL & EDL générique: WICALE & EMEU_w
  - Motivations
  - Exemple du méta-langage
  - Application pour le moniteur EMEU_w dans le projet EOLSS
  - Vers un EDL universel
- Réingénierie de LSPL: les systèmes-Q
  - Motivations
  - Réalisation
  - Application pour le projet OMNIA
  - Extensions
- Conclusion & perspectives: généricité, simplicité et flexibilité

EDL classique

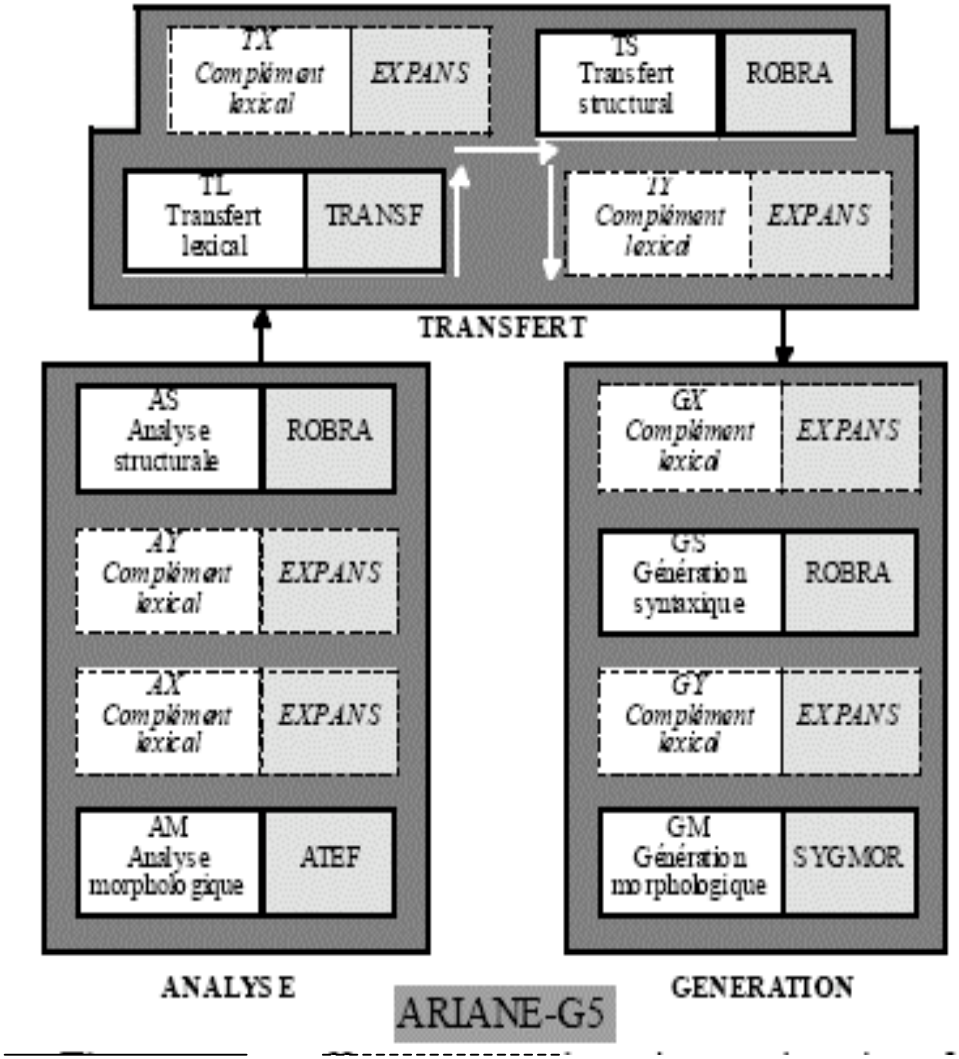
caractéristiques générales

(1976) C'est un environnement de programmation linguistique qui connecte ou intègre un ou plusieurs LSPL (langage spécialisé pour la prog. linguistique).

Un EDL permet aux développeurs linguistes de construire et mettre au point des applications « langagières » (gestion, manipulation des linguiciels ou données, compilation, test, débogage).

	EDI (IDE en anglais) (Environnement de Développement Intégré)	**EDL** (Environnement de Développement Linguistique)
Exemple	ECLIPSE, JDeveloper(Java), Visual Studio (Visual Basic, Visual C++), …	Ariane-G5 (ATEF, ROBRA …)
Utilisateur	Programmeur	linguiste, lexicographe, gestionnaire, utilisateur …
Langage	Langage de programmation	LSPL (Langage Spécialisé pur la Programmation Linguistique)
Type de composant	variables, procédures, modules, …	variables, grammaires, dictionnaires, …
Taille de composant	assez petit	grande (dictionnaire, grammaire)
« Entrée »	Donnée	Corpus, texte
Cycle de développement de ressource	plutôt stable	en perpétuelle évolution
Génie	Génie logiciel	Génie linguiciel

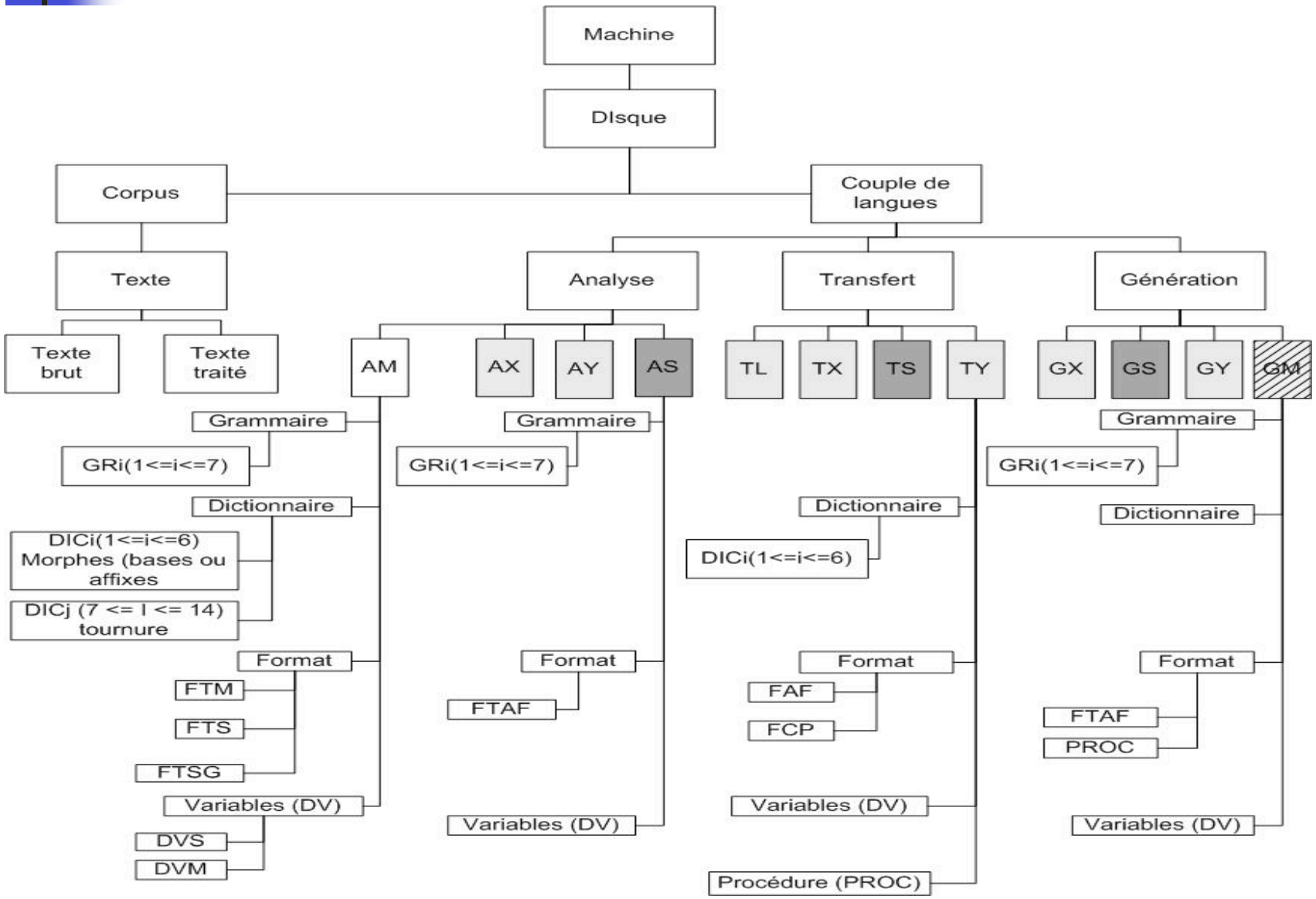
Exemple : Ariane-G5



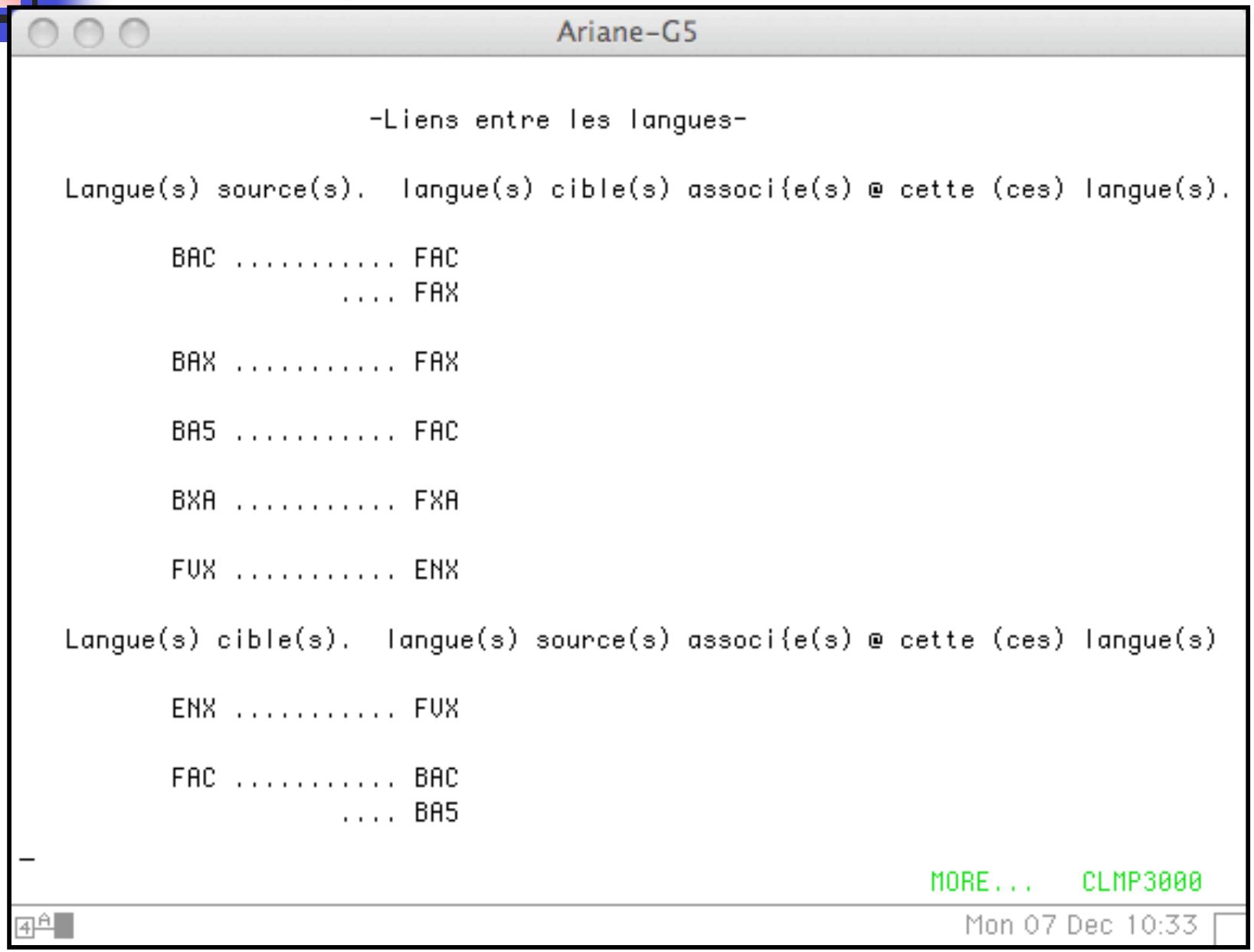
Points spécifiques:

