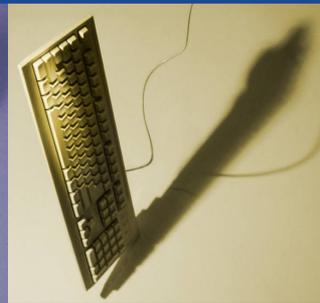




Formation technique ORI-OAI

Installation et configuration

Rennes, 26-27-28 mars 2008



Introduction

Fonctionnalités

Architecture

Installation basique

Configuration avancée



Introduction

■ Organisation

■ 3 jours

■ 26 mars 9h30 → 28 mars 16h

■ Public

■ Services informatiques

■ Pré-requis

■ Avoir un projet d'établissement construit

■ Technologies

■ Linux, Tomcat, XML, Ant

■ Spring serait un plus



- Installation et configuration de ORI-OAI version 1.0
 - Fonctionnalités
 - Architecture
 - Installation basique
 - Configuration avancée

- Quelles sont vos objectifs pour cette formation ?
- Quel est le projet de votre établissement ?
- Avez-vous déjà dans votre établissement une expérience d'archive institutionnelle ou de portail de ressources numériques ?
- Envisagez-vous d'utiliser ORI-OAI pour gérer votre projet d'archive institutionnelle ou de portail ?

- A quelle étape vous situez-vous dans votre démarche de projet avec ORI-OAI ?
- Avez-vous constitué une équipe projet dans votre établissement ?
 - Combien de personnes ? Quels profils ?
- Votre niveau de connaissance technique
 - Linux ?
 - Tomcat ?
 - XML ?
 - ANT ?

- **Besoin** de former des **personnes relais** dans les UNR
- **Engagement** des stagiaires dans notre démarche
 - Les formés organisent des **formations régionales**
 - **Identification** des futurs formateurs
- Formalisation de l'**organisation** de ces formations
 - Qui ? Quand ? Où ? À Qui ?
- Personnes relais entre les **stagiaires régionaux** et les **développeurs**
 - Tous les compléments à la formation peuvent être donnés sur la liste *ori-oai-utilisateurs*

Fonctionnalités

- **Gérer et publier** les ressources numériques de l'établissement (ressources pédagogiques, scientifiques...)
 - **Référentiel unique** de ressources connecté au SI
 - **Production** de métadonnées et **diffusion** des ressources produites par l'établissement

- **Valoriser** la production de l'établissement par une indexation de qualité
 - **Indexation avec des normes et standards**
 - Utilisation de **classifications paramétrables** articulés sur des standards comme Dewey

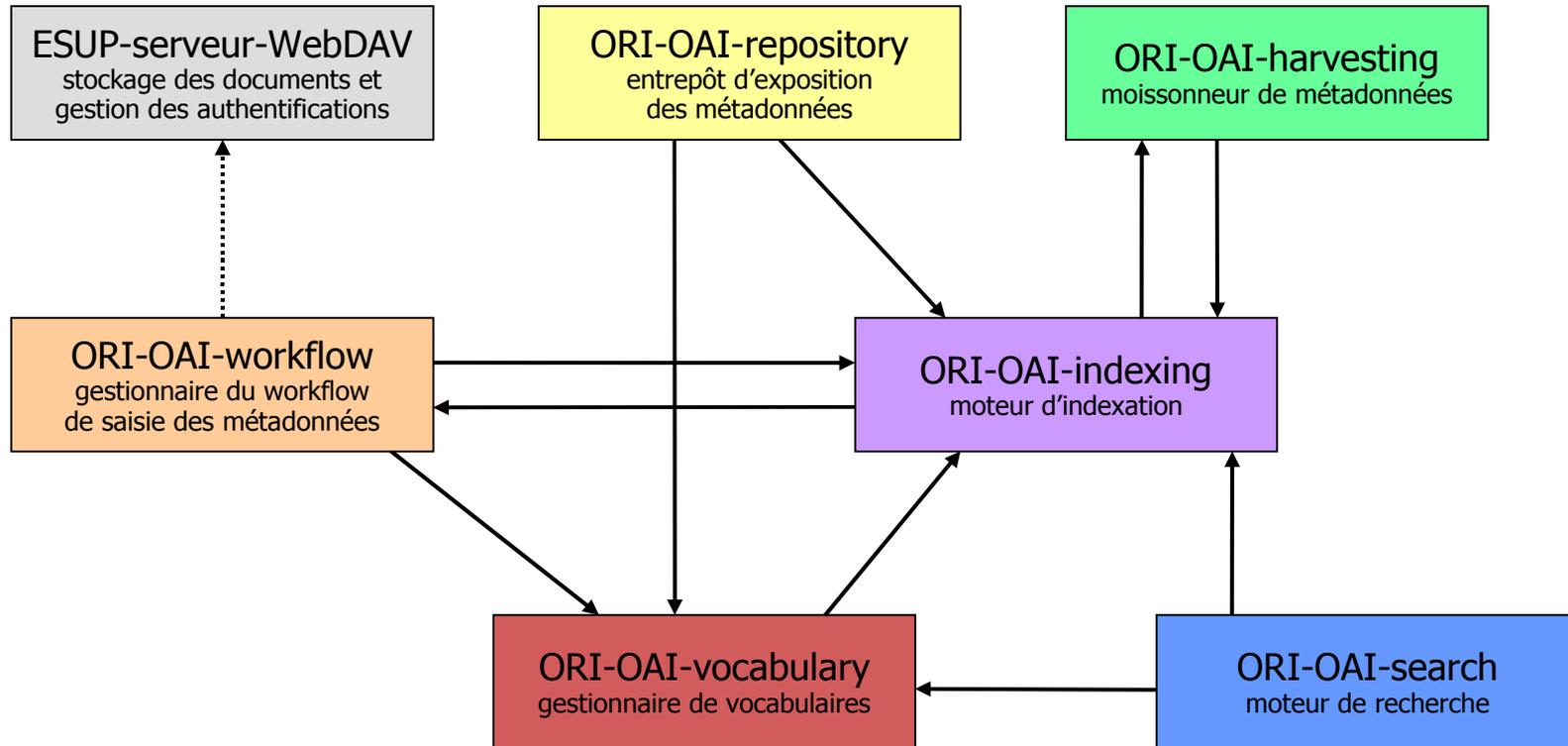
- **Partager** les ressources numériques avec d'autres établissements
 - **Exposition** des fiches de métadonnées de l'établissement
 - **Agrégation** de fiches de métadonnées

