



Université du Québec

École de technologie supérieure

Base de données texte

Survol du cours

- **Types de manipulation de textes**
- Introduction à la recherche d'information textuelle
- Manipulation de base des types de données texte en SQL
- Méthodes statistiques de manipulation de textes
- Exemple d'implémentation : Oracle Text

Techniques de manipulation du texte

- Repérage de l'information (TR : Text Retrieval)
- Extraction de l'information (SDE : Schema-Directed Extraction – predefined schema structure)
- Autres manipulations de textes (QDE : Query-Directed Extraction - Ontologies)

Repérage de l'information (TR)

- Recherche textuelle = filtrage + tri des documents selon leur pertinence
 - parfois séquentiels
 - parfois réalisés simultanément
- Types de filtrage
 - Correspondance exacte aux mots et phrases
 - Recherches inexactes
 - Recherche en proximité
 - Recherche intelligentes, ex: par thèmes, poids

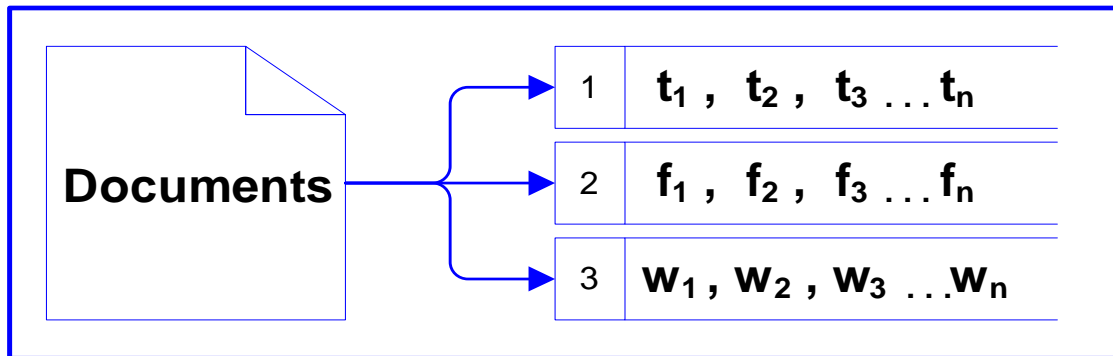
Repérage de l'information (TR)

- Tri : approximation de la précision par une fonction mathématique tenant compte de divers facteurs
 - nombre d'occurrences des mots de la requête dans le doc retrouvé,
 - précision des mots de la requête (plus un mot est fréquent dans la langue, moins il est précis),
 - proximité des mots de la requête dans le document
 - position des occurrences des mots de la requête dans le document
 - ...
 - on privilégie les documents
 - employant fréquemment les mots de la requête,
 - en assignant un poids plus élevé aux termes rares dans la langue,
 - en favorisant les documents employant les termes de la requête à proximité les uns des autres,
 - et en favorisant les documents employant les termes de la requête tôt dans le document

Repérage de l'information (TR)

Requêtes et documents

- ➔ document = { termes }
- ➔ tf_{ik} : fréquence du $i^{\text{ème}}$ terme dans le document k
- ➔ idf_i : fréquence documentaire inverse du $i^{\text{ème}}$ terme
- ➔ w_{ik} : poids du $i^{\text{ème}}$ terme dans le document k



Repérage de l'information (TR)

● Pondération des termes par document

Ex.: <http://fr.wikipedia.org/wiki/TF-IDF>

$$tf_{i,k} = \frac{tf_{i,k}}{\max_k tf_{i,k}}$$

$$idf_i = \log \frac{N}{n_i}$$

N: nombre total de documents dans le corpus
n_i: nombre de documents qui contient le terme t_i

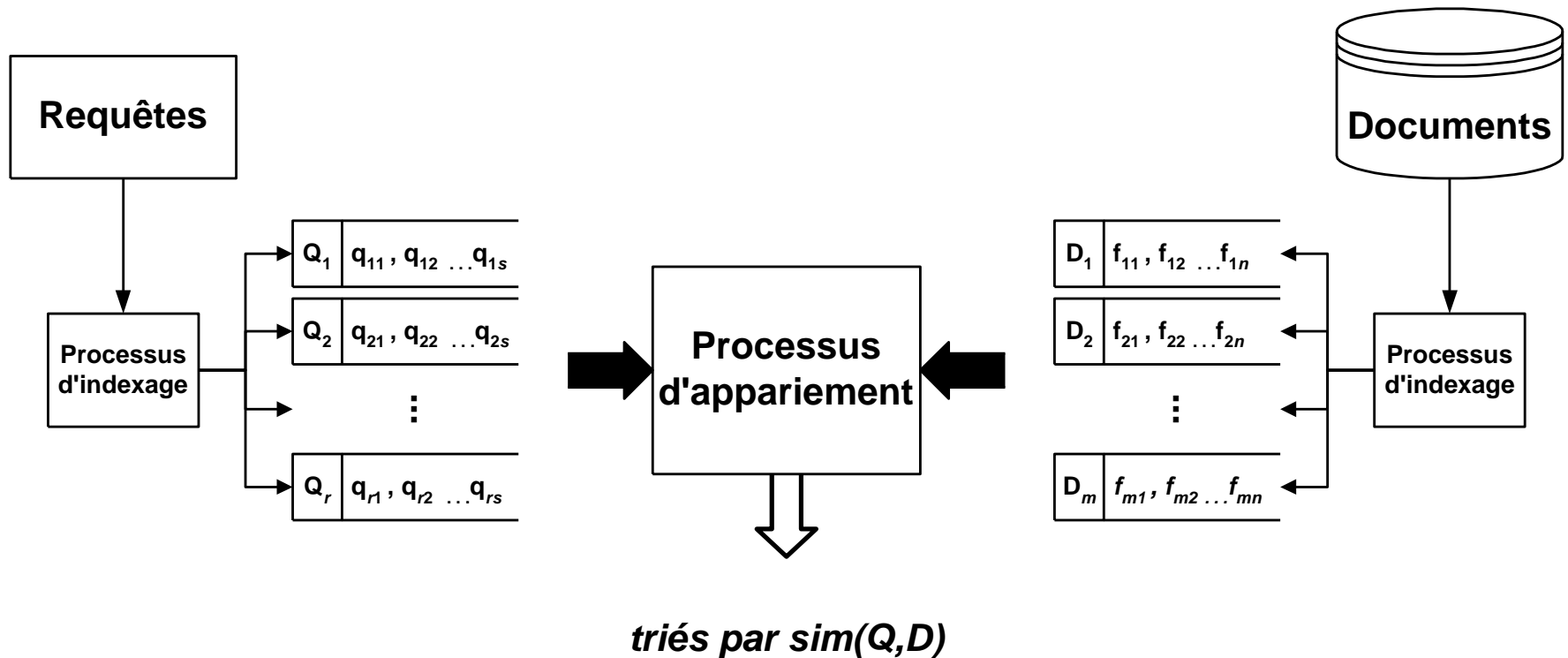
$$w_{i,k} = tf_{i,k} \times idf_i$$

Repérage de l'information (TR)

● Appariement requête-documents

$$sim(q, d_k) = \frac{\sum_{i=1}^n w_{i,k} \times w_{i,q}}{\sqrt{\sum_{i=1}^n w_{i,k}^2} \times \sqrt{\sum_{i=1}^n w_{i,q}^2}}$$

Repérage de l'information (TR)



Repérage de l'information (TR)

- Variations d'utilisation

- recherche ad hoc : (repérage)

- les besoins de l'utilisateur changent constamment,
 - la BD textuelle évolue peu
 - Exemple : moteurs de recherche du Web

- recherche récurrente : (filtrage)