- Chaînes d'exécution (mise au point)
- Chaînes de production
- LSPL (5 dans ARIANE-G5)
- Composants
- Accessibilité par le réseau LIDIA: http, socket, smtp

Objets manipulés par un EDL
exemple d'Ariane-G5



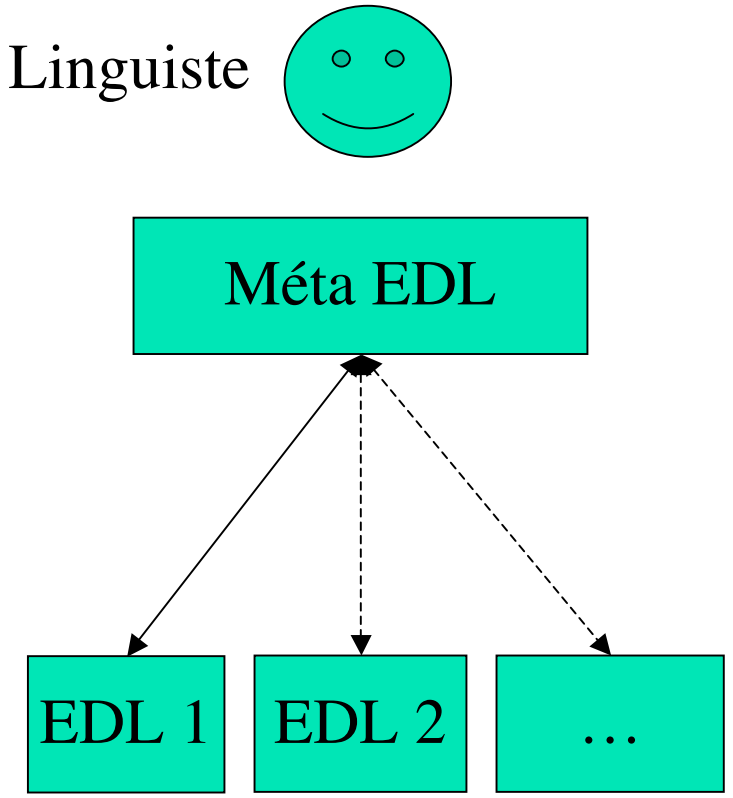
Exemple de console



méta-EDL, et méta-EDL générique

- Méta-EDL: système permettant de « piloter » un ou N EDL à distance
- N = 1: méta-EDL spécifique
  - CASH [Blanc, 1999] pour Ariane-G5
  - SYSTRAN Lite pour SYSTRAN
- N > 1: Méta-EDL générique
  - WICALE 1.0 [Carpena, 2004]

  (Web Interface for Communication with All Linguare Environnement)

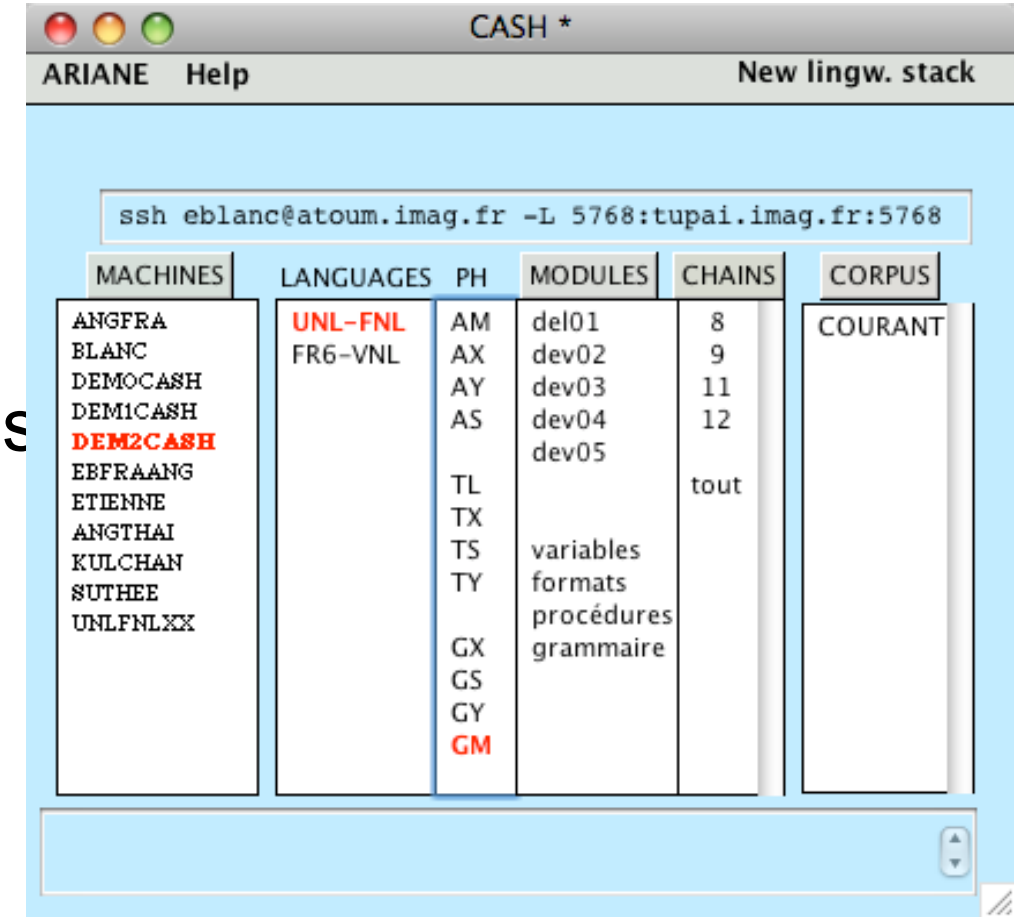


Architecture générale d'un méta-EDL générique

Exemple d'un méta-EDL

CASH pour Ariane-G5

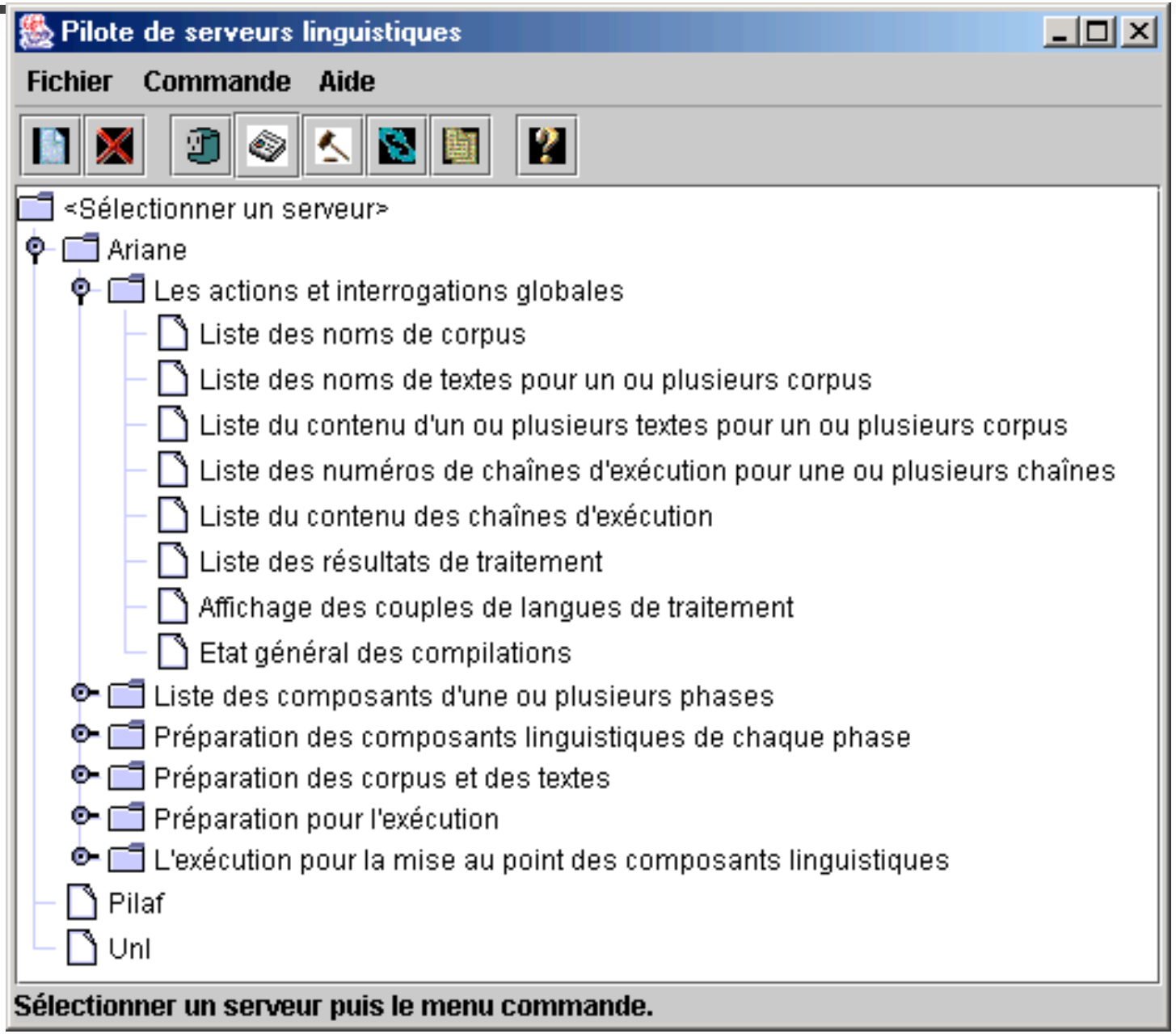
- Dédié à Ariane-G5
- Interface graphique en local
- Échange de commandes et données
- Fonctions d'aide au développeur: indexage de dictionnaire, navigation, édition de graphe…