- **Accéder** aux ressources numériques à **distance** en fonction des **droits d'accès**
 - Système de **recherche avancée** et **thématique**
 - Définition des **droits**

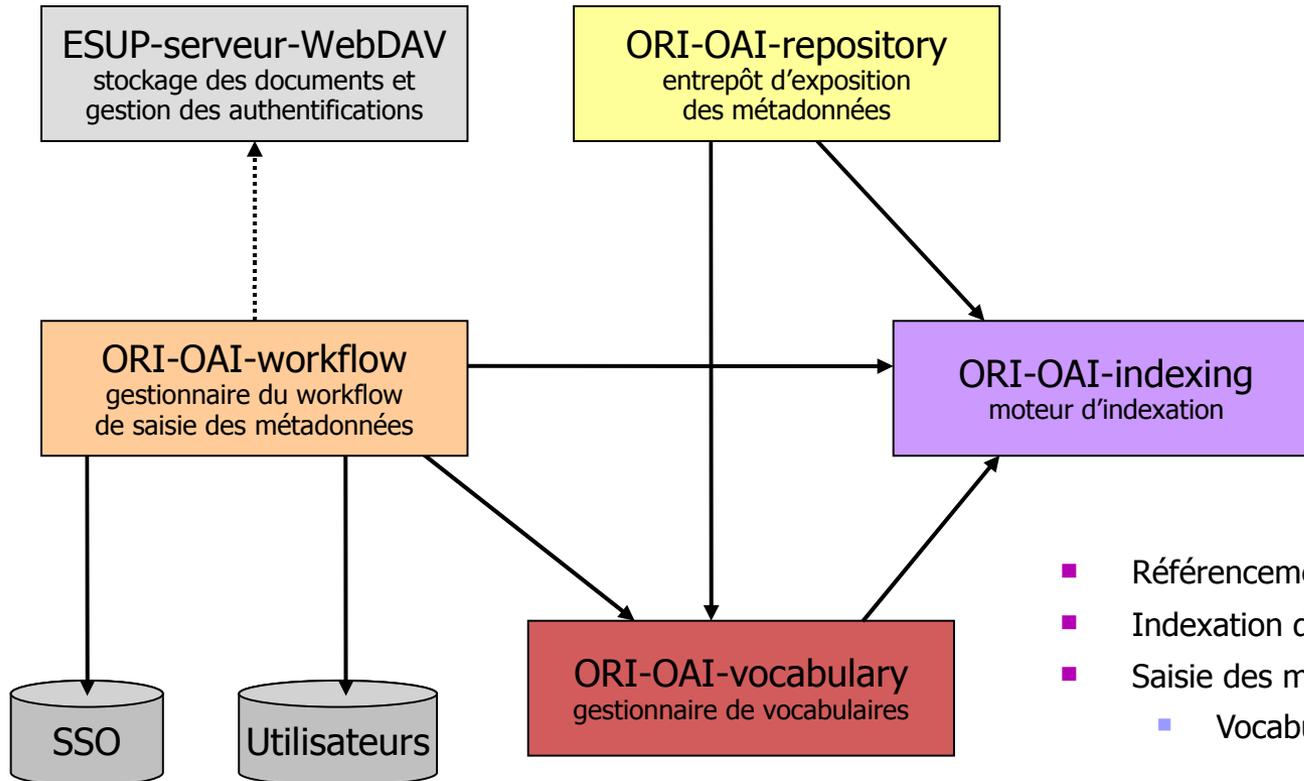
- Référentiels partagés
- Système d'identification et d'authentification de l'établissement
- Intégration dans l'ENT
- ...

recherche	sur les métadonnées, texte intégral
diffusion	en respectant le droit d'auteur, les licences et les droits d'accès
publication web	versions de publication, workflows, plusieurs schémas de métadonnées
archivage	documents natifs toujours disponibles pour les auteurs
partage	protocole OAI/PMH, fédération d'identités
référencement	indexation de qualité adaptée aux divers types de documents, classification pivot