- les besoins de l'utilisateur changent peu,
 - la BD évolue constamment
 - Exemple : application recherchant sur le Web les documents nouveaux sur un thème fixe
 - Possibilité de construire un profil utilisateur

Repérage de l'information (TR)

- Variations d'utilisation : l'outil de recherche peut être de trois niveau de complexité:
 - monolingue :
 - les documents sont dans une seule langue
 - la requête est exprimée dans cette langue
 - multilingue :
 - les documents sont dans diverses langues
 - on ne cherche que les documents exprimés dans la langue de l'utilisateur
 - translingue :
 - les documents sont dans diverses langues
 - on cherche tous les documents satisfaisant le besoin en information de l'utilisateur, peu importe la langue de la requête



Extraction de l'information (SDE)

- Manipule de larges collections de textes, e.g. emails
- Documents traités un par un
- Extrait des informations spécifiées par un schéma prédéfini
- Ignore les autres informations des documents
- Focalise sur le contenu qui répond aux critères
- L'objectif est de remplir le schéma plutôt que de répondre à une requête
- Représente des objets, des propriétés et des relations de l'ontologie du domaine

Désavantages de cette approche

- Ne traite pas le corps du document – choisi des mots reconnus facilement et les insèrent dans le schéma
- On doit utiliser des méthodes additionnelles pour identifier des mots complexes et des phrases dans le contenu du texte
- Le schéma résultant est utilisé comme métadonnées pour des requêtes

Autres manipulations de textes (QDE)

- Résumé automatique de texte
- Catégorisation automatique de textes selon une série de catégories prédéfinies (classification):
 - notamment le routage automatique de documents : catégorisation automatique des courriels reçus
- Regroupement de textes similaires, sans catégories prédéfinies (regroupement - *clustering*)
- **Dans ce qui suit, on ne traitera que du repérage de l'information ad hoc unilingue**

Plan du cours

- Types de manipulation de textes
- **Introduction à la recherche d'information textuelle**
- Manipulation de base des types de données texte en SQL
- Méthodes statistiques de manipulation de textes
- Exemple d'implémentation : Oracle Text

Qualité d'une extraction de texte

L'outil d'extraction doit permettre l'équilibre entre rappel et précision

- Deux mesures : **rappel** et **précision**
- Évaluent la qualité de l'engin à trouver tous et seulement les documents pertinents
- $rappel = |Extraits \cap Pertinents| / |Pertinents|$
 - Mesure la couverture des documents repérés par l'engin p/r aux documents pertinents
 - Ex: repère 8 doc.p / 10 pertinents : rappel = 80%
- $précision = |Extraits \cap Pertinents| / |Extraits|$
 - Mesure la pertinence des documents repérés par l'engin p/r au total des documents repérés
 - Ex: repère 8 doc.p / 40 repérés : précision = 20%
- Précisions extrapolées aux rappels 0%, 10%,...,100%
- Idéal : rappel = 100% et précision = 100%

Principes de la recherche d'information textuelle

- En recherche basée sur le contenu, on cherche les documents traitant de concepts donnés, éventuellement dans une relation particulière:
- Par exemple, je cherche:
 - Des documents traitant des bases de données;
 - Des documents sur les aspects novateurs des BDR (Base de données relationnelle);
 - Des documents comparant les mérites respectifs des BDRO (Base de données relationnelle-objet) et des BDOO (Base de données orientée-objet).

Principes de la recherche d'information textuelle

- On suppose que le critère de recherche est exclusivement thématique:
 - pas de recherche sur le type de document (scientifique/technique/de vulgarisation, son style, son niveau de langage ...
- On considère aussi que les mots individuels correspondent à des concepts.

Quelques problèmes de la recherche d'information textuelle

1) Problème de délimitation des mots:

- Nombreuses langues asiatiques n'ont pas de délimiteur de mots;
- Nécessite la segmentation préalable du texte avant indexation et recherche

Ex: Trokenbeerenauslese = Troken beeren auslese

Quelques problèmes de la recherche d'information textuelle

2) Problème de correspondance entre mots individuels et concepts:

- une **notion d'expression** est nécessaire à l'identification d'un concept : "bases de données"
 - Recherche sur les mots individuels alors que l'on cherche le concept : précision trop faible
- certaines langues germaniques construisent des expressions sous la forme d'un seul mot: allemand Weltkrieg = guerre mondiale (Welt = monde, Krieg = guerre)
 - Recherche sur Krieg : rappel trop faible, car on ignorera les occurrences de Weltkrieg !

Quelques problèmes de la recherche d'information textuelle

On doit traiter les variations dans le texte

- 3) Variations morphologiques : généralement, le pluriel et le singulier sont équivalents *base* = *bases* ;
- 4) Variations morphosyntaxiques des expressions : configuration de la base de données = configuration de bases de données

Quelques problèmes de la recherche d'information textuelle

5) Variations sémantiques:

- **Homonymie** : plusieurs sens sans liens entre eux : fraise (fruit) et fraise (outil);
- **Polysémie** : plusieurs sens reliés mais distincts : banque (institution financière) et banque (de données), journal (publication) et journal (intime);
- certaines de ces ambiguïtés sont levées de façon implicite:
 - requête *recettes avec des fraises* : il est peu probable que l'on retrouve des documents utilisant le sens fraise outil.
- **Synonymie** : plusieurs mots de sens quasi-identique : melon d'eau = pastèque, travail = job, épais = idiot.

Plan du cours

- Types de manipulation de textes
- Introduction à la recherche d'information textuelle
- **Manipulation de base des types de données texte en SQL**
- Méthodes statistiques de manipulation de textes
- Exemple d'implémentation : Oracle Text

Avec SQL2 ou SQL3

- Le SQL2 (le SQL le plus répandu) n'offre que des fonctions limitées de requêtes pour des chaînes de caractères (like, present)
- Le SQL3 (la nouvelle norme du SQL) offre des fonction étendues pour les textes. OracleText est basé sur le SQL3

Rappel des requêtes texte en SQL2

- Région = 'France' ← *correspondance exacte* p. 325
- Région IN ('France', 'Spain')
- *Correspondance Inexacte*
 - Where SOUNDEX (var1)=SOUNDEX('Errazuriz') p. 327
 - Where var LIKE 'Erraz%' p. 327 Where var LIKE 'Erraz_'
- LIKE 'P_b%medium_bodied%' p. 327
- LIKE '%medium Bodied%' p. 328
- INSTR(wine_name, 'au') p. 328

Stockage d'un texte

- Si un document est semi structuré, il y a possibilité de stocker chaque champ séparément, par exemple:
 - Des courriels: suivant la structure (Expéditeur, Récipiendaires, Date, Sujet, Contenu du message)
 - Un Curriculum Vitae (CV): 4 champs d'entête et le reste du texte dans un champ texte CV.