Un méta-EDL générique
WICALE 1.0

- Ouverture sur N EDL de TA/TALN
  - Exemples avec Ariane-G5, PILAF
- Méta-langage de description
  - Serveur (EDL)
  - Commande
    - Syntaxe
    - Retour
    - Interface
- Génération d'interface de gestion de données, lancement des commandes
- Échange de commandes et données
- Extension
  - Édition des fichiers linguiciels en local : WICALE 1.0
  - Navigation : WICALE 2.0

Interface principale de WICALE 1.0



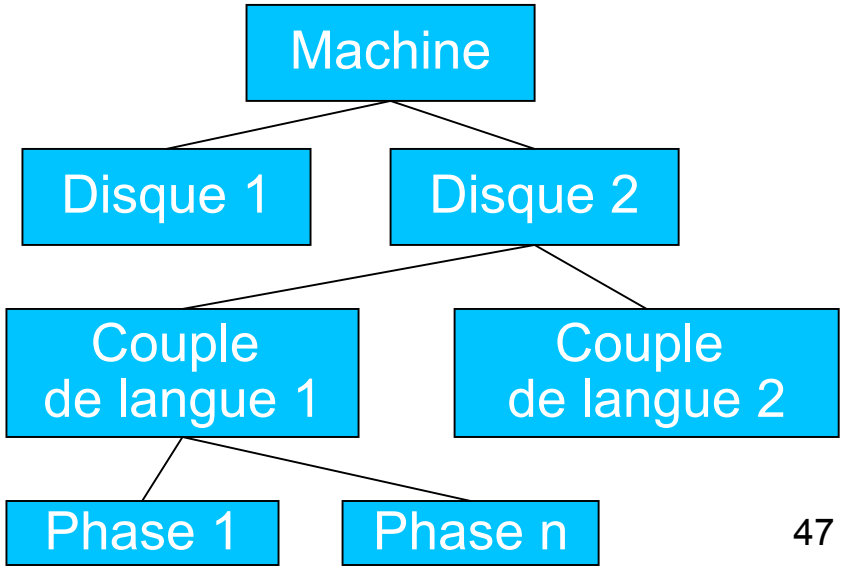
WICALE - exemple

Description de l'architecture d'Ariane-G5

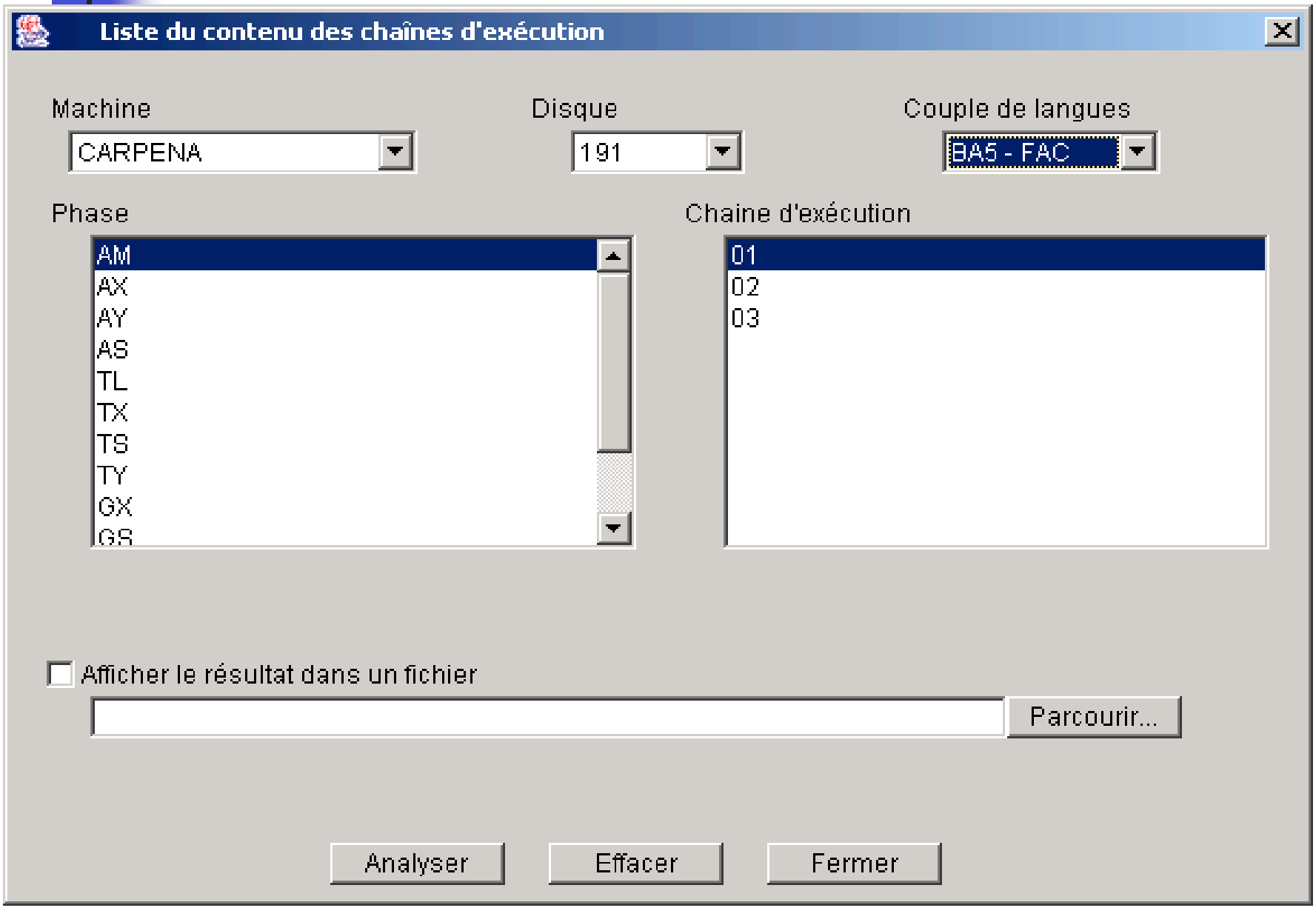
```
<ARCHITECTURE>
   <STRUCTURE id="0">
    <id>0</id>
    <niveau>1</niveau>
    <data>MACHINE</data>
    <libelle_data>LBL_MACHINE</libelle_data>
    <mise_a_jour>M</mise_a_jour><!--manuelle-->
    <param_id></param_id>
    <suiv>1</suiv>
    <nom_fichier>Liste.xml</nom_fichier>
    <valeur_tous></valeur_tous>
    <valeur_def>CARPENA</valeur_def>
    <expression>.*</expression>
  </STRUCTURE>
```

```
<STRUCTURE id="1">
    <id>1</id>
    <niveau>2</niveau>
    <data>DISQUE</data>
    <libelle_data>LBL_DISQUE</libelle_data>
    <mise_a_jour>M</mise_a_jour><!--manuelle-->
    <param_id>0</param_id>
    <suiv>2;3;6</suiv>
    <nom_fichier>Liste.xml</nom_fichier>
    <valeur_tous></valeur_tous>
    <valeur_def>191</valeur_def>
    <expression>.*</expression>
</STRUCTURE>
```

```
<STRUCTURE id="2">
    <id>2</id>
    <niveau>3</niveau>
    <data>LANGUE</data>
    <libelle_data>LBL_LANGUE</libelle_data>
    <mise_a_jour>A</mise_a_jour><!--
    automatique commande LIENLANG-->
    <param_id>0;1</param_id>
    <suiv>3;4</suiv>
    <nom_fichier>Liste.xml</nom_fichier>
    <valeur_tous>*</valeur_tous>
    <valeur_def></valeur_def>
    <expression>.*</expression>
  </STRUCTURE>
```



Interface générée

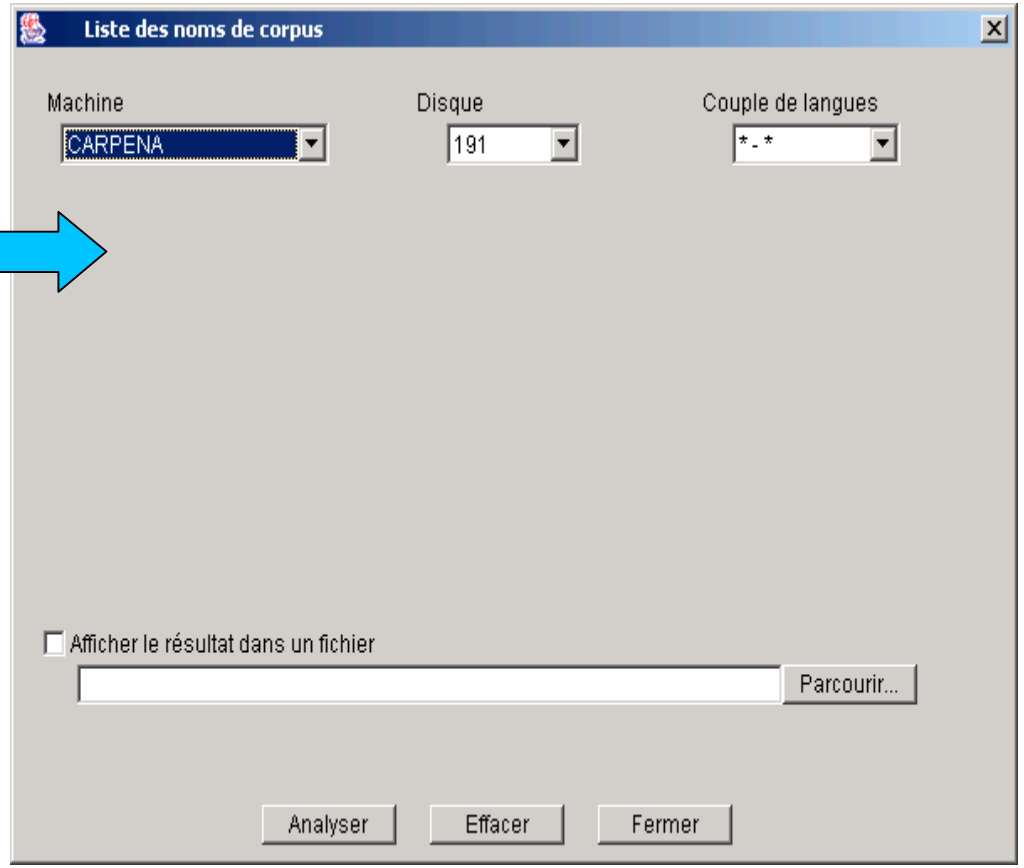


Traitement d'une commande (1)