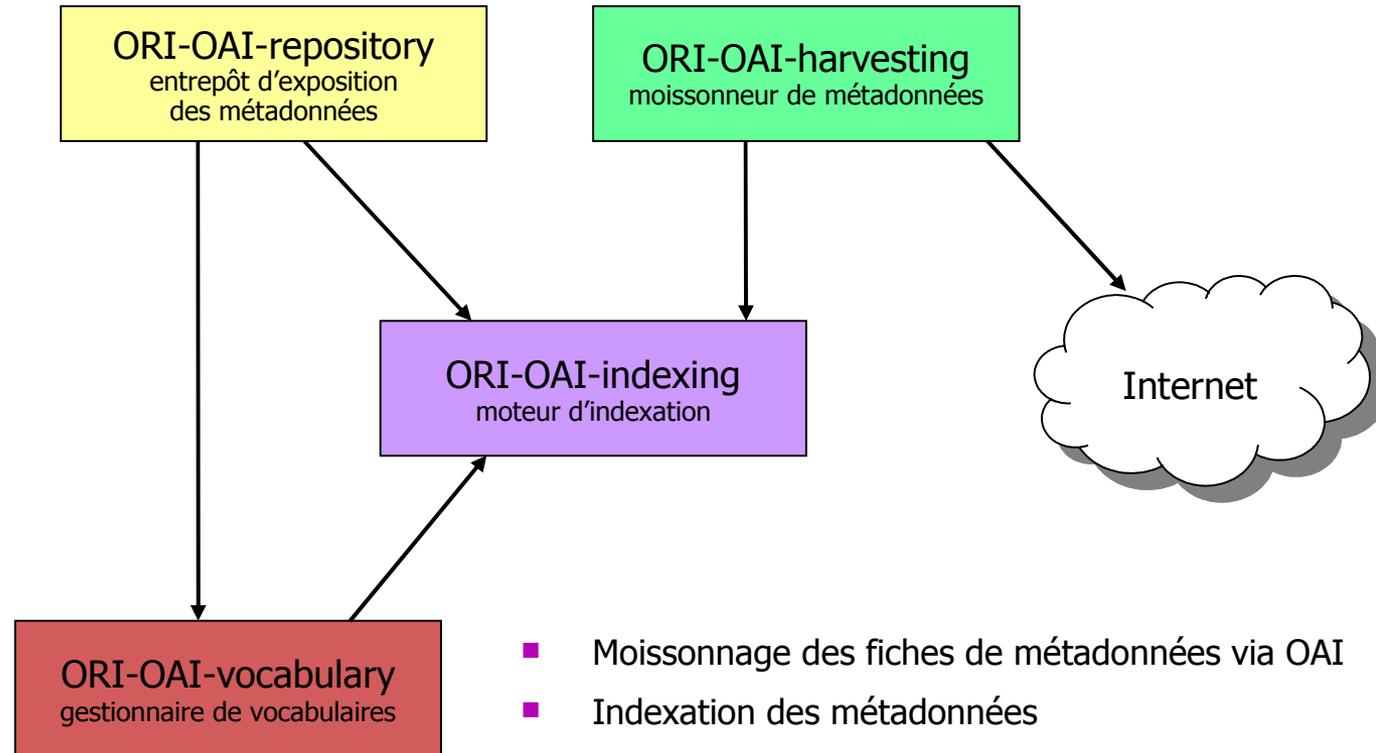
Architecture



- 7 modules
- interopérables

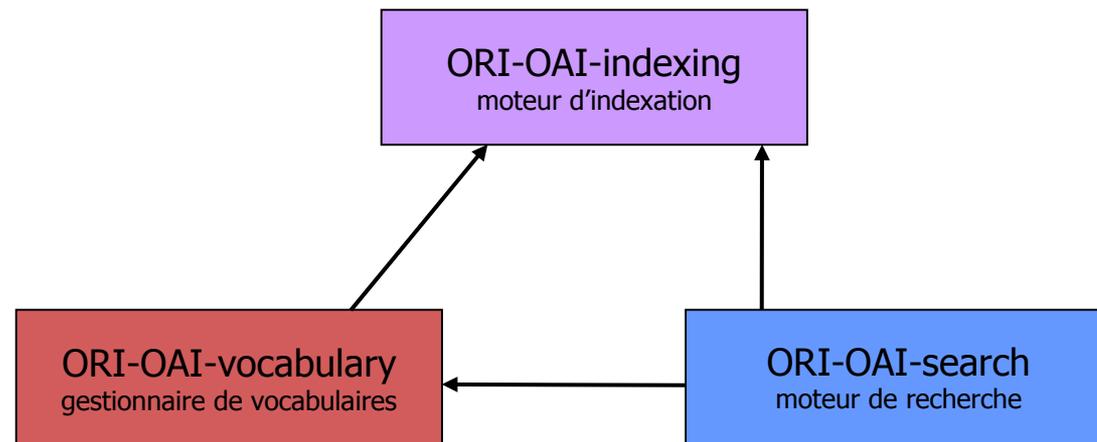


- Référencement du lien vers le(s) document(s)
- Indexation des métadonnées
- Saisie des métadonnées via vocabulaires
 - Vocabulaires dynamiques basés sur l'index
- Exposition OAI (optionnelle)
 - Exposition des fiches par requêtes sur l'index
 - Génération de sets OAI via le vocabulaire



- Moissonnage des fiches de métadonnées via OAI
- Indexation des métadonnées
- Exposition OAI (optionnelle)
 - Exposition des fiches par requêtes sur l'index
 - Génération de sets OAI via le vocabulaire

- Génération des interfaces à partir du vocabulaire
- Recherche dans l'index
- Traduction de certains résultats via vocabulaire



- ori-oai-md-editor
 - Saisie des métadonnées
 - Pré-configuré Dublin Core et LOM

- ori-oai-workflow
 - Point d'entrée du référencement
 - Processus de publication de fiches
 - Paramétrage très fin
 - Adaptable suivant les besoins
 - Pré-configuré Dublin Core et LOM

- ori-oai-indexing
 - Indexation des fiches locales et/ou moissonnées issues de fichiers XML
 - Recherche sur des métadonnées
 - Analyseur suivant la langue
 - Pré-configuré Dublin Core, LOM et CDM

- ori-oai-repository
 - Exposition OAI des fiches locales et/ou moissonnées

- ori-oai-harvester
 - Moissonnage OAI de fiches de métadonnées

- ori-oai-search
 - IHM du moteur de recherche
 - Différents modes de recherche
 - Pré-configuré Dublin Core, LOM et CDM

- ori-oai-vocabulary
 - Gestionnaire de tous les vocabulaires
 - Libellés, classifications, listes déroulantes, etc.
 - Hub de référentiels/vocabulaires
 - Différents types de vocabulaire
 - Statiques, dynamiques: LDAP, SQL, indexing, etc.