Donc certains champs sont traités d'une manière classiques, d'autres sont textuels

Exemple: Stockage d'un CV

Table employés, avec CV en mode texte

```
CREATE TABLE employes (  
    id_employe      INTEGER,  
    nom_employe     VARCHAR(20),  
    prenom_employe  VARCHAR(20),  
    date_naissance  DATE,  
    cv              VARCHAR(2000)  
)
```

Stockage du texte d'un CV

- Contenu de la table employés:

ID_EMPLOYE	NOM_EMPLOYE	PRENOM_EMPLOYE	DATE_NAISS	CV
1	April	Alain	1966-12-11	Spécialiste en bases de données
2	Garant	Paul	1986-01-20	Données de bases du CV

Exemples d'utilisation SQL2

- Recherches booléennes
 - Recherche des employés dont le CV contient les mots 'bases' et 'données'
 - en tenant compte de la casse (casse = maj. et min.)

```
SELECT id_employe
FROM employes
WHERE cv LIKE '%bases%'
AND cv LIKE '%données%'
/
ID_EMPLOYE
-----
1
```

Exemples d'utilisation SQL2

- Recherches booléennes
 - Recherche des employés dont le CV contient les mots 'bases' et 'données'
 - sans tenir compte de la casse

```
SELECT id_employe
FROM employes
WHERE lower(cv) LIKE '%bases%'
AND lower(cv) LIKE '%données%'
/
ID_EMPLOYE
-----
1
```

lower() retourne la chaîne de caractères en minuscules

upper() retourne la chaîne de caractères en majuscules

Exemples d'utilisation SQL2

- Recherches booléennes
 - Recherche des employés dont le CV contient les mots 'bases' et 'données'
 - sans tenir compte de la casse, mais en tenant compte de l'ordre (*bases* avant *données*)

```
SELECT id_employe
FROM employes
WHERE lower(cv) LIKE '%bases%données'
/
ID_EMPLOYE
-----
1
```

Exemples d'utilisation SQL2

- Limitations fonctionnelles
 - Pas de recherche basée sur la proximité:
 - *bases* et *données* dans une fenêtre de 10 mots.
 - Pas d'ordonnancement des résultats:
 - à moins de créer une fonction utilisateur SCORE qui retournerait un 'score' = implantation complexe.
 - Pas de recherche sur des fichiers externes, ni sur des fichiers textes en format binaire (.DOC, .PPT, .PDF) et définis comme BFILE ou BLOB.

Exemples d'utilisation SQL2

- Limitations techniques : performance très mauvaise sur de grands textes:
 - si on représente *CV* par un CLOB ou un VARCHAR(4000), on peut indexer cv, mais index utilisable uniquement dans quelques cas
 - cv = 'blablabla'
 - cv IN ('blablabla','toto','titi')
 - cv LIKE 'blablabla%' (% final uniquement)
 - Index inutilisables sur *fonction*(cv), sur LIKE avec % ou _ en position autre que finale ...

Plan du cours

- Types de manipulation de textes
- Introduction à la recherche d'information textuelle
- Manipulation de base des types de données texte en SQL
- **Implémentation d'un moteur de recherche**
- Exemple d'implémentation : Oracle Text

Principe de l'implémentation d'un moteur de recherche

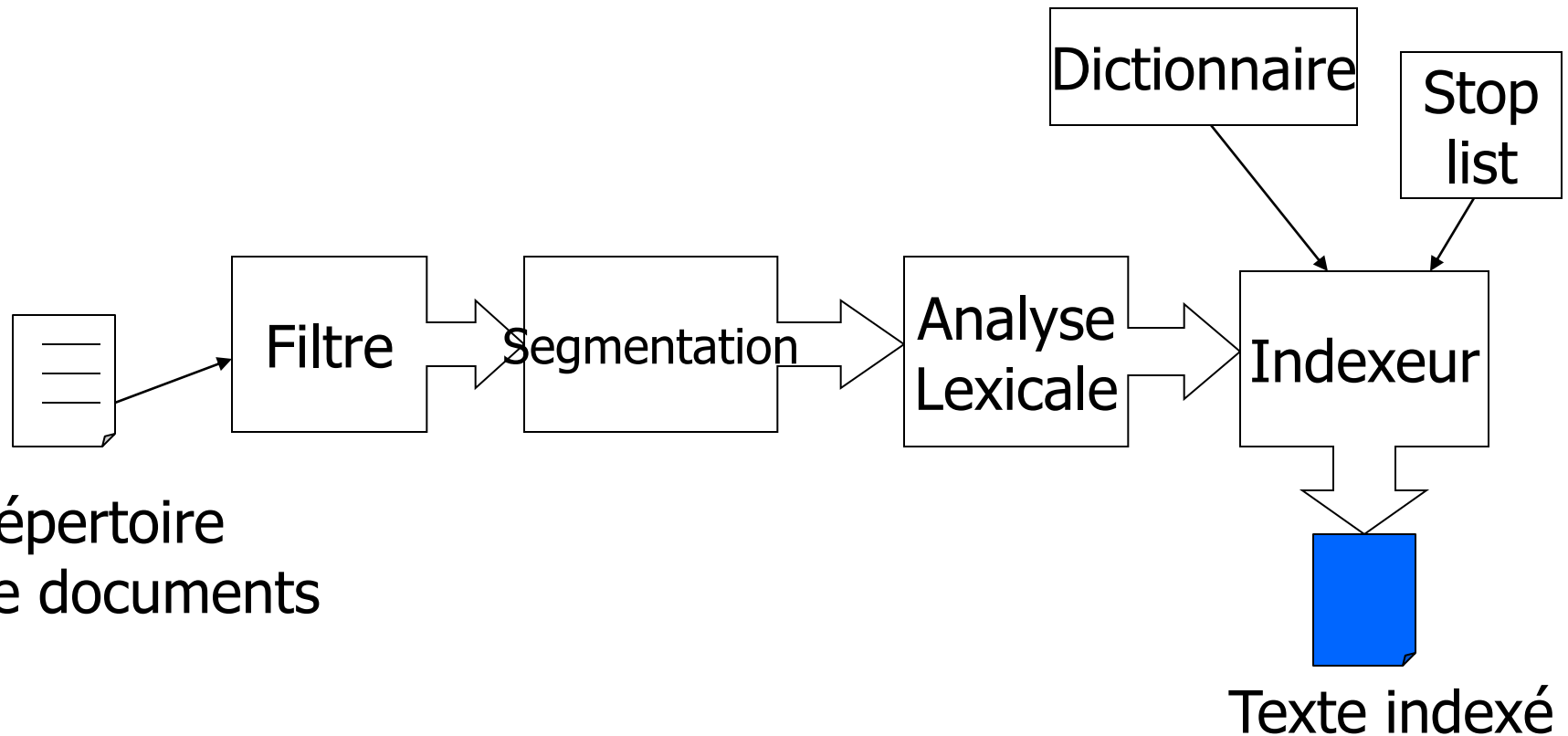
A- Création de la collection de textes

- Traitement préliminaire des textes à indexer : optionnel
- Indexation des textes traités : obligatoire

B- Traitement des requêtes de l'utilisateur

- Reformulation de la requête de l'utilisateur : optionnel
- Exécution de la requête : obligatoire

Étapes typiques de préparation du texte



A- Traitement préliminaire des textes

- 1) Suppression de la casse et/ou de l'accentuation:
 - peut augmenter l'ambiguïté : (la) marche/ (le) marché;
- 2) Correction orthographique automatisée;
- 3) Assimilation de diverses variantes orthographiques:
 - en anglais : centre et center;
 - abréviations/acronymes : ONU remplacé par *Organisation des Nations Unies*.
- 4) Segmentation du texte en mots (langues asiatiques);
- 5) Segmentation des mots composés en mots (langues germaniques);

A- Traitement préliminaire des textes

6) Lemmatisation : on élimine les variations morphologiques

- marchee, marchai, marchais...
- journal, journaux

7) Élimination des mots vides :

- mots très fréquents de la langue : *de, le, la ...*;
- diminue la taille de l'index;
- mais empêche l'exécution de requêtes basées sur des expressions ("*base **de** données*").

8) Scission du texte en différents passages :

- de taille arbitraire;
- ayant une thématique spécifique : utile pour des documents traitant de divers thème;
- chaque passage sera traité comme un document.

9) ...et bien d'autres...