LISNOMCORP: Liste des noms de corpus

LISNOMCORP (<Langue source | *, Liste des langues sources LIENLANG> , <Langue cible | *, Liste des langues cibles LIENLANG>, <nom corpus | *>).

```
<COMMANDE num_cde="1">
   <num_cde>1</num_cde>
   <nom_cde>LISNOMCORP</nom_cde>
   <intitule_cde>Liste des noms de corpus
   </intitule_cde>
   <PARAMETRE_SAISIE>
   <PARAMETRE>
    <nom_param>Langues</nom_param>
    <libelle_param>LBL_LANG</libelle_param>
          <pos_lib_X>10</pos_lib_X>
          <pos_lib_Y>90</pos_lib_Y>
          <dim_lib_X>120</dim_lib_X>
          <dim_lib_Y>20</dim_lib_Y>
          <type_param>Popup</type_param>
          <valeur_def_param>*</valeur_def_param>
          <pos_X>30</pos_X><pos_Y>110</pos_Y>
          <dim_X>10</dim_X><dim_Y>80</dim_Y>
       <VALEUR_LISTE>
        <libelle_liste>
           LISTE_LANG
         </libelle_liste>
            <multiligne>false</multiligne>
         <sep_multiligne></sep_multiligne>
          <valeur_liste>LIENLANG</valeur_liste>
          <pos_X>20</pos_X><pos_Y>60</pos_Y>
          <dim_X>160</dim_X><dim_Y>20</dim_Y>
       <initialise_liste>*-*</initialise_liste>
         </VALEUR_LISTE>
   </PARAMETRE>
   </PARAMETRE_SAISIE>
```




```
MACHINE = CARPENA
DISQUE  = 191
LGS     = *
LGC     = *
TRAIT = LISNOMCORP(*,*)
```

Traitement d'une commande (2)

```
MACHINE  = CARPENA
DISQUE   = 191
LGS      = *
LGC      = *
TRAIT = LISNOMCORP(*,*)
```

```
/*-------------------------------------------------------*/
** Opération de phase: M C ou G, opération composante: M C G ou S.
    MACHINE = CARPENA
    DISQUE  = 191
    LGS     = *
    LGC     = *
**
    TRAIT = LISNOMCORP ( * , * )
**
/*-------------------------------------------------------*/
-> Tout est O.K.
/*-------------------------------------------------------*/
/*-------------------------------------------------------*/
Trace de console ARIANE-G5 (traitements).
                  +-----------------------------------+
                  !                                   !
                  !           ARIANE-G5.3             !
                  !           -----------             !
                  !                                   !
                  !   ....  DEBUT DE SESSION  ....    !
                  !                                   !
                  +-----------------------------------+
ARIANEG5: type de traitement?
```

Liste des noms de corpus
Nom de corpus avec date de cr{ation.
****

ACCOMOD 04/05/01
ARIANY 30/08/05
A709 18/09/01
A710 18/09/01
CMU 15/05/00
CORPUS 06/10/99
CP 26/11/02
CSH 14/05/01
CSTAR 26/03/97
CSTIF 03/04/98

```
ARIANEG5: type de traitement?
                  +-----------------------------------+
                  !                                   !
                  !           ARIANE-G5.3             !
                  !           -----------             !
                  !                                   !
                  !    ....  FIN DE SESSION  ....     !
                  !                                   !
                  +-----------------------------------+
Voulez-vous votre mouchard de session: OUI/NON/FERMER/CONT/?/DET ?
/*-------------------------------------------------------*/
```

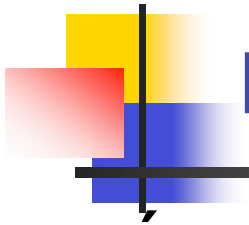
```
<SERVEUR>
     <nom_serveur>ARIANE-G5</nom_serveur>
     <classe>ServerAriane</classe>
     <Communication>Socket</Communication>
     <adresse>tupai.imag.fr</adresse>
     <port>5768</port>
     <codage>iso-8859-1</codage>
     <entete_ligne> ARIANET - LIDIA20
      </entete_ligne>
     <fin_ligne> ARIANET - LIDIA20
      </fin_ligne>
</SERVEUR>
```

```
<RESULTAT>
      <resultat_OK>-> Tout est O.K.</resultat_OK>
      <resultat_type>
         -> Tout est O.K.
        </resultat_type>
      <ETAPE>
<nom_methode>find</nom_methode>
      <expr_deb>
           Liste des noms de corpus
        </expr_deb>
      <expr_corps>
        [A-Za-z0-
9.,?!()&caract_sub;&caract_accent;:=_$:+]*
        </expr_corps>
      <expr_fin>-LISTE TERMINEE-</expr_fin>
      <expr_concat>&retour_chariot;</expr_concat>
           <expr_remplacement></expr_remplacement>
      </ETAPE>
   </RESULTAT>
```

Fichier local

Serveur Ariane-G5

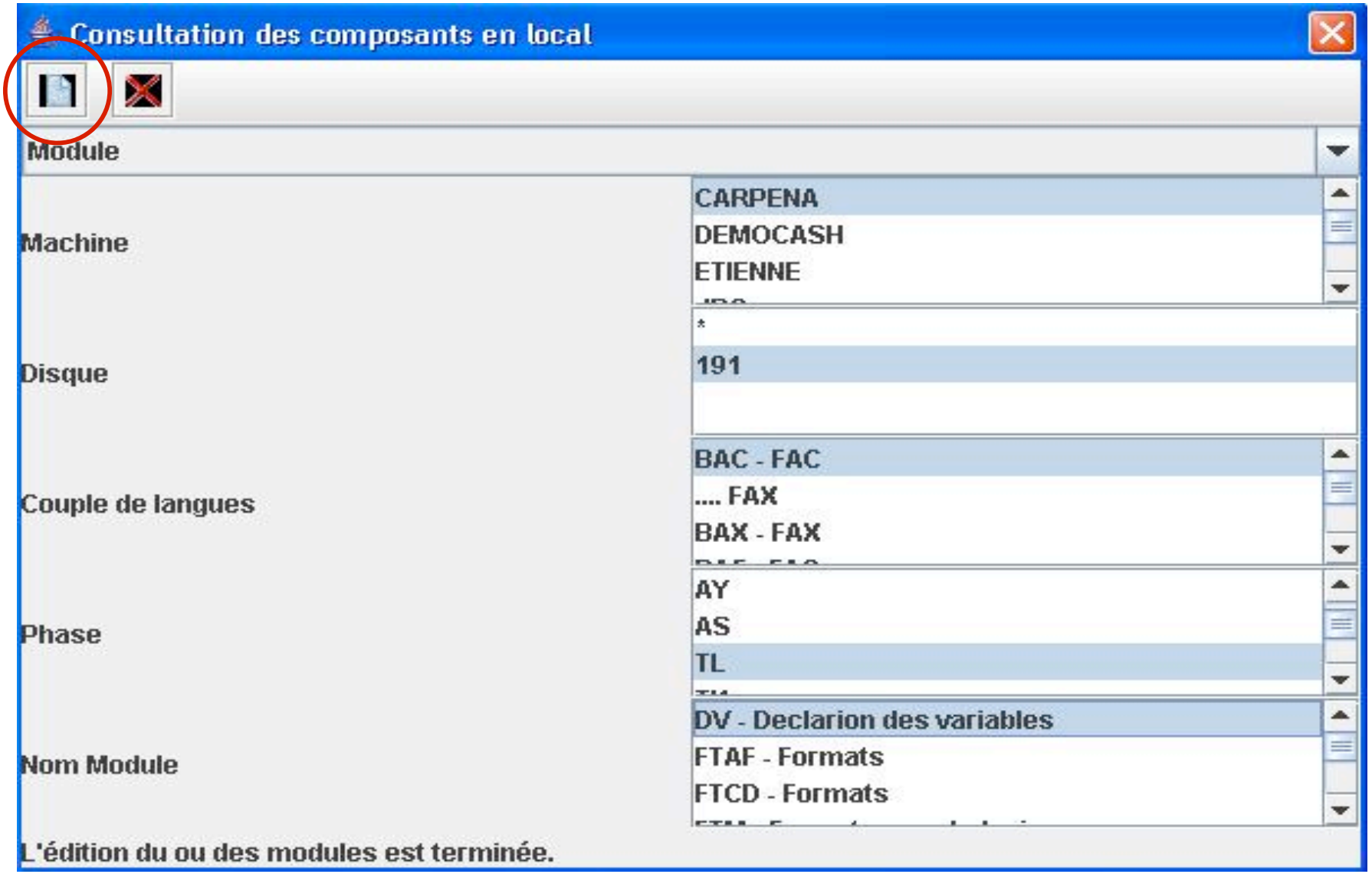
PC



Extension à WICALE 1.1 avec l'édition

- Édition par appel d'un éditeur de texte quelconque
  - Inspiration d'Ariane-G5
- Édition simultanée de plusieurs fichiers
- Protection contre les erreurs de manipulation
  - Deux modes d'ouverture : V (voir) ou M (modifier)
  - Édition sur une copie du source (sécurité totale)
  - Gestion de cohérence à l'aide des dates de dernière modification