- **ESUP-serveur-WebDAV**
 - Solution de stockage des ressources
 - Connexion au SSO
 - Fédération d'identités (shibboleth)
 - Connexion possible à l'ENT

- Web services entre chaque module
- Possibilité de « débrancher » un module ou de venir se « brancher » depuis une application tierce
- Protection des URL d'accès à certains WS

Installation basique



Projet ORI-OAI

Environnement de la formation

- MySql distant sur un serveur avec 1 base par stagiaire
- Machines linux avec JDK + SVN + éditeurs de texte (vi et emacs)
- Un fichier **formationORI.zip** est à disposition
 - Annuaire ldap (openDS)
 - Ldapbrowser
 - Un apache tomcat avec l'application eXist (BD XML)
 - Tous les tomcats (attention : les fichiers *server.xml* doivent être modifiés pour correspondre aux bons numéros de ports)
 - Apache ANT
 - => lire le readme.txt et penser aux variables d'environnement env.sh

- Les variables
 - ORI_HOME = /tmp/ORI
 - JAVA_HOME = /usr/local/jdk1.5
- LDAP
 - localhost:1389
 - users
 - jack/auth, joe/adm, william/mod
 - Dans ou=people,dc=demo,dc=ori-oai,dc=org
- SQL
 - une base MySql par poste (en InnoDB!) sur un serveur mysql de Rennes1 distant

- Le principe
 - Status, diff, info, switch
- Le pourquoi
 - Maintenabilité
- LA commande de base
 - Checkout

=> `svn co http://subversion.cru.fr/ori-oai-commons/docs !`



TP 1

Mise en place de l'environnement

- Dézipper l'archive récupérée via subversion

- > `unzip formationORI.zip`

- Lire le readme.txt

- Variables d'environnements

- > `export CATALINA_OPTS="-Dfile.encoding=UTF-8 $CATALINA_OPTS"`

- > `export JAVA_HOME=/usr/local/jdk1.5`

- > `export ANT_HOME=????`

- > `export ORI_HOME=/tmp/ori (ici pour infos : $ORI_HOME est utilisée dans les slides)`

→ attention à la portée des variables d'environnement (session terminal, etc.) => ajout dans le `.bashrc` [ou `.tcshrc` etc...].

Si vous le souhaitez, vous pouvez aussi modifier votre environnement : nombre de bureaux virtuels, etc.

- Copier le contenu de l'archive dans \$ORI_HOME

- > `cp -r formationORI/* $ORI_HOME`

- Lancer openDS

- > `cd $ORI_HOME/openDS/bin`

- > `sh start-ds`

- Vérifier avec ldapBrowser que le LDAP fonctionne

- > `sh lbe.sh`

- Modifier la configuration d'eXist

- > `emacs $ORI_HOME/tomcat-exist/webapps/exist/WEB-INF/conf.xml`

- Dans la balise `indexer`, passer `validation` de `auto` à `no`

■ Accès aux serveurs tomcat

→ Note si besoin changer les droits dans le répertoire bin de tomcat:

```
> cd $ORI_HOME/tomcat-*/bin
```

```
> chmod a+x *.sh
```

■ Lancement des serveurs tomcat

```
> cd $ORI_HOME/tomcat-*/bin
```

```
> sh catalina.sh run
```

→ En production, lancer `sh startup.sh`

■ Vérifier que vous obtenez « server startup »



TP 2

Installation de ori-oai-indexing

- Introduction
 - Besoin des modules harvester et workflow
- Installation
 - Checkout
 - Répertoires
- Configuration
 - Modif du build.properties
 - ConfigIndexing.xml
 - Rep index
 - Rep fic temporaires
 - url des modules harvester et workflow
- Déploiement
- Tests
 - ant testIndex et ant testSearch
=> testIndex ajoute une entrée dans l'index, pour la supprimer, penser à supprimer l'index
 - ==> On supprime l'index

- Dans `$ORI_HOME`, création du répertoire qui contiendra l'index d'ori-oai-indexing
- `mkdir -p ${ORI_HOME}/data/indexes/index`
- `mkdir ${ORI_HOME}/data/indexes/tmp`
- Depuis `src` :
`svn checkout http://subversion.cru.fr/orioai-indexing/tags/1.0.1 ori-oai-indexing-svn`
- Depuis `ori-oai-indexing-svn`
 - Modif du `build.properties`
 - Modif du `properties/configIndexing.xml`
 - [rép. Index, fichiers tmp, urls/modules distants]
 - Modif du `log4j.properties` (rép logs du tomcat)
 - `${ANT_HOME}/bin/ant all`
- Depuis `tomcat-indexing/bin`, on lance le tomcat
 - `sh catalina.sh run`
- Test de l'URL : `http://localhost:8182/ori-oai-indexing/xfire/IndexingService?wsdl`
- Depuis `ori-oai-indexing-svn`, on peut lancer (pour test)
 - `${ANT_HOME}/bin/ant testIndex` (attention : écriture dans l'index)
 - `${ANT_HOME}/bin/ant testSearch`
- Supprimer l'index (stopper tomcat) :
 - `rm -r ${ORI_HOME}/data/indexes/index/*`