Université du Québec

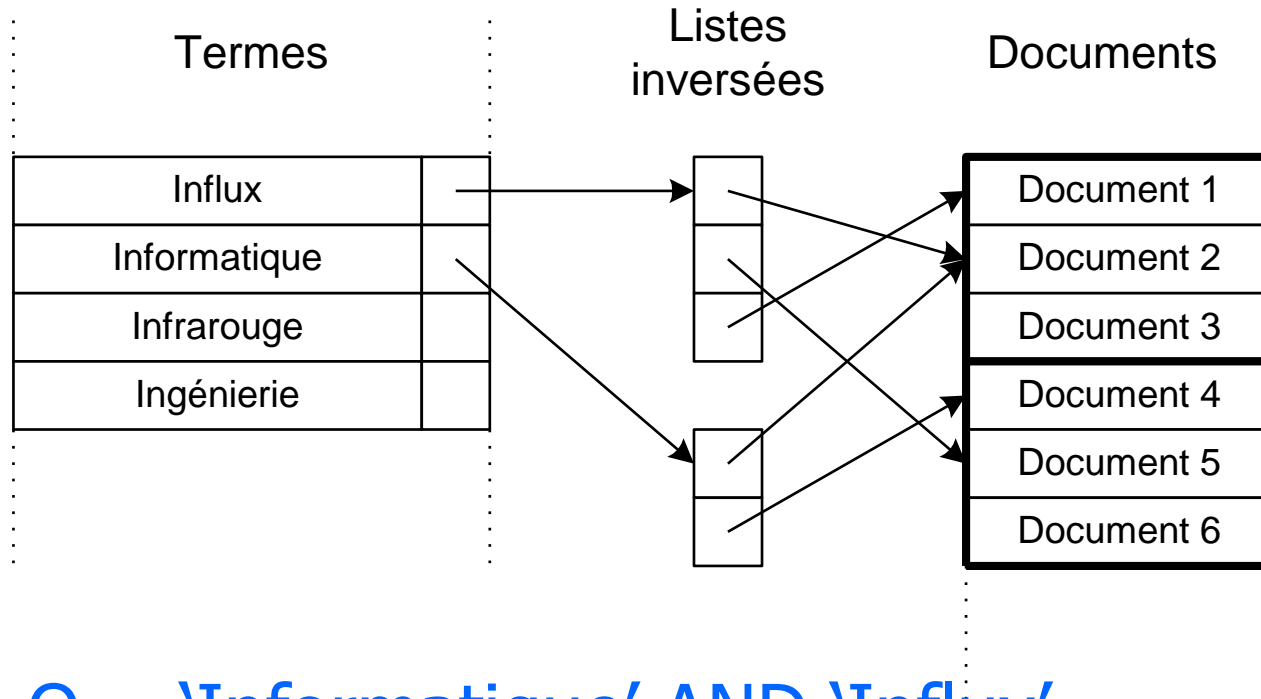
École de technologie supérieure

Département de génie logiciel et des TI

A- Indexation des textes

- 1) Indexation plein texte au moyen du fichier inversé
- 2) Index = liste des mots apparaissant dans la collection de documents
- 3) À chaque mot est associé la liste des documents où il apparaît
- 4) Optionnellement complétée par d'autres infos
 - positions exactes du mot dans le document
 - nombre d'occurrences du mot dans le document
 - Etc.

A- Indexation des textes – liste inversée (de Robert Godin)



Q = 'Informatique' AND 'Influx'

A- Indexation des textes : exemple

- Documents :
 - d1 : Il est entré dans le marché des actions.
 - d2 : Entre autres actions, il a marché pour le Tibet.
- Index : avec liste de positions
 - a d2(5)
 - actions d1(8), d2(3)
 - autres d2(2)
 - dans d1(4)
 - des d1(7)
 - entre d2(1)
 - entré d1(3)
 - ...

B- Reformulation automatique de la requête de l'utilisateur

- Ajout de termes à la requête de l'utilisateur :
 - ajout de variantes orthographiques;
 - ajout des variantes morphologiques d'un mot (si index direct sur les mots, sans lemmatisation);
 - ajout de synonymes;
 - ...
- Remplacement de termes de la requête de l'utilisateur, si transformation préliminaire des documents:
 - les variantes morphologiques sont remplacées par la forme canonique retenue;
 - ...

B- Exécution de la requête de l'utilisateur

- 2 approches principales : **booléenne**, **vectorielle**
- Approche **booléenne** pure
 - requêtes exprimées sous forme de formules booléennes
Ex: recherche de documents sur les 'bases de données' autres que 'hiérarchiques' et 'réseaux'
 $Q = \text{'bases'} \text{ AND } \text{'données'} \text{ AND NOT } (\text{'hiérarchiques'} \text{ OR } \text{'réseaux'})$
- Approche **vectorielle** pure
 - requêtes exprimées sous forme de liste de mots (vecteur)
Ex: recherche de documents sur les 'bases de données relationnelles'
 $Q = \{ \text{'bases'}, \text{'données'}, \text{'relationnelles'} \}$

B- Exécution de la requête de l'utilisateur

- Exécution des requêtes booléennes
 - À chaque mot on associe la liste de documents associés
 - OU est implémenté comme union de listes
 - ET est implémenté comme intersection de listes
 - NON est implémenté comme différence de listes
 - Si les positions des mots sont enregistrées dans l'index, on peut implémenter un NEAR (proximité)
 - mot1 NEAR mot2 si mot1 et mot2 sont distants de moins de 10 mots (par exemple)
- Très performant, mais
 - implémente un repérage sans tri (similarité)
 - souffre du problème du 'tout ou rien'

B- Exécution de la requête de l'utilisateur

- Exécution des requêtes vectorielles

- chaque document est considéré comme un vecteur dans un espace à N dimensions, où N est le nombre de mots de l'index (corpus - N est très grand)
- Exemple à partir de l'index précédent (limité aux 7 premières dimensions)

- a d2(5)
- actions d1(8), d2(3), d3(2)
- autres d2(2), d3(1)
- dans d1(4)
- des d1(7)
- entre d2(1)
- entré d1(3)

d1 : Il est entré dans le marché des actions.

d2 : Entre autres actions, il a marché pour le Tibet.

- $d1 = \{0,1,0,1,1,0,1\}$ et $d2 = \{1,1,1,0,0,1,0\}$ d3 : Autres actions associées.

B- Exécution de la requête de l'utilisateur

- Exécution des requêtes vectorielles
 - la requête est également considérée comme un vecteur dans cet espace à N dimensions
 - Exemple avec une représentation binaire
 $d3 = \text{'Autres actions associées.'}$
 - Plusieurs autres représentations possibles
 - fréquences (tf)
 - fréquences normalisées ($tf / \max(tf)$)
 - poids ($tf \times idf$)
 - entropie (notion d'incertitude)
 - etc.

B- Exécution de la requête de l'utilisateur

- Exécution des requêtes vectorielles

- la pertinence de chaque document est déterminée par une mesure de similarité entre le vecteur requête et chaque vecteur document
- exemple : le cosinus de l'angle entre les deux vecteurs

$$sim(q, d_i) = \frac{\sum_{k=1}^N d_{ik} q_k}{\sqrt{\sum_{k=1}^N d_{ik}^2} \times \sqrt{\sum_{k=1}^N q_k^2}} \quad \text{p. 338}$$

- $sim(q, d1) = 0.25$
- $sim(q, d2) = 0.50$
- les résultats sont retournés par ordre décroissant de similarité : d2, d1

B- Exécution de la requête de l'utilisateur

- **Exécution des requêtes vectorielles**

- en général, on utilise un poids compris entre 0 et 1, prenant en compte divers facteurs (e.g. $w = tf \times idf$)
- tous les documents comportant au moins un terme de la requête sont retournés
- le modèle vectoriel assure un repérage
 - basé sur la formule $mot1 \text{ OR } mot2 \text{ OR } \dots \text{ OR } motn$
 - impossibilité de spécifier des termes obligatoires
 - impossibilité de spécifier des termes interdits
- le modèle vectoriel assure un tri (similarité)
- permet des requêtes par similarité avec un document (QBE); le document constitue la requête

B- Exécution de la requête de l'utilisateur

- Autres modèles et variantes

- Booléen étendu: utilise *p-norm* pour similarité
- Vectoriel étendu: termes obligatoires/optionnels (+/-)
- Vectoriel généralisé: utilise la corrélation entre termes
- Modèles probabilistes: utilise le théorème de Bayes
- Ensembles fréquents, ensembles approximatifs
- Index sémantique latent: utilise factorisation SVD
Singular Value Decomposition
- Algorithmes génétiques: pour cooccurrences de termes
- Réseaux de neurones artificiels
 - MLP (« Multi Layer Perceptron »): induire des règles
 - SOM (« Self-Organized Map »): catégoriser documents, termes
 - AA (« Auto-Associatif »): catégoriser documents par patron

Après l'exécution de requêtes...