Démonstrations de la fonction d'édition de WICALE 1.1

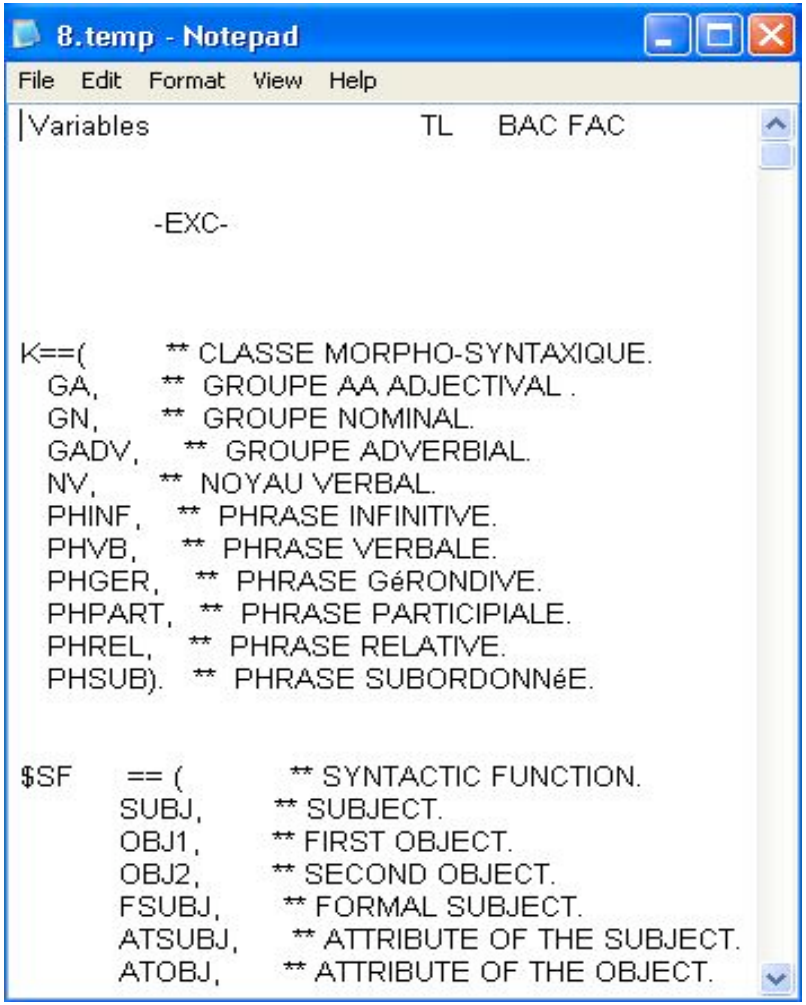


Fonction d'édition de WICALE 1.1

exemple


```
<?xml version="1.0" encoding="UTF-8" ?>
<!-- LIST FILE OPEN -->
- <LST_EDIT>
 + <EDIT>
 + <EDIT>
 + <EDIT>
 + <EDIT>
 + <EDIT>
 + <EDIT>
 + <EDIT>
 + <EDIT>
 - <EDIT>
    <nom>8</nom>
    <!-- Nom fichier temporaire 8.temp -->
    <fichier>../DATA/ARIANE-G5/CARPENA191BAC - FACTL.xml</fichier>
    <!-- Chemin vers copie locale -->
    <path>//MODULE[@nom="DVS"]/contenu/</path>
    <!-- Xpath vers source dans fichier -->
    <taille>105</taille>
    <!-- Taille du fichier -->
    <moment>3920980989</moment>
    <!-- Moment de modification tranforme par Java -->
    <id>1</id>
    <!-- Identification de copie du source -->
  </EDIT>
```

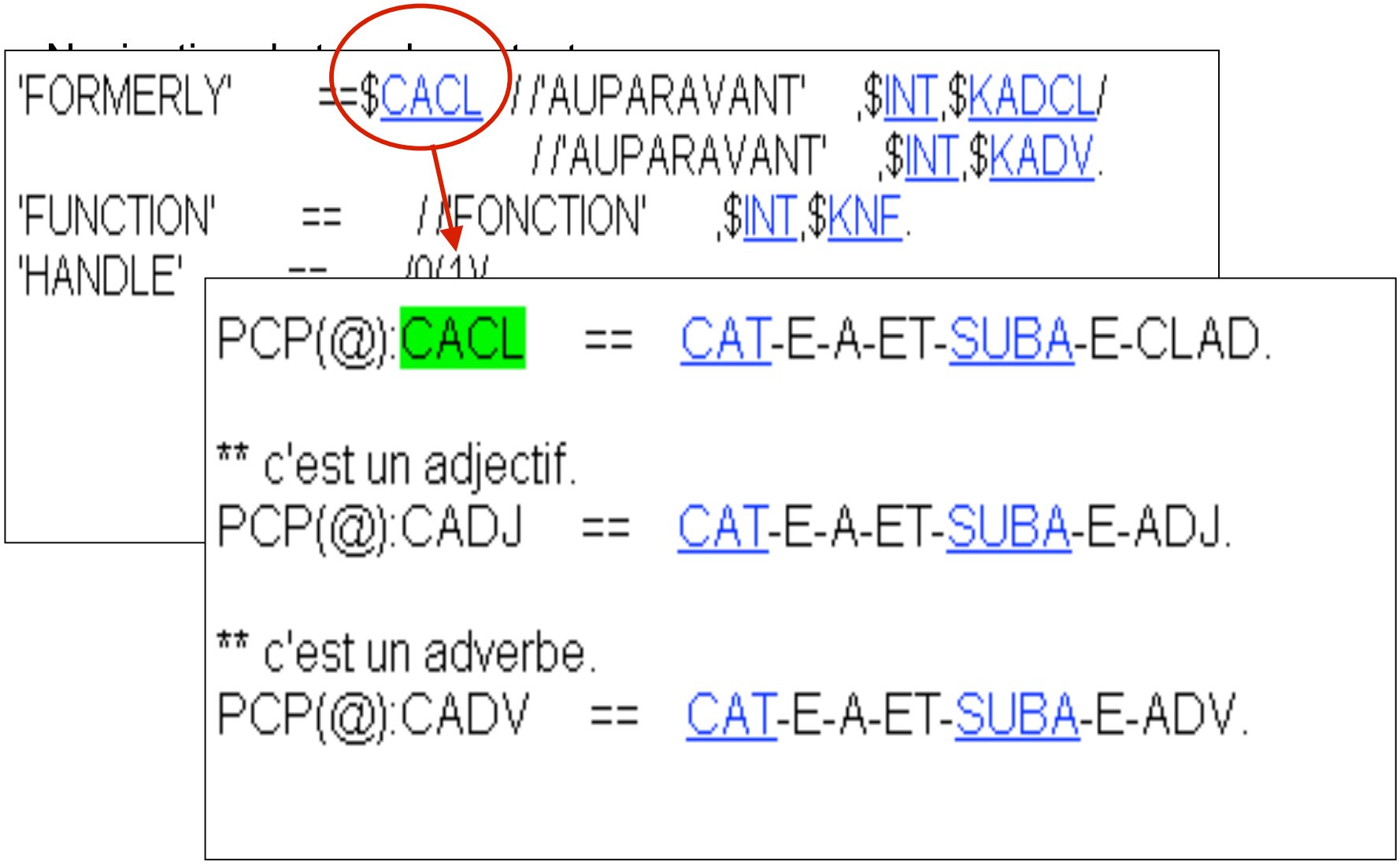
Liste des fichiers ouverts



Édition par Notepad

WICALE 2.0

navigation dans un linguiciel

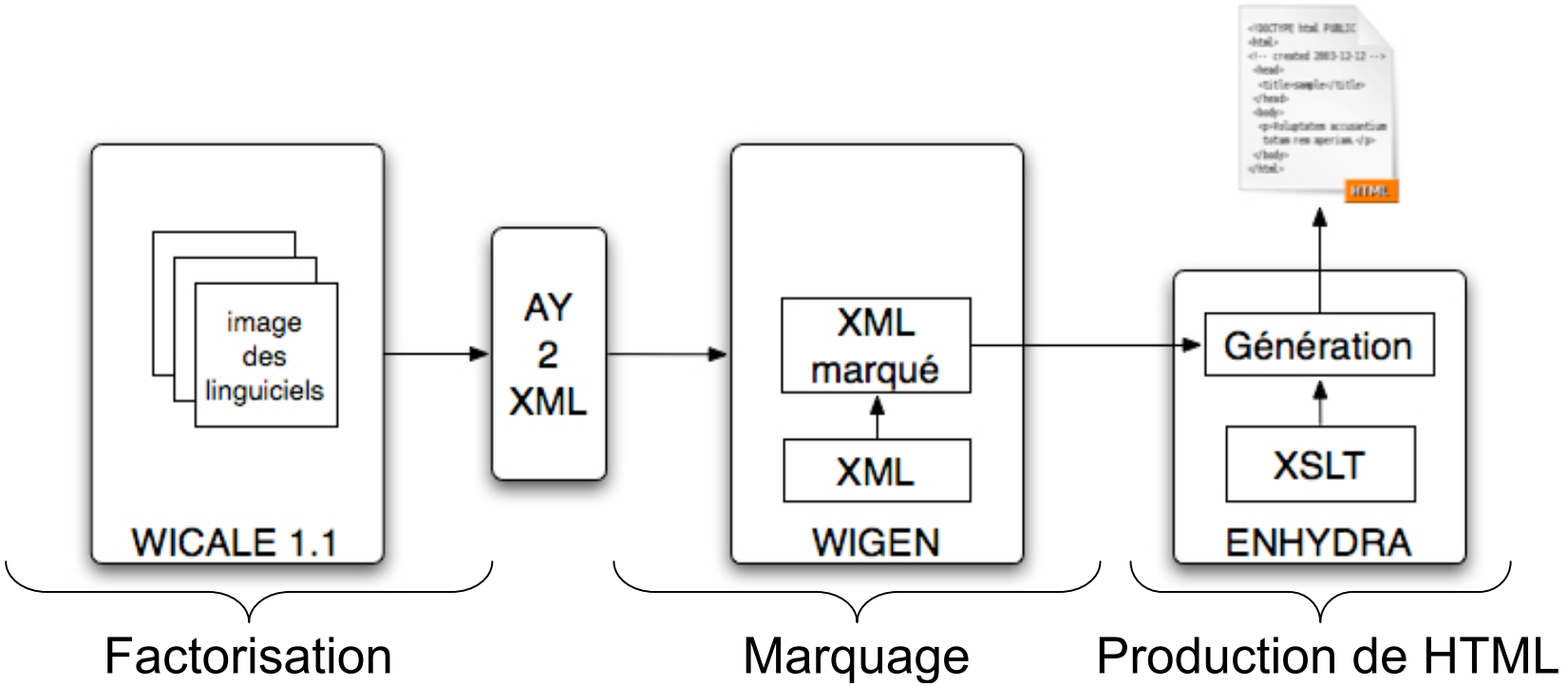


→ Génération du source linguiciel au format HTML

Navigation: solution retenue

Génération statique « à la Doxygen » en local:

- **Factorisation**:
  - Extraction des éléments dans le source
  - Transformation du source en un format intermédiaire adéquat (comme XML)
- **Marquage**: Création des liens entre les occurrences d'un élément et sa définition
- **Production**: Fabrication de la sortie HTML