TP 3

Installation de ori-oai-vocabulary

- Introduction
 - Peut utiliser le module indexing, un ldap ...
- Installation
 - Checkout
- Configurations
 - Modif du build.properties
 - main-config.properties
 - url ldap
 - url module indexing
 - Les urls des modules vocabulary distants ne sont pas à modifier
 - ...
- Déploiement
- Tests

- Depuis src :
svn checkout <http://subversion.cru.fr/ori-vocabulary/ori-oai-vocabulary-spring/tags/ori-oai-vocabulary-1.0.0> ori-oai-vocabulary-svn
- Depuis ori-oai-vocabulary-svn
 - Modif du build.properties
 - Modif du conf/properties/main-config.properties
 - [ldap, exceptions, smtp, indexing]
 - `${ANT_HOME}/bin/ant deploy`
- Depuis tomcat-vocabulary/bin, on lance le tomcat
 - `sh catalina.sh run`
- Test de l'URL : <http://localhost:8183/ori-oai-vocabulary/xfire/OriVocabularyService?wsdl>
- Depuis ori-oai-vocabulary-svn, on peut lancer (pour test)
 - `${ANT_HOME}/bin/ant vocabularytests`
 - => tests ok si failures = 0



TP 4

Installation de ori-oai-search

■ Dépendances des modules

- Besoin de ori-oai-indexing et ori-oai-vocabulary

■ Téléchargement des sources

- > `cd $ORI_HOME`
- > `svn checkout http://subversion.cru.fr/ori-oai-search/tags/1.0.1 ori-oai-search-svn`
- > `cd $ORI_HOME/src/ori-oai-search-svn`

■ Configurations

- Modification du fichier build.properties
 - > `emacs build.properties`
 - Voir le support de formation pour le contenu du fichier
- Choisir une configuration du module (config.xml)
 - > `cd properties`
 - > `cp config.example.complet.xml config.xml`

- Éditer config.xml pour pointer vers les modules ori-oai-indexing et ori-oai-vocabulary
 - > `emacs config.xml`
 - Voir le support de formation pour le contenu du fichier
- Modifier la configuration du plugin openSearch
 - > `emacs simple_complet.xml`
 - Voir le support de formation pour le contenu du fichier

■ Déploiement

- > `cd $ORI_HOME/src/ori-oai-search-svn`
- > `$ANT_HOME/bin/ant all-servlet`

■ Tests

- Démarrer le serveur tomcat-search
 - > `cd $ORI_HOME/tomcat-search/bin`
 - > `sh catalina.sh run`
- Tester l'URL <http://localhost:8184/ori-oai-search>



TP 5

Installation de ori-oai-harvester

■ Dépendances des modules

- Besoin de ori-oai-indexing et de la base eXist

■ Téléchargement des sources

- > `cd $ORI_HOME`
- > `svn checkout http://subversion.cru.fr/ori-oai-harvest/tags/1.0.0
ori-oai-harvester-svn`
- > `cd $ORI_HOME/src/ori-oai-harvester-svn`

■ Configurations

- Modification du fichier `build.properties`
 - > `emacs build.properties`
 - Voir le support de formation pour le contenu du fichier
- Copier le fichier `ori.example.properties` en `ori.properties`
 - > `cd properties`
 - > `cp ori.example.properties ori.properties`

- Éditer `ori.properties` pour pointer vers les modules `ori-oai-indexing` et la base `eXist`

> `emacs ori.properties`

- `harvester.*` (pour utiliser `harvesterConfig.xml` et `reloadConfig=true`, copier/coller le `harvesterConfig.example.xml` en `harvesterConfig.xml`)
- `indexing.*`
- `exist.*`

→ Voir le support de formation pour le contenu du fichier

■ Déploiement

- > `cd $ORI_HOME/src/ori-oai-harvester-svn`
- > `$ANT_HOME/bin/ant deploy`

■ Tests

- Démarrer le serveur `tomcat-harvester`
 - > `cd $ORI_HOME/tomcat-harvester/bin`
 - > `sh catalina.sh run`
- Tester l'URL <http://localhost:8181/ori-oai-harvester>

■ Moisson

- On saisie une définition de moissons et on lance la moisson
- On vérifie dans l'interface du module `ori-oai-search` qu'on retrouve bien les fiches moissonnées : URL <http://localhost:8184/ori-oai-search>



TP 6

Installation de ori-oai-repository

■ Dépendances des modules

- Besoin de ori-oai-indexing et ori-oai-vocabulary

■ Téléchargement des sources

- > `cd $ORI_HOME`
- > `svn checkout http://subversion.cru.fr/ori-oai-repo/tags/1.0.0
ori-oai-repository-svn`
- > `cd $ORI_HOME/src/ori-oai-repository-svn`

■ Configurations

- Modification du fichier build.properties
 - > `emacs build.properties`
 - Voir le support de formation pour le contenu du fichier
- Copier le fichier ori-oaicat.example.properties en ori-oaicat.properties
 - > `cd properties`
 - > `cp ori-oaicat.example.properties ori-oaicat.properties`

- Éditer `ori-oaicat.properties` pour pointer vers les modules `ori-oai-indexing` et `ori-oai-vocabulary`, et définir l'identité de l'entrepôt

```
> emacs ori-oaicat.properties
  ■ identify.* (information pour définir l'identité de l'entrepôt)
  ■ indexing.*
  ■ vocabulary.*
```