- Afficher les résultats

- Un ensemble de documents, triés selon la pertinence
- Présentation des résultats et hyperliens
- Visualiser les relations spatiales
 - L'ensemble résultant peut être présenté comme un ensemble de concepts ou de patrons, organisé en un espace multidimensionnel; ex: [KartOO](#) (moteur de recherche cartographique)

- Contrôle de la pertinence

- L'utilisateur examine et choisit les premiers M résultats
- Une nouvelle requête est émise basée sur la requête originale et cette information additionnelle

Plan du cours

- Types de manipulation de textes
- Introduction à la recherche d'information textuelle
- Manipulation de base des types de données texte en SQL
- Méthodes statistiques de manipulation de textes
- **Exemple d'implémentation : Oracle Text**

Oracle Text

- On a vu qu'utiliser SQL2 n'est pas recommandé pour des textes complexes !
- **Oracle Text** (avec SQL+, PL/SQL et JDeveloper)
- L'administrateur doit vous donner accès:
 - CTXAPP rôle
 - CTX PL/SQL packages
- Il y a un 'wizard' de création d'applications pour aller plus rapidement !

Oracle Text : 3 types d'applications

- Repérage d'information et requêtes plein texte :
 - approprié pour les textes non structurés de format usuel (.txt, .html, .xml, .doc, .pdf, ...);
 - index de type CONTEXT : *le seul décrit dans le cours!*
- Repérage d'information sur données semi-structurées :
 - approprié pour les textes stockés en plusieurs champs, certains de type classique (INTEGER, DATE ...), d'autres de type texte (VARCHAR2, CLOB, BLOB, BFILE) en format usuel;
 - index de type CTXCAT.
- Catégorisation, routage, filtrage de documents
 - sur textes de format .txt, .html et .xml seulement;
 - index de type CTXRULE.

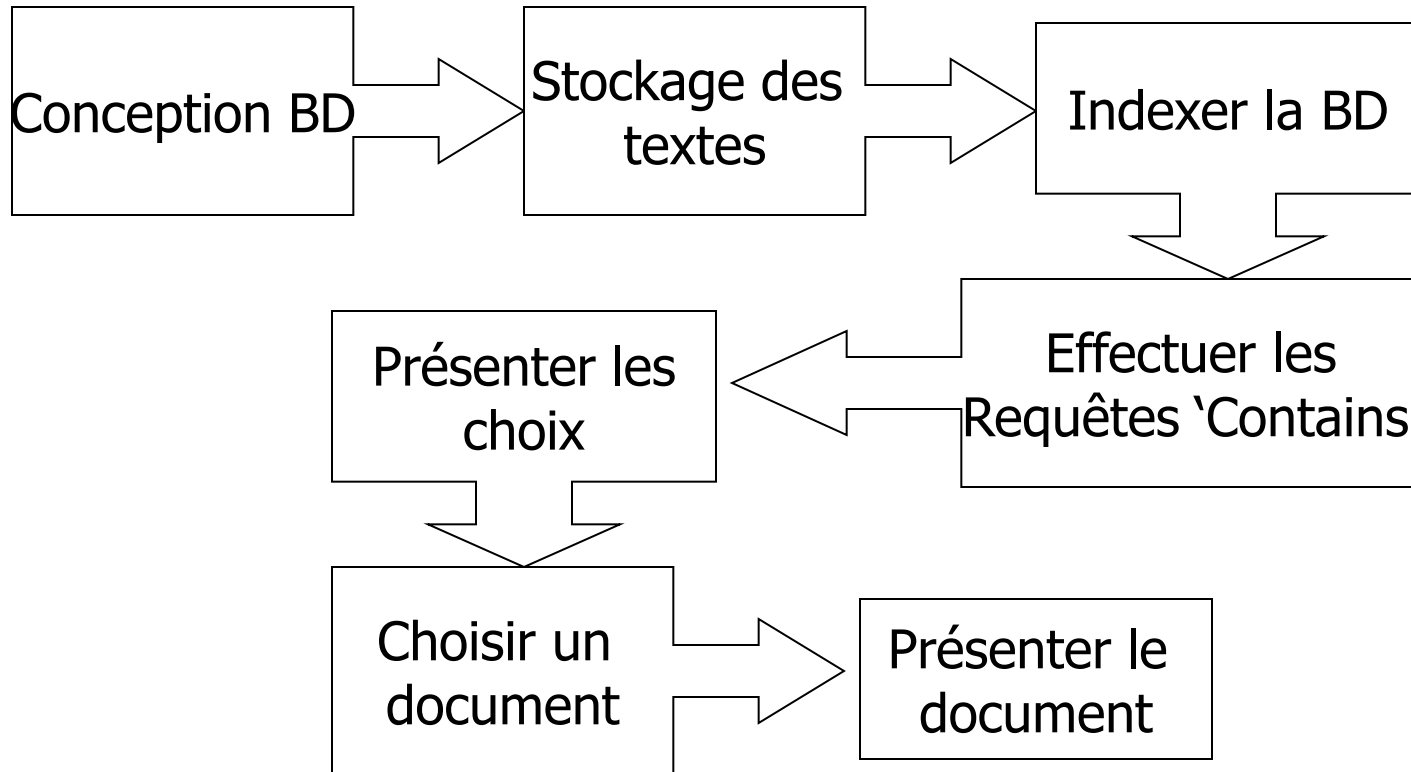
Oracle Text : stockage des textes

- Principales méthodes de stockage des textes dans une table pour indexation de type CONTEXT:
 - texte stocké dans une table (champ CLOB, BLOB, VARCHAR(2), CHAR, BFILE)
 - texte stocké sur le serveur, en dehors de la BD: chemin et nom du fichier stockés dans un champ de type caractère/texte
 - texte accessible sur le Web : URI stocké dans un champ de type caractère/texte
- Possibilité d'ajouter des colonnes pour méta données : identifiant, format, date, description, auteur ...