Apports du travail sur WICALE

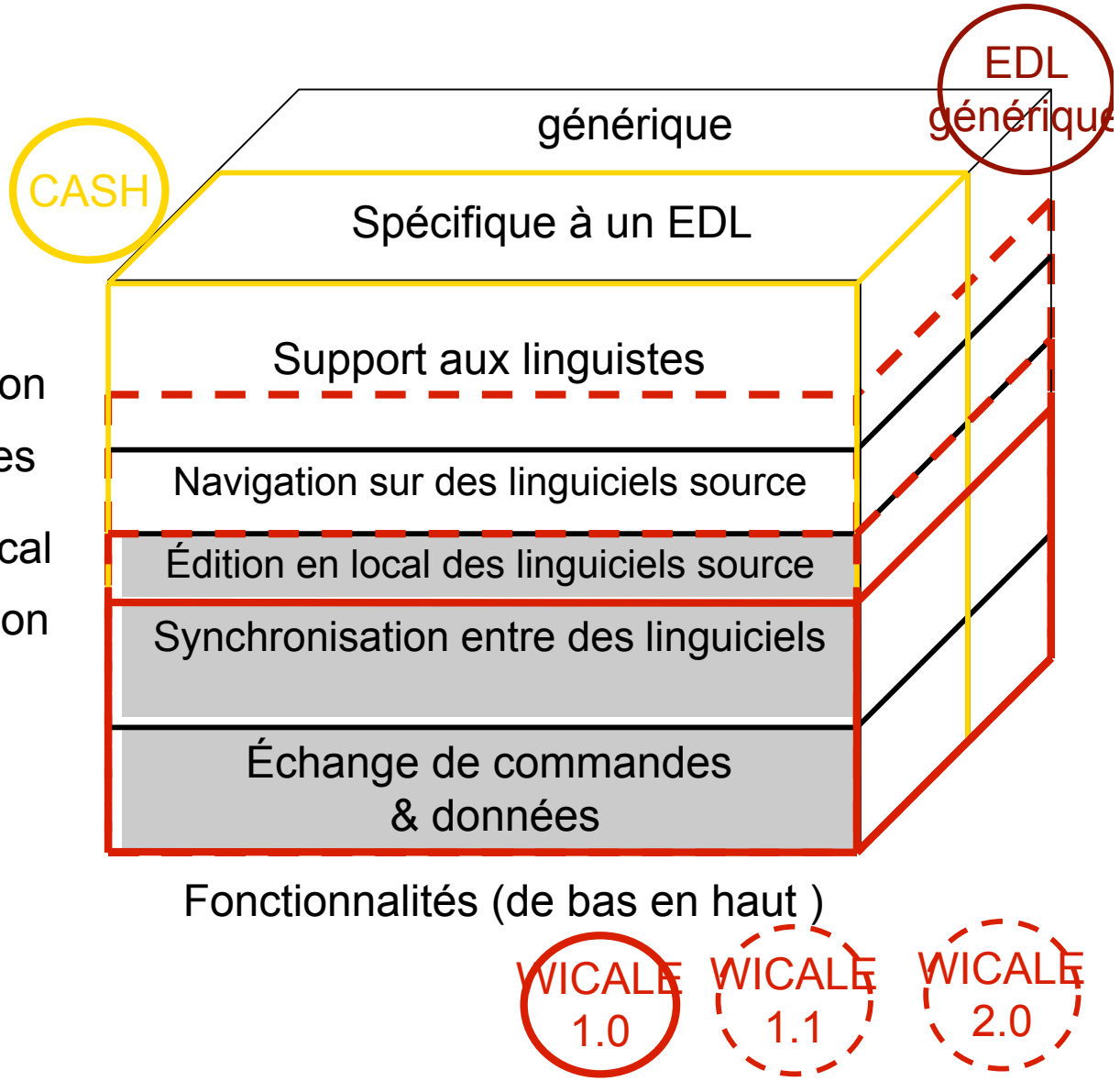
Implémenté en Java, XML.
WICALE 1.0 par [Carpena], version 1.1 (15h), version 2.0 (35h)

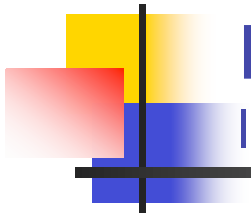
•Déport des EDL

•Généricité de la programmation

•EDL générique: intégration des aides aux linguistes, des traitements sophistiqués en local

→ Intégration ou implémentation des LSPL





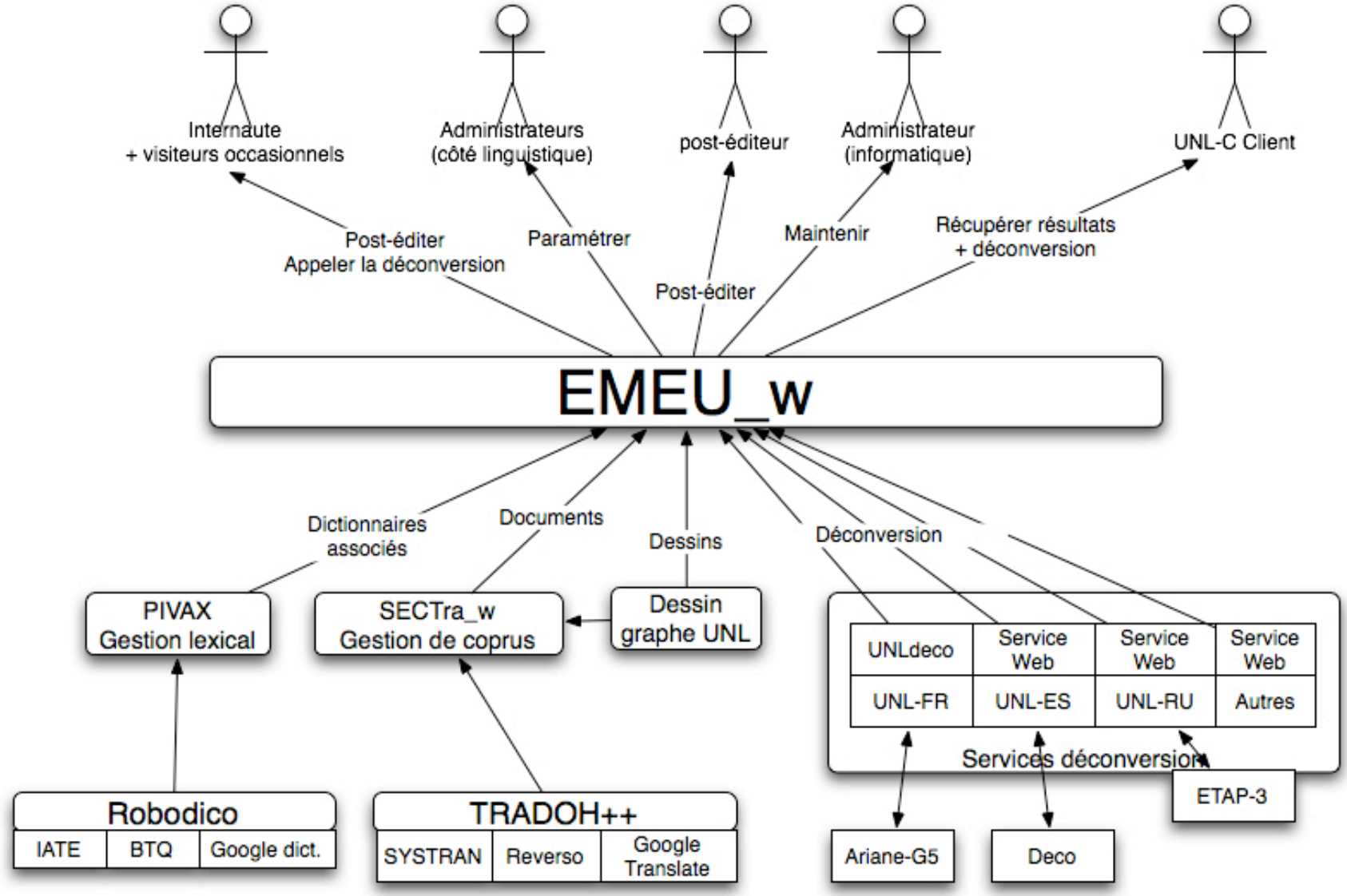
Réingénierie & extension
le moniteur EMEU_w pour le projet EOLSS

- Composants utilisés dans le projet EOLSS
  - Documents source: site EOLSS
  - Documents traduits: SECTra_w
  - Documents prétraduits: TRADOH++
  - Dictionnaire associé: PIVAX
  - Service de déconvertisseur: UNLdeco
- Types d'utilisateurs
  - Visiteur, administrateur, post-éditeur, Client UNL
- Contexte Web

→ Moniteur Web EMEU_w (Moniteur interactif Web pour des systèmes de TA et TAO fond" sur UNL)

Moniteur Web EMEU_w

Architecture



Technique inspirée de WICALE

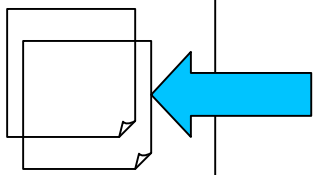
Contexte Web

```
<COMMANDE num_cde="4">
<num_cde>4</num_cde>
<nom_cde>DOCUMENT_REV_LIST</nom_cde>
<url>xwiki/bin/view/Corpus/PostEdit</url>
<method>get</method>
<delay>1000</delay>
<num_cde_previous_list/>
<PARAMETRE_SAISIE>
<PARAMETRE>
<nom_param>projName</nom_param>
<libelle_param>Project</libelle_param>
<dim_lib_parent>id_parent_a_attacher_dans_interface</dim_lib_
parent>
<dim_lib_lenght>20</dim_lib_lenght>
<type_param>Hidden</type_param>
<valeur_def_param>EOLSS</valeur_def_param>
<local_valeur_list>/PROJECT_LIST/PROJECT</local_valeur_list>
<dim_parent>id_parent_a_attacher_dans_interface</dim_parent>
<dim_lenght>20</dim_lenght>
</PARAMETRE>
</PARAMETRE_SAISIE>
```

Description de commande

Requête HTTP

Page HTML



```
<RESULTAT>
 <item-group name="rev-projet">
      <item name="no" type="xpath" out="text"
            expression="/html[1]/body[1]/div[1]/tr/td[1]/div[1]"/>
      <item name="documentName" type="xpath" out="text"
            expression="/html[1]/body[1]/div[1]/table[1] tr/td[2]"/>
      <item name="documentLanguage" type="xpath" out="object"
            expression="/html[1]/body[1]/div[1]/table[1] tr/td[3]"/>
 </item-group>
</RESULTAT>
```

Xpath ou expression régulière

EMEU_w: interface

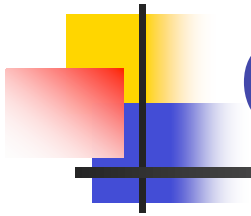
Home | Import | Evaluation | Post-edtion | Multimodal | TM | Admin | Translation | Contact us

Corpus: EOLSS Document: – Select – Source-target: – Select –

No	Document name	Language pairs	post-edition percent
1	D1_E1_37_05_14_TXT	english_french english_spanish english_russian	311/311 = 100.0 % 0/311 = 0.0 % 16/311 = 5.145 %
2	D2_E2_03_05_TXT	english_french english_german	502/502 = 100.0 % 15/502 = 2.988 %
3	D3_E2_24D_04_05_TXT	english_french english_russian	512/512 = 100.0 % 0/512 = 0.0 %
4	D4_E2_24M_02_04_TXT	english_french	582/591 = 98.477 %