→ Voir le support de formation pour le contenu du fichier

- Modifier `properties/repository-crosswalks.xml` -> voir support
- Modifier `webapps/WEB-INF/web.xml` (le `context-param` de nom `properties`)
- Modifier `WebContent/xsl/ihm_main.xsl` (nom du contexte) -> voir support

■ Déploiement

```
> cd $ORI_HOME/src/ori-oai-repository-svn
> $ANT_HOME/bin/ant deploy
```

■ Tests

- Démarrer le serveur `tomcat-repository`

```
> cd $ORI_HOME/tomcat-repository/bin
> sh catalina.sh run
```
- Tester l'URL <http://localhost:8180/ori-oai-repository>
- Tester la requête OAI-PMH <http://localhost:8180/ori-oai-repository?verb=Identify>



TP 7

Installation de ori-oai-md-editor

- Introduction
 - Besoin des modules vocabulary et workflow
- Installation
 - Checkout directement dans le webapps de Tomcat
- Configurations
 - Modif du fichier WEB-INF/resources/apps/ori-md-editor/config.xml
- On teste l'éditeur non connecté au module de workflow : édition rapide (et partielle!) d'une fiche LOM

- Depuis `tomcat-editeur/webapps` :
`svn checkout http://subversion.cru.fr/ori-workflow/ori-oai-md-editor/tags/ori-oai-md-editor-1.0.0 ori-oai-md-editor`
- Depuis `ori-oai-md-editor`
 - Modif de `WEB-INF/resources/apps/ori-md-editor/config.xml`
- Depuis `tomcat-editeur/bin`, on lance le tomcat
 - `sh catalina.sh run`
- Depuis `http://localhost:8186/ori-oai-md-editor` tester un formulaire ...



TP 8

Installation de ori-oai-workflow

- Introduction
 - Besoin des modules md-editor, indexing
- Création de la collection pour ori-oai-workflow
- Installation
 - Checkout
- Configurations
 - Modif du fichier conf/properties/main-config.properties
 - Modifications permissions/groupes
- Déploiement
- Initialisation de la base de données
- Tests
- On teste le workflow, l'éditeur, la publication d'une fiche de métadonnées.

- Via `http://localhost:8187/exist` , lien admin, créer une collection `/db/ori/workflow`
- Depuis src :
svn checkout `http://subversion.cru.fr/ori-workflow/ori-oai-workflow-spring/tags/ori-oai-workflow-spring-1.0.0` ori-oai-workflow-svn
- Depuis ori-oai-workflow-svn
 - Modif de `build.properties`
 - Modif de `conf/properties/main-config.properties`
 - => pour les groupes ldap, pour `groupSearchbase`, supprimer `ou=groups`
 - Modif de `conf/properties/spring/acegi/acegi-authentication-additional-groups.xml`
 - ant deploy
 - => compilation et déploiement
 - ant init
 - => initialisation de la base de données
 - ant update-acls
 - => mises à jour des permissions/rôles par défaut dans la base de données
- Depuis `tomcat-workflow/bin`, on lance le tomcat
 - `sh catalina.sh run`
- Tester l'interface `http://localhost:8185/ori-oai-workflow`



Projet ORI-OAI

Solution de stockage des ressources

- Versions
 - V1 → non intégration
 - Futur
 - Intégration en 1.5
 - Evolution du serveur
- Serveur et Canal stockage
- Copier / coller
 - Avec canal stockage
 - Sans canal stockage

Configuration avancée

- Certains modules ne sont pas configurés avec la métadonnée « **semantic density** » du LOM
 - [pour infos: non présente dans le LOMFR]

- Cette métadonnée peut admettre un certain nbre de valeurs définies par LOM:
 - Very low
 - Low
 - Medium
 - High
 - Very high

- But
 - Intégration de cette métadonnée dans les différents modules

TP 9

Ajout d'un vocabulaire dans ori-oai-vocabulary

- Nouveau vocabulaire
common_lom_educ_semantic density
- Vocabulaire de type « statique » et local au module
(spécifique à notre installation ori-oai)

- Création du nouveau fichier XML de vocabulaire `conf/properties/custom_vocabularies/common_lom_educ_semantic_density.xml`
 - On copie/colle le fichier `conf/properties/ori_vocabularies/common/lom/educ_difficulty.xml` pour point de départ
- Ajout d'un point d'entrée de déclarations de vocabulaires « custom » dans `conf/properties/domain/domain.xml`
 - `<import resource="customVocabulary.xml" />`
- Création de `customVocabulary.xml` (en se basant sur `oriVocabularies.xml` qui déclare des vocabulaires de type statique).
 - => on définit un nouveau bean de type `XmlStaticVocabularyProvider` avec pour id **`common_lom_educ_semantic_density`**
- On déploie (`ant deploy`)
- On teste via l'éditeur de métadonnées :
 - `http://NOM_SERVEUR_ORI:8186/ori-oai-md-editor/ori-md-editor/vocab/common_lom_educ_semantic_density`

TP 10

Saisie de la nouvelle métadonnée dans ori-oai-md- editor

- L'éditeur « auteur » / « Fiche simple » ne donne pas la possibilité d'éditer la métadonnée « semantic density »
- => on ajoute dans le formulaire la « widget » qui permet l'édition de cette métadonnée en s'appuyant sur le vocabulaire préalablement défini et déclaré.
- Les fichiers spécifiques à l'éditeur de métadonnées se trouvent dans le répertoire WEB-INF/resources/apps/ori-md-editor/
 - Description
 - Xforms : contient les formulaires XForms
 - Prototypes : contient des fichiers XML contenant tous les blocs XML possibles
 - xml-blank : ...
 - ...
 - => on va modifier certains d'entres eux.