Oracle Text : chargement des textes

- Directement dans SQL+ si vous entrez le texte directement au clavier !
- Ou par SQL*Loader (à partir de fichiers sur disque)
- par la procédure PL/SQL
DBMS_LOB.LOADFROMFILE() pour des LOB

Étapes typiques pour la recherche dans un ou plusieurs texte(s)



Étapes 1, 2 et 3

- Création de table

```
CREATE TABLE mesdocuments (  
    identifiant          INTEGER PRIMARY KEY,  
    date_insertion      DATE,  
    format               CHAR(3) CHECK format IN  
        ('doc', 'pdf', 'txt', 'htm', 'xml'),  
    titre                VARCHAR2(100),  
    texte                CLOB)
```

- Insertion des données dans la table
- Création de l'index

```
CREATE INDEX indexdocs ON mesdocuments(texte)  
    INDEXTYPE IS CTXSYS.CONTEXT
```

Oracle Text : indexation par CONTEXT

- NB : l'index est statique

- N'évolue pas automatiquement lorsque l'on modifie la colonne texte
- Il faut réindexer la table lorsque l'on modifie des textes
- Synchronisation en utilisant 2 Mo de mémoire par :

```
begin  
    ctx_ddl.sync_index('indexdocs', '2M');  
end ;
```

Oracle Text : indexation par CONTEXT

- Nombreux paramètres pour l'indexation
- Par défaut :
 - suppose que le texte est stocké dans une colonne texte
 - détecte automatiquement le type de texte
 - suppose que la langue est celle définie lors de la configuration du SGBD
 - utilise une liste prédéfinie de mots vides, non indexés, associée à la langue
 - stocké en convertissant en majuscules

Oracle Text : indexation par CONTEXT

- Quelques 'paramétrisations' possibles:
 - indexer des fichiers externes au SGBD
 - ne pas filtrer/convertir les documents (intéressant si format HTML et TXT)
 - utiliser une liste différente de mots vides
 - créer un index multilingue
 - indexer en tenant compte de la casse
- Possibilité de synchroniser l'index à intervalles réguliers
- Voir "Oracle Text Reference" pour plus de détails

Étape 4 : Requêtes sur CONTEXT

- Deux fonctions additionnelles:
 - CONTAINS(<colonne>, <requête>, <étiquette>) retourne un score entre 0 et 100
 - SCORE(<étiquette>) retourne un score calculé par CONTAINS

- Exemple 1 : requête sur un mot:

```
SELECT titre
```

```
FROM mesdocuments
```

```
WHERE CONTAINS (texte, 'informatique') > 0
```

Oracle Text : requêtes sur CONTEXT

- Calcul du score d'un mot pour un document donné:
 - tient compte du nombre total de documents dans la base
 - tient compte du nombre de documents contenant le mot
 - tient compte du nombre d'occurrences du mot dans le document
- Voir "Oracle Text Reference" pour plus de détails

Quelques fonctions sur les chaînes de caractères en SQL CONTAINS

SQL CONTAINS pour texte	Objectif
ABOUT	Recherche les textes sur un concept ex: 'about(dogs) not about(labradors)' , <i>heat</i> might return documents related to temperature, even though the term <i>temperature</i> is not part of the query.
ACCUMulate	Donne une note aux documents qui possède plus d'une occurrence d'un 'terme' ex: <i>dog ACCUM cat (2 termes 51:100%, 1 terme 1:50%)</i>
EQUIV (=)	Indique des mots équivalents dans une requête, ex: 'labradors=alsatians are big dogs'
Fuzzy	Augmente la portée en incluant des mots qui sont semblablement épelés, ex: 'fuzzy(government, 70, 6, n)'

Quelques fonctions sur les chaînes de caractères en SQL CONTAINS

SQL CONTAINS pour texte	Objectifs
Soundex (!) (voir p. 326)	Augmente la portée de la requête et regarde les similitudes phonétiques, ex: WHERE CONTAINS (COMMENT, '!SMYTHE') > 0 ;
near	Trouve des mots qui sont proches les uns des autres dans le texte, ex: 'near((dog, cat), 6)'
wildcard (%)	Convertit en majuscules une chaîne de caractères
Threshold (>)	Ne retourne que les documents qui vont avoir une note > limite, ex: '(lion > 30) and tiger'

Oracle Text : requêtes sur CONTEXT

- Exemple 1 : requête sur une expression avec tri des résultats et affichage du score:

```
SELECT SCORE(1), titre  
FROM mesdocuments  
WHERE CONTAINS(texte, 'bases de  
données', 1) > 0  
ORDER BY SCORE(1) DESC
```

Oracle Text : requêtes sur CONTEXT

- Exemple 3 : opérateurs booléens + ordre :
`near((liste de mots), taille fenêtre, ordre)`

```
SELECT SCORE(1), titre
FROM mesdocuments
WHERE CONTAINS(texte,
'NEAR((bases de données), objet), 5, TRUE)', 1) > 0
ORDER BY SCORE(1)
```

- bases de données doit précéder objet, dans une fenêtre de taille maximale 5
- `score(near((a,b),10))` = dépend de la fréquence et de la proximité des termes dans le NEAR

Oracle Text : requêtes sur CONTEXT

- Autres catégories d'opérateurs

- Manipulation de textes structurés (typiquement : XML)
 - haspath, inpath
- Utilisation d'un thésaurus (Oracle ou propre à l'utilisateur)
 - broader term, narrower term, preferred term, related term, top term ...
- Utilisation d'un thésaurus multilingue
 - translation term ...
- Requête thématique (base de connaissances d'Oracle, extensible)
 - about
- Lemmatisation
 - \$travail = travail | travaux
- Recherche par phrase, paragraphe et autres sections
 - within

Oracle Text : Écrire le contenu d'un CLOB dans un fichier texte à l'ÉTS

- 4 Étapes typiques:

- 1) Créer un répertoire physique sur le serveur Oracle (142.137.17.104) utilisez SSH pour ça
- 2) Donner le droit d'écriture sur les répertoires de votre compte
- 3) Créer un répertoire oracle (create directory ...)
- 4) Exécuter du code de style: (p. 324)

LOOP

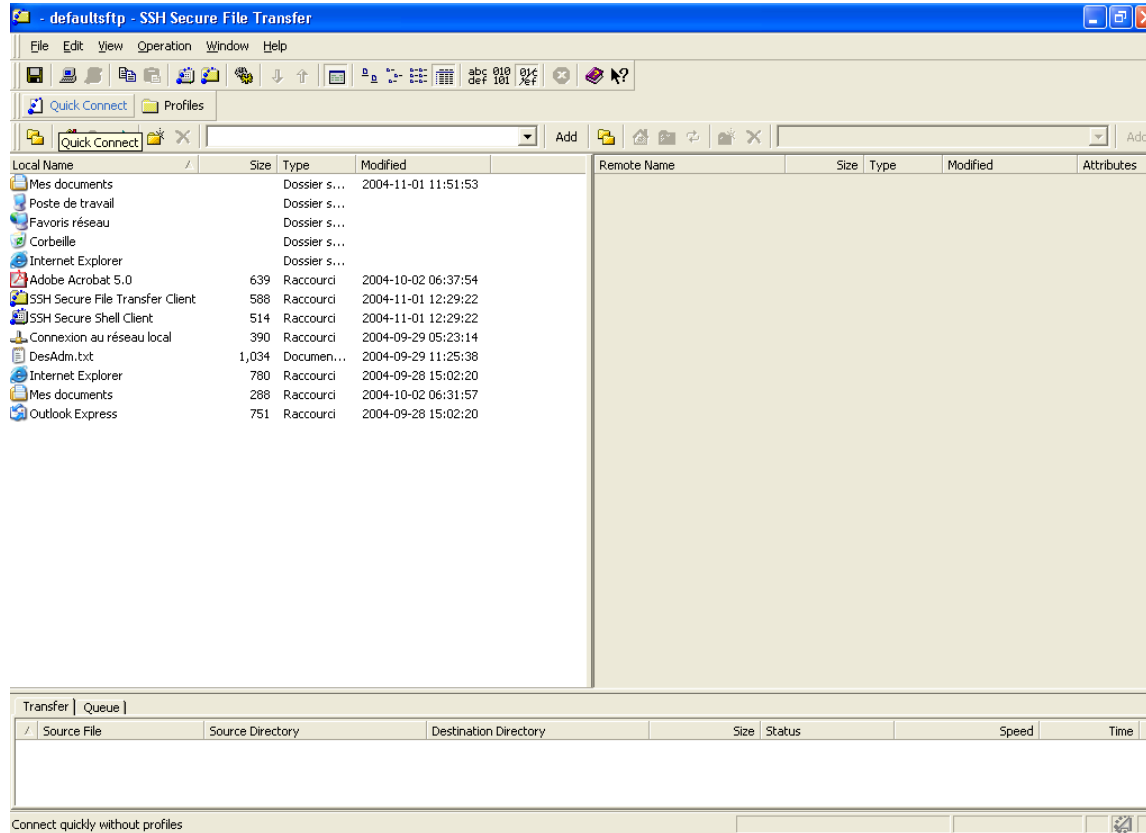
```
Dbms_Lob.Read (v_clob, v_amount, v_pos, v_buffer);
```

```
UTL_FILE.PUT(v_file, v_buffer);
```

```
v_pos := v_pos + v_amount;
```

END LOOP;

Étape 1: ssh



Cliquer sur quick connect.