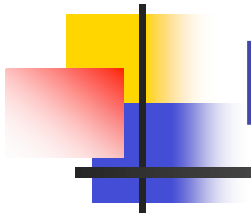
Home | UNLDL (Client) | GETALP-LIG | Post-editor | Internaute (visito | Administration (linguistic) | Administration (informatic) | Full screen

no	documentName	documentStatus	documentLanguage	source (English)	unl (UNL format)	French (HTML format)	UNL-FR Deconverter result	EOLSS translation result	post-edit (via SECTra_w)	see revised Document (HTML)
1	D1_E1_37_05_14_TXT	311/311 = 100.0 %0/311 = 0.0 %16/311 = 5.145 %	english_french english_spanish english_russian	download	download	download	download	download	do post-Edit	See
2	D2_E2_03_05_TXT	502/502 = 100.0 %15/502 = 2.988 %	english_french english_german	download	download	download	download	download	do post-Edit	See
3	D3_E2_24D_04_05_TXT	512/512 = 100.0 %0/512 = 0.0 %	english_french english_russian	download	download	download	download	download	do post-Edit	See



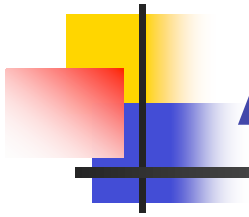
Conclusion pour l'EDL

- EDL, méta-EDL et méta-EDL générique
- Généricité: travailler à la fois avec plusieurs EDL à distance
- Flexibilité: utiliser la technique d'implémentation avec le méta-langage
- Réalisation dans la thèse
  - Extension de WICALE: édition et navigation
  - Application au moniteur Web EMEU_w pour un système de TAO hétérogène



Plan

- Introduction: TA, THAM et TAO hétérogène
- BDLex pour la TAO hétérogène: PIVAX
  - Motivations
  - Architecture linguistique
  - Contrôle sur le partage des données
  - Expérimentation et validation pour le projet U++C et EOLSS
- Méta-EDL & EDL générique: WICALE & EMEU_w
  - Motivations
  - Exemple du méta-langage
  - Application pour le moniteur EMEU_w dans le projet EOLSS
  - Vers un EDL universel
- Réingénierie de LSPL: les systèmes-Q
  - Motivations
  - Réalisation
  - Application pour le projet OMNIA
  - Extensions
- Conclusion & perspectives: généricité, simplicité et flexibilité

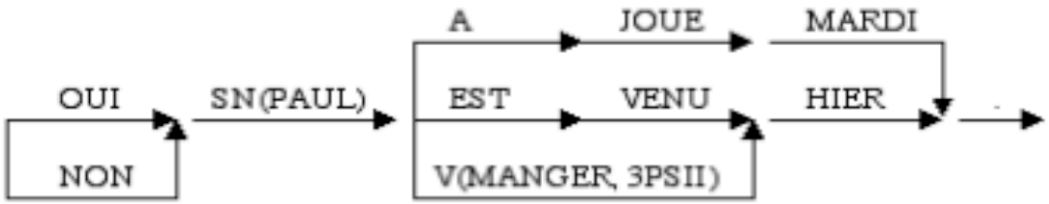


Approche langage

- Motivations
  - Puissance dans l'implémentation (WICALE, EMEU_w)
  - Efficace dans le développement (LSPL)
    - Exemples: Ariane-G5 (5 LSPL), MU (GRADE), MÉTÉO (les systèmes-Q)
- De la TA à la TAO hétérogène générique
  - Importer des traitements des systèmes distants vers un système générique
  - Intégration ou réingénierie des LSPL
- Principes de la réalisation d'un LSPL
  - Réingénierie des systèmes-Q [A.Colmerauer, 1971]
  - Application pour des applications de TALN

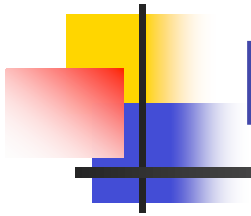
Les systèmes-Q

```
OUI + SN(PAUL) + EST + VENU + HIER + .
OUI + SN(PAUL) + A + JOUE + MARDI + .
OUI + SN(PAUL) + V(MANGER, 3PSII) + HIER + .
NON + SN(PAUL) + EST + VENU + HIER + .
NON + SN(PAUL) + A + JOUE + MARDI + .
NON + SN(PAUL) + V(MANGER, 3PSII) + HIER + .
```




```
OUI == YES.
NON == NO.
EST + VENU + HIER == V(COME, PAST) + YESTERDAY.
EST+VENU + HIER == V(COME,PRESENT_PERFECT,SIN) + YESTERDAY.
V(MANGER, 3PSII) = V(EAT, 3PSII).
...
```

```
V(I*,PRESENT_PERFECT,SIN) = V(HAVE, I*) / I* -EQ- (COME).
V(I*, 3PSII) == V(EAT, 3PSII) / I* -DANS- (MANGER, DINNER).
V(J*, 3PSII) == J* + S / J* -HORS- (GO, DO).
--                 == J* + ES / J* -DANS- (GO, DO).
```

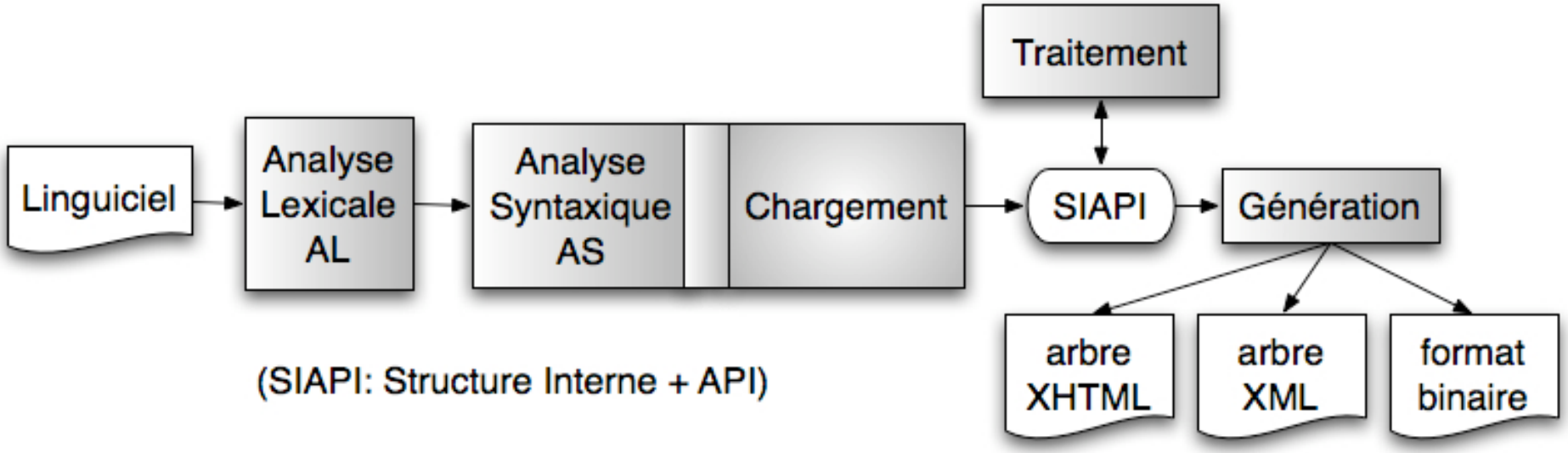


Pourquoi les systèmes-Q ?

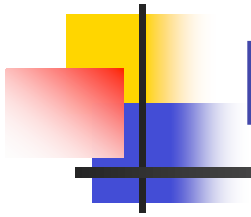
- Application
  - MÉTÉO [Chandioux et Guéraud, 1981]
  - TAUM-AVIATION (1976-1980)
- Puissance et simplicité (appropriabilité) dans des applications de TALN
- Syntaxe et sémantique opérationnelles connues
- Exemples et expérimentations

Principes de la réalisation

- Implémentation directe vs. utilisation des générateur de compilateur



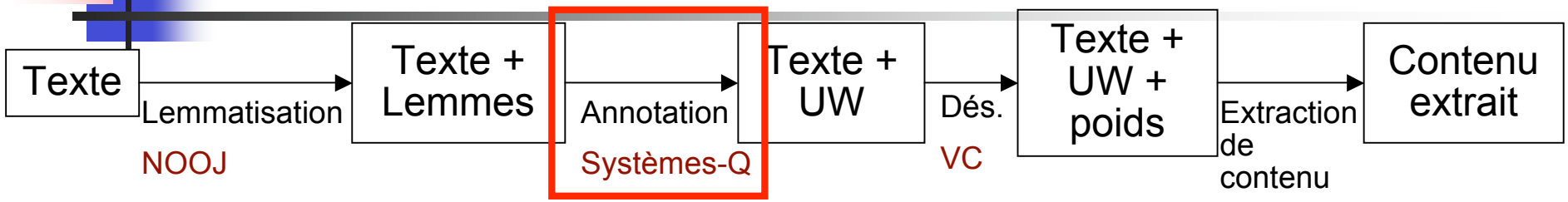
- Incrémentabilité
- SI (Structure Interne) = SA (Structure Abstraite) + EPA (Extensions pour Présentation & Algorithmes)



Réalisation

- Générateur des compilateurs ANTLR/C: [http://antlr.org]
- Définition de SI [Verdurand, Boitet, Bellynck, 2005]
- Algorithme reconstitué [Boitet et Nguyen, 2007]
- Validation
  - Exemples pédagogiques [Colmerauer 1968, Boitet 1978, Guilbaud 1980]
  - Annotation de textes dans le projet OMNIA [Rouquet et Nguyen 2009]
  - Moniteur: http://sway.imag.fr/unldeco/SystemsQ.po
- Extensions
  - Support Unicode
    - étiquette = caractère (caractère_non_special)*
  - Règles en mode « dictionnaire »
    - A =d= B, B ne sera pas réutilisable pour appliquer une règle

Application dans le projet OMNIA (1)