- Xforms/lom-author-light/main-form.xhtml
 - « XForms Model »
 - Instance de vocabulaire common_..._density.xml
- Xforms/lom-author-light/content-xforms.xml
 - « XForms Content »
 - Ajout du bloc « semantic density » (en prenant exemple sur le bloc « learning resource type »)
- I18n/fr_FR.xml / i18n/en_EN.xml
 - Ajout des balises add/del-semanticDensity
- Xml-blank/lom-author-light-blank.xml
 - On ajoute par défaut le bloc semanticDensity dans le XML « d'initialisation ».
 - Valable pour le mode non connecté au workflow seulement ...
- => tester le formulaire modifié

- Pour ajouter une nouvelle métadonnée non LOM, il faut tenir compte des espaces de noms.
- Un espace de noms permet dans un fichier XML d'identifier de manière unique les éléments et attributs d'un fichier XML
- Ici l'espace de noms associé au LOM est **<http://ltsc.ieee.org/xsd/LOM>**, aussi nous avons ajouté la métadonnée « semantic density » dans cet espace de noms.
- Si nous voulions prendre en compte une métadonnée spécifique à notre établissement, au LOMFR, etc. nous devrions le faire dans l'espace de noms spécifique à notre établissement, LOMFR, etc.
- => voir en annexe, l'exemple plus poussé de l'ajout d'une métadonnée dans l'éditeur ORI-OAI

TP 11

Obligation de saisie de la nouvelle métadonnée dans ori-oai-workflow

- On veut rendre obligatoire la saisie de la métadonnée « semantic density »
- == on ne veut pas que l'auteur puisse demander la validation de sa fiche si celui-ci n'a pas saisie cette métadonnée
- => on modifie le workflow pour ajouter/modifier une condition sur une transition.

- Modification de `properties/spring/osworkflow/workflows/workflow_easy.xml`
 - `<arg name="xpath7">/lom:lom/lom:educational/lom:semanticDensity/lom:value[normalize-space(.) != "]</arg>`
- On redéploie (`ant deploy`) le workflow et on redémarre

TP 12

Indexation de la nouvelle métadonnée dans ori-oai-indexing

- Modification du fichier de configuration de LIUS

- 3 ajouts à apporter
 - Partie indexation : balise luceneField avec attributs : xpathSelect, name, et type (ConcatDate, Vcard ou Text)
 - Partie recherche
 - Partie affichage de résultats

- `ORI_HOME/src/ori-oai-indexing-svn/properties/liusConfig.xml`
- Ajout de la balise `luceneField` dans la balise `index`, à l'intérieur de la partie LOM (5. Pédagogie)
- Ajout du `xpath` encodé dans la balise `<multiFieldQueryParser>` (Attention à précéder d'une virgule)
- Ajout de la balise `luceneField` dans `<fieldsToDisplay>`
- `ant all`
- Redémarrage du Tomcat

TP 13

Recherche dans la nouvelle métadonnée dans ori-oai-search

1) Modification de la recherche avancée

- Ajout d'un champ de recherche pour « semantic density »

→ modification de la config du formulaire de recherche avancée

```
> cd $ORI_HOME/src/ori-oai-search-svn/properties/advanced  
> emacs lom_advanced.xml
```

→ Voir le support de formation pour le contenu du fichier

→ ajout du libellé du champ dans les fichiers i18n

```
> cd $ORI_HOME/src/ori-oai-search-svn/properties/messages  
> emacs forms_fr.properties
```

→ Voir le support de formation pour le contenu du fichier

1) Affichage des résultats

- Ajout de la métadonnée « semantic density » dans la liste des résultats de recherche

→ modification de la configuration des résultats

```
> cd $ORI_HOME/src/ori-oai-search-svn/properties  
> emacs config.xml
```

→ Voir le support de formation pour le contenu du fichier

→ ajout du libellé du champ dans les fichiers i18n

```
> cd $ORI_HOME/src/ori-oai-search-svn/properties/messages  
> emacs menus_fr.properties
```

→ Voir le support de formation pour le contenu du fichier

1) Affichage de « semantic density » dans la notice

→ modification de la XSL de transformation

- > `cd $ORI_HOME/src/ori-oai-search-svn/webapp/WEB-INF/xsl`
- > `emacs lom1.xsl`

→ Voir le support de formation pour le contenu du fichier

→ ajout du libellé du champ dans les fichiers i18n

- > `cd $ORI_HOME/src/ori-oai-search-svn/properties/messages`
- > `emacs xsl_fr.properties`

→ Voir le support de formation pour le contenu du fichier

→ ajout du passage de paramètres depuis la JSP

- > `cd $ORI_HOME/src/ori-oai-search-svn/webapp/WEB-INF/stylesheets/notice`
- > `emacs notice.jsp`