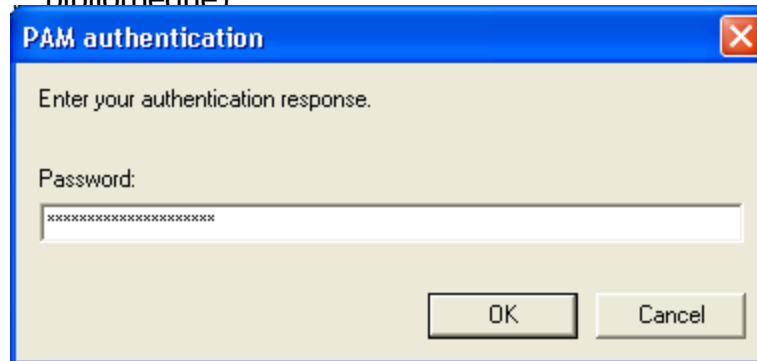
Adresse Ip 142.137.17.104

Compte étudiant de la forme AG99999 (9 représente un chiffre entre 0 et 9)

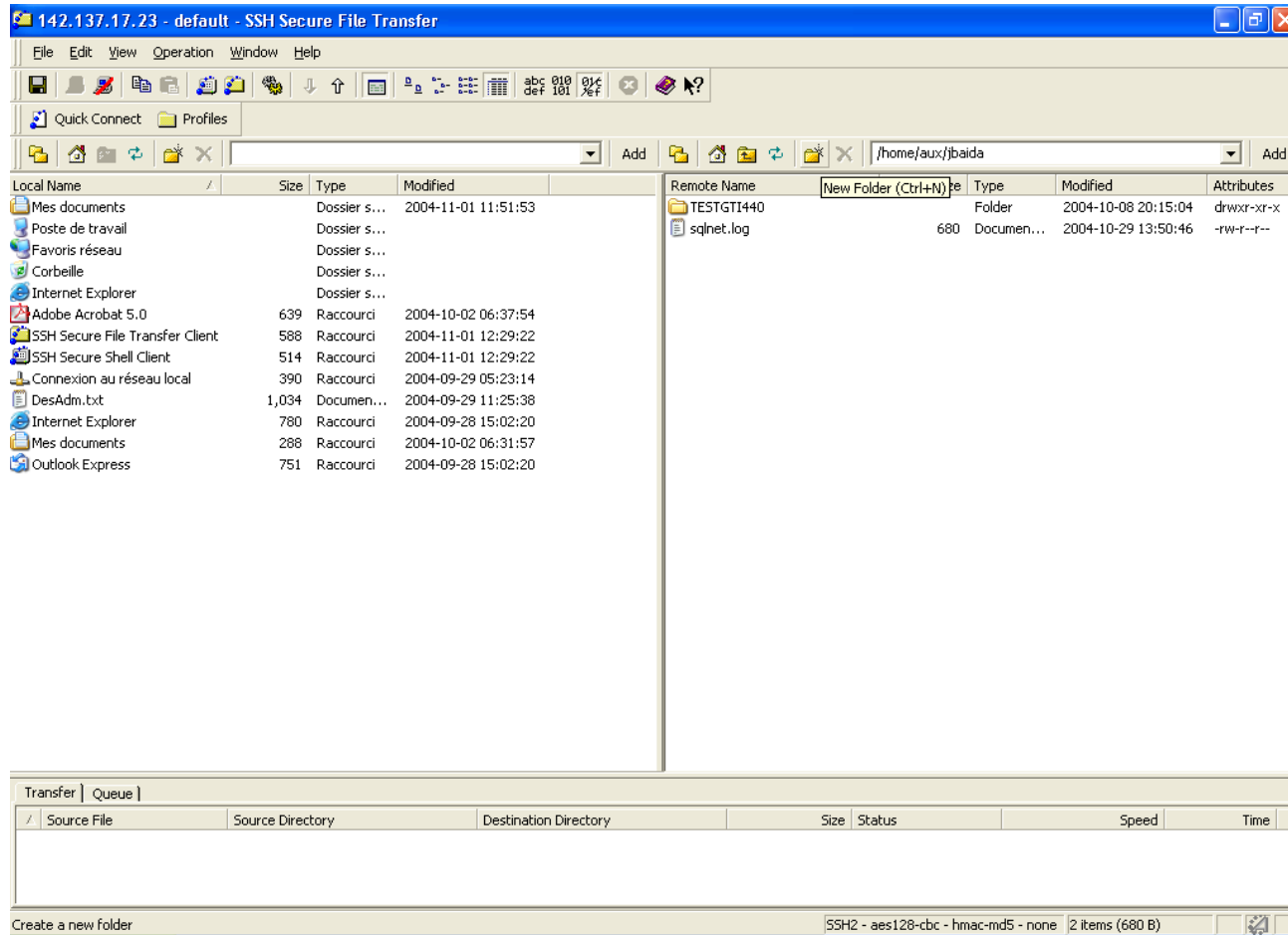
Étape 1: ssh (suite)



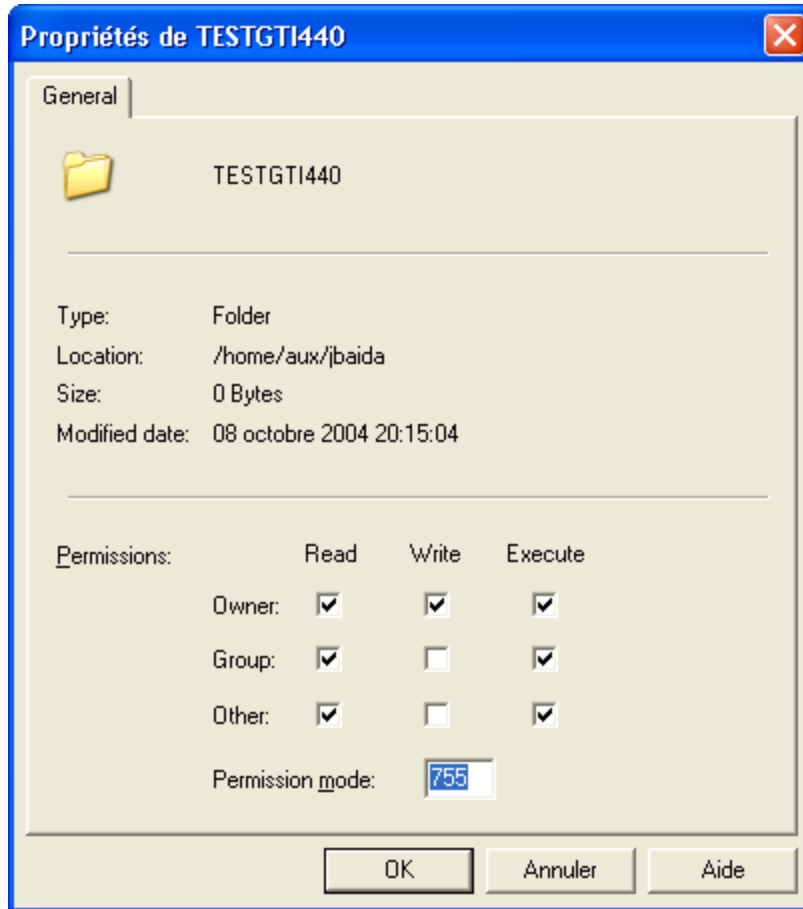
Entre le mot de passe (code de la bibliothèque)