Iraqi women sitting in a waiting room of Bagdad hospital

lemmatisation

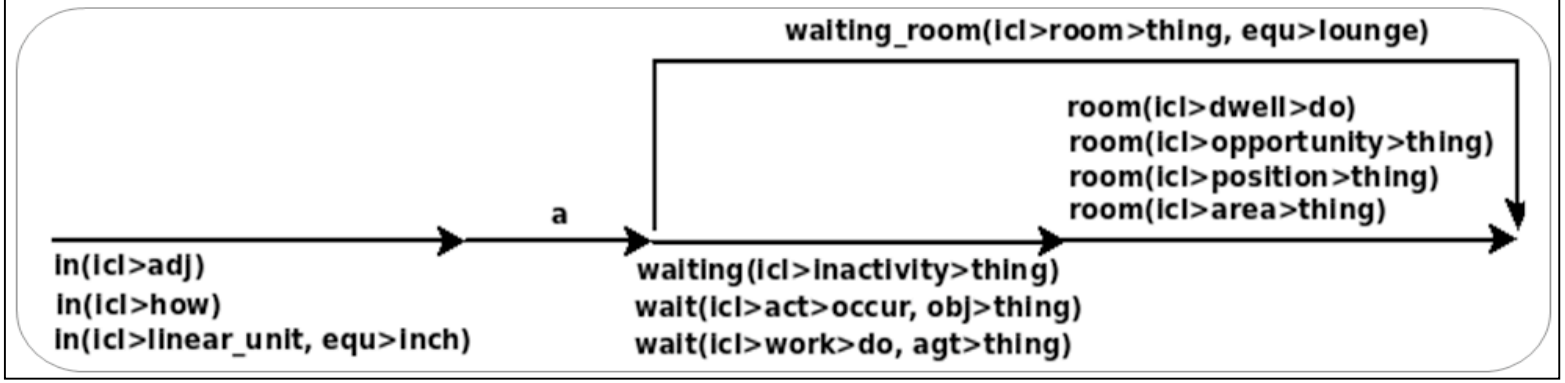
Q-System (lemmas)

```
-01- $OCC($LU(LEMME(in), CAT(N)), $LU(LEMME(in), CAT(A))) -02-
-02- $OCC($LU(LEMME(a), CAT(UNK))) -03-
-03- $OCC($LU(LEMME(waiting room), CAT(N))) -04-
-03- $OCC($LU(LEMME(waiting), CAT(N)), $LU(LEMME(wait), CAT(V)))
+ $OCC($LU(LEMME(room), CAT(N)),$LU(LEMME(room), CAT(V))) -04-
```

Q-rule (dictionary)

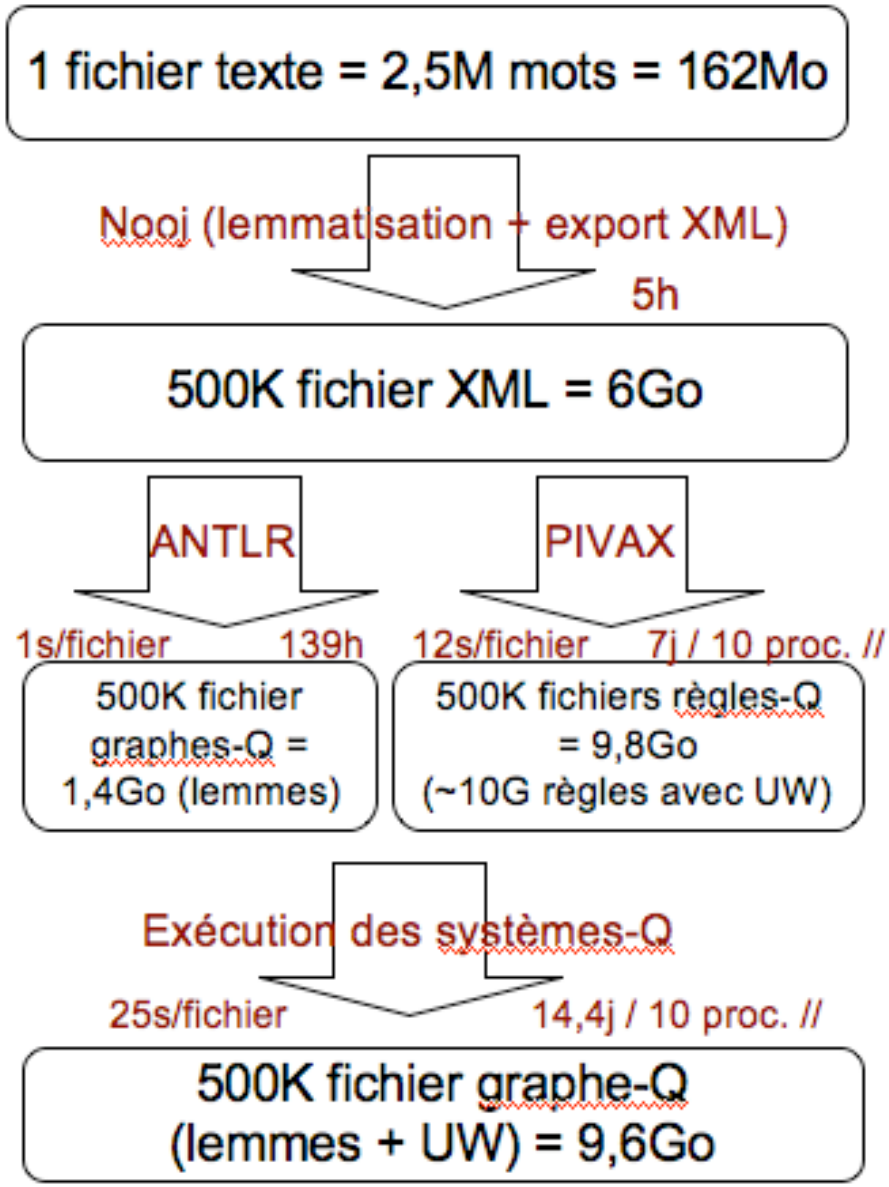
```
$LU(LEMME(in), CAT(N), U*)) ==
$LU(UWS(in(icl_!GT_linear_unit,equ_!GT_inch)), LEMME(in), CAT(N)).
```

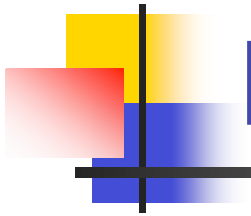
Q-System (UWs)



Application dans le projet OMNIA (2)

- Normalisation & annotation par les systèmes-Q
  - Ouverture et extensibilité dans le processus de traitement
- Mini-dictionnaire pour chaque texte compagnon généré par PIVAX
- Mise en service selon le principe de WICALE (en cours)



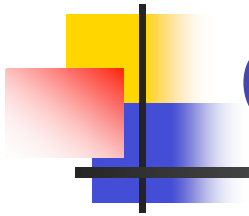


Plan

- Introduction: TA, THAM et TAO hétérogène
- BDLex pour la TAO hétérogène: PIVAX
  - Motivations
  - Architecture linguistique
  - Contrôle sur le partage des données
  - Expérimentation et validation pour le projet U++C et EOLSS
- Méta-EDL & EDL générique: WICALE & EMEU_w
  - Motivations
  - Exemple du méta-langage
  - Application pour le moniteur EMEU_w dans le projet EOLSS
  - Vers un EDL universel
- Réingénierie de LSPL: les systèmes-Q
  - Motivations
  - Réalisation
  - Application pour le projet OMNIA
  - Extensions
- Conclusion & perspectives: généricité, simplicité et flexibilité

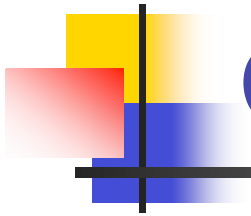
Conclusion & perspectives: base lexicale

- Base de données lexicales PIVAX
  - Généricité
    - entre des systèmes de TA
    - usage multiple entre la TA et la THAM
  - Simplicité
    - Support le développement lexicale des systèmes de TA à « pivot lexical »
  - Flexibilité
    - Partage partiel des données
    - Interface à la colonne comme PARAX sur le Web
    - Serveur lexical
- Perspective
  - Court terme
    - Intégrabilité aux systèmes de TA (Ariane-Y)
    - Programmabilité
  - Long terme
    - Extensibilité pour un ensemble arbitraire des systèmes



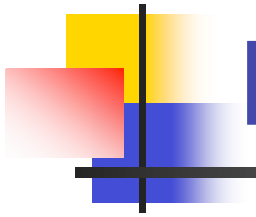
Conclusion & perspectives: EDL

- Méta-EDL, méta-EDL générique WICALE
  - Généricité
    - Travailler aux plusieurs EDL (Environnement de Développement Linguiciel)
  - Simplicité
    - Délégation aux éditeurs dans l'édition
    - Navigation à la Doxygen
  - Flexibilité
    - Méta-langage pour la description d'organisation, de commandes et d'interfaces
    - Application dans le moniteur EMEU_w dans le contexte Web
- Perspective
  - Court terme
    - Expérimentation avec autres EDL
    - Intégration dans une architecture en agent à gros grains
  - Long terme
    - Construction un gros système de TAO hétérogène avec des composants différents
      - Flux de travaux
      - Flot de donnés



Conclusion & perspectives: LSPL

- Approche langage: réingénierie des systèmes-Q
  - Flexibilité
    - LSPL (Langage Spécialisé pour la Programmation Linguistique)
    - Extensibilité dans le processus de traitement dans le projet OMNIA
  - Simplicité
    - Utilisation
- Perspective
  - Court terme
    - Limitation de taille de traitement
    - Implémentation d'autre LSPL
  - Long terme
    - Appropriabilité



Publications

[Boitet et al., 2008] **Christian Boitet, Cong-Phap Huynh, Hervé Blanchon, Hong-Thai Nguyen et David Rouquet (2008)** *A Web-oriented System to Manage the Translation of an Online Encyclopedia Using Classical MT and Deconversion from UNL*. ASWC/IC-08, Bangkok, 11 p.

[Mangeot et Nguyen, 2009] **Mathieu Mangeot et Hong-Thai Nguyen (2009)** *Building lexical resources: towards programmable contributive platforms*. IEEE-RIVF 2009, International Conference on Computing and Communication Technologies, July 13-17, 2009. Danang, Vietnam, pp. 84-92.

[Nguyen, 2005] **Hong-Thai Nguyen (2005)** *Vers un "méta-EDL", puis un "EDL générique" pour la TAO*. Mémoire de M2R, UJF (Grenoble 1), 85 p.

[Nguyen et Boitet, 2007] **Hong-Thai Nguyen et Christian Boitet (2007)** *Vers un méta-EDL complet, puis un EDL universel pour la TAO*. TALN 2007, Toulouse, 5–8 juin 2007, 10 p.

[Nguyen et Boitet, 2009] **Hong-Thai Nguyen et Christian Boitet (2009)** *Lexical synergy between MT & Translator Aids: PIVAX, a generic online contributive lexical database platform.* International Conference "Machine Translation 25 Years On", Cranfield, England, November 21-22, 2009, 8 p.

[Nguyen, Boitet et Sérasset, 2007] **Hong-Thai Nguyen, Christian Boitet et Gilles Sérasset (2007)** *PIVAX, an online contributive lexical data base for heterogeneous MT systems using a lexical pivot.* SNLP-2007, December 2007, Bangkok, Thailand, 6 p.

[Rouquet et Nguyen, 2009a] **David Rouquet et Hong-Thai Nguyen (2009a)** *Interlingual annotation of texts in the OMNIA project* 4th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, November 6-8, 2009, Poznań, Poland, 5 p.

[Rouquet et Nguyen, 2009b] **David Rouquet et Hong-Thai Nguyen (2009b)** *Multilinguïsation d'une ontologie par des correspondances avec un lexique pivot*. Terminologie & Onthologie: Théories et Applications (TOTh-09), Annecy, 4-5 juin 2009, 19 p.