→ Voir le support de formation pour le contenu du fichier

→ ajout de la traduction de la valeur de la métadonnée

- > `cd $ORI_HOME/src/ori-oai-search-svn/properties`
- > `emacs config.xml`

→ Voir le support de formation pour le contenu du fichier

■ Déploiement des modifications

- > `cd $ORI_HOME/src/ori-oai-search-svn`
- > `$ANT_HOME/bin/ant all-servlet`

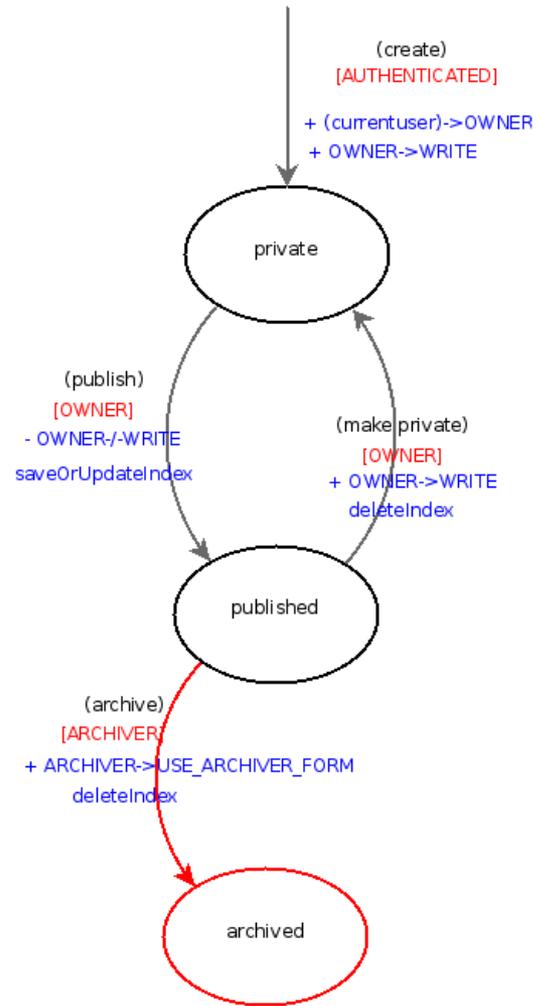
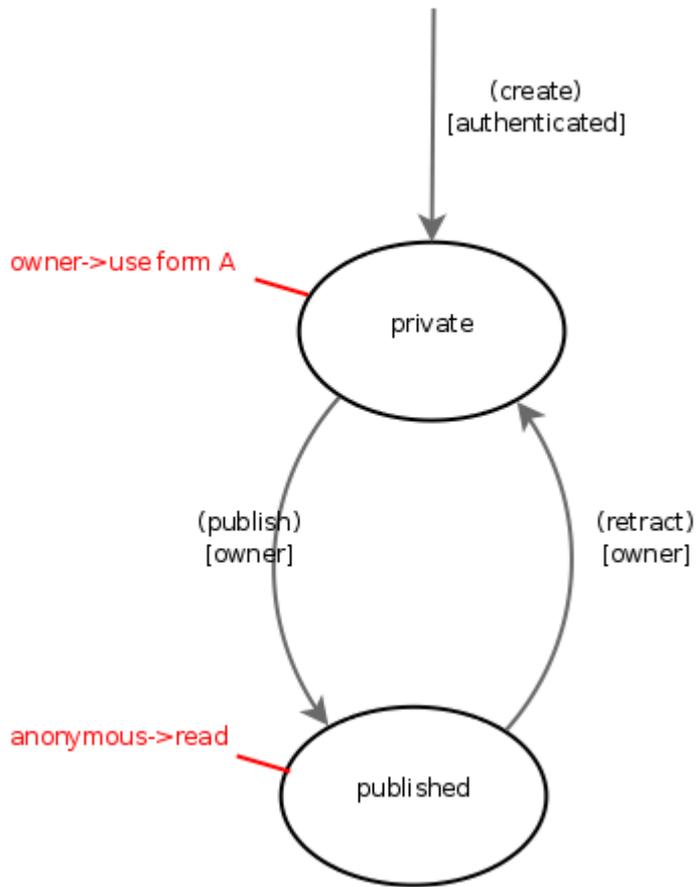
■ Tests

- Arrêter le serveur tomcat-search
- Démarrer le serveur tomcat-search
 - > `cd $ORI_HOME/tomcat-search/bin`
 - > `sh catalina.sh run`
- Tester les changements depuis l'URL
 - `http://localhost:8184/ori-oai-search`



TP 14

Modification/personnalisation d'un workflow dans ori-oai-workflow



- But : Modification/Personnalisation d'un workflow
 - Ajout d'un état
 - Ajout d'une transition
 - Avec actions/conditions sur la transition
 - => Ajout de permissions/rôles
 - Ajout d'un formulaire spécifique à l'état archived
 - Ajout de 2 « catégories »
 - « ressources publiées » (pour l'archiviste)
 - « mes documents archivés » (pour le propriétaire et l'archiviste)

- Ajout de la permission et du rôle relatif à l'archivage
 - acegi-permissions.xml
- Ajout d'une « action » et d'un « step » osworkflow
 - workflow_very_easy.xml
- Ajout de la déclaration d'un formulaire pour OAI_DC
 - spring-metadata-types.xml
- Création du formulaire
 - ori-oai-md-editor
- Création d'un groupe virtuel
 - Acegi-authentication-additional-groups.xml
- Affectation dun rôle à un groupe virtuel
 - Acegi-acls-root.xml
- Ajout de 2 Catégories « ressources publiées », « mes documents archivés »
 - Spring-categories.xml
 - Properties/spring/common/i18n/bundles/Custom_*.properties

- Copyright © 2008 Consortium ORI-OAI
 - Ce document peut être copié et distribué dans son intégralité, sans modification, retrait ou ajout.
 - Tout usage commercial est interdit.
 - L'utilisation de ce document dans un cadre de formation collective est soumise à l'approbation explicite et préalable de ses auteurs.