Créer nouveau répertoire Étape 1: ssh (fin)



Étape 2: droits d'accès



Pointer sur le
répertoire, cliquer
sur le bouton droit
et choisir propriété

Nouveau avec 10g (Edwin Balthes)

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.ppt

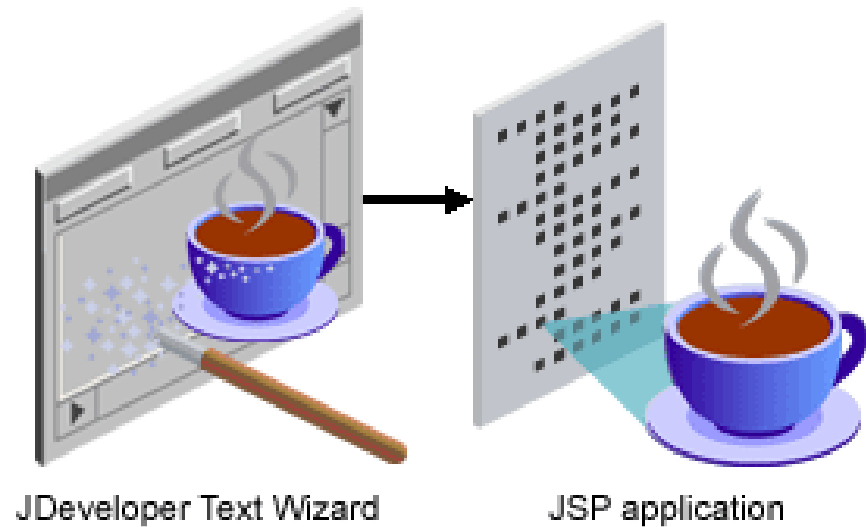
- Unicode 4.0
- Réécriture de requête (REWRITE)
- Nouvelle requête (near + accum)
- Relaxation Progressive de requêtes (REWRITE ordonnée)
- JDeveloper Text Wizard
- Pointage alternatifs



Nouveau Wizard dans JDeveloper

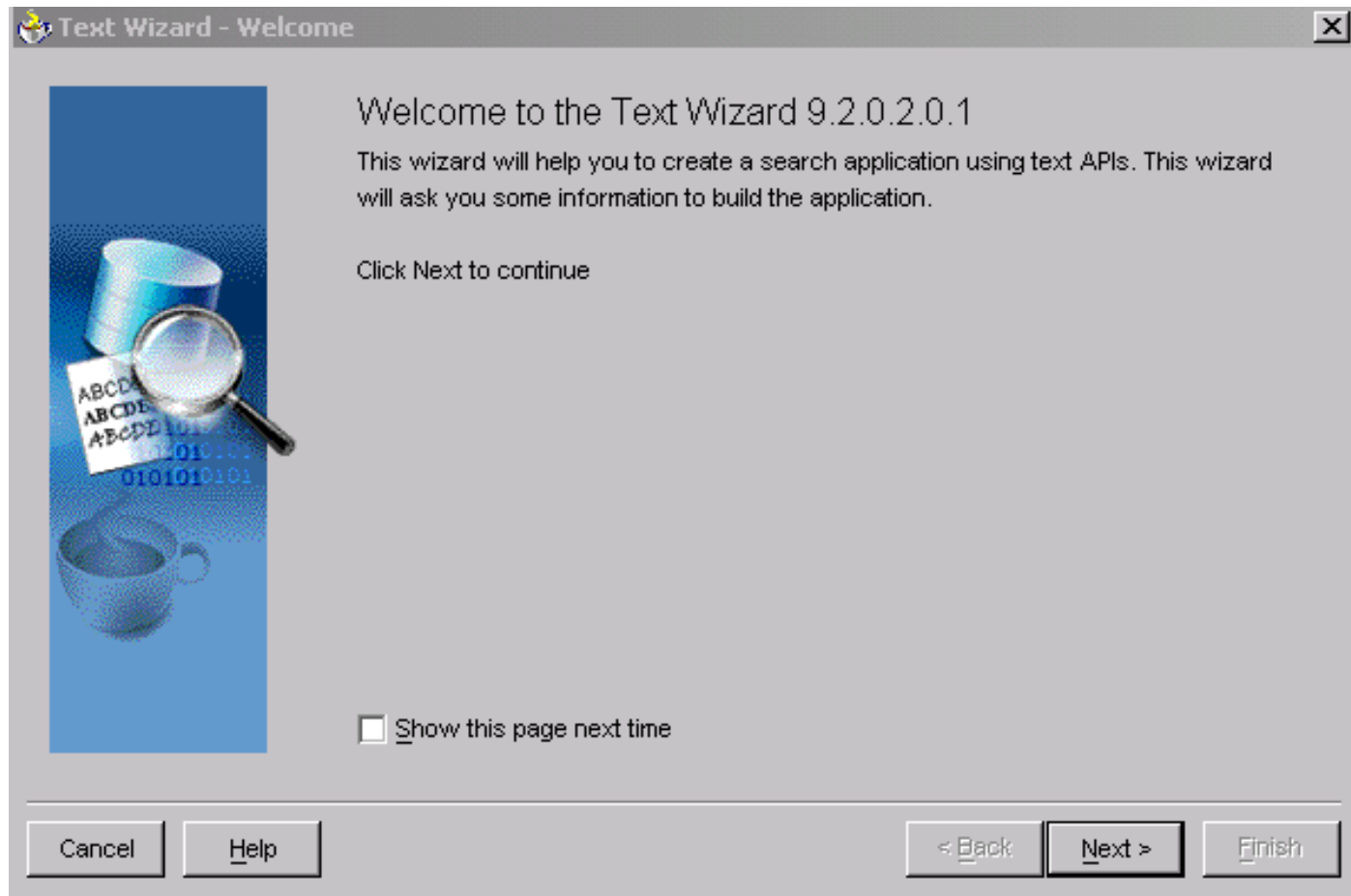
www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.ppt

- **Text Wizard**
- **Classification Wizard**
- **Catalog Wizard**



Démo du Wizard Texte


www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.opt



Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.opt

Text Wizard - Step 1 of 6: Application Name



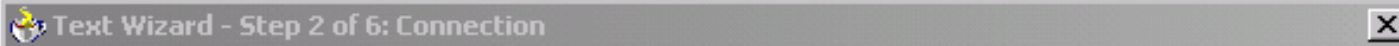
The location and the application name are initialized based on defaults for the current project. Change these values to create the application in another location or with another name. The application name will become the prefix of all files created. All the files created by this wizard will be put in the location you specify below.

Directory Name:


Application Name

Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.pdf

 Text Wizard - Step 2 of 6: Connection

Select the named connection which refers to the database that you want to deploy to.



Connection:


Username:

Driver:

URL:

Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.pdf

 Text Wizard - Step 3 of 6: Search Table and Column

Please pick one table you want to search on, then select one column which you want to create CONTEXT index and do search on, so only the searchable columns are displayed, they must be the following types: VARCHAR2, BLOB, CLOB, BFILE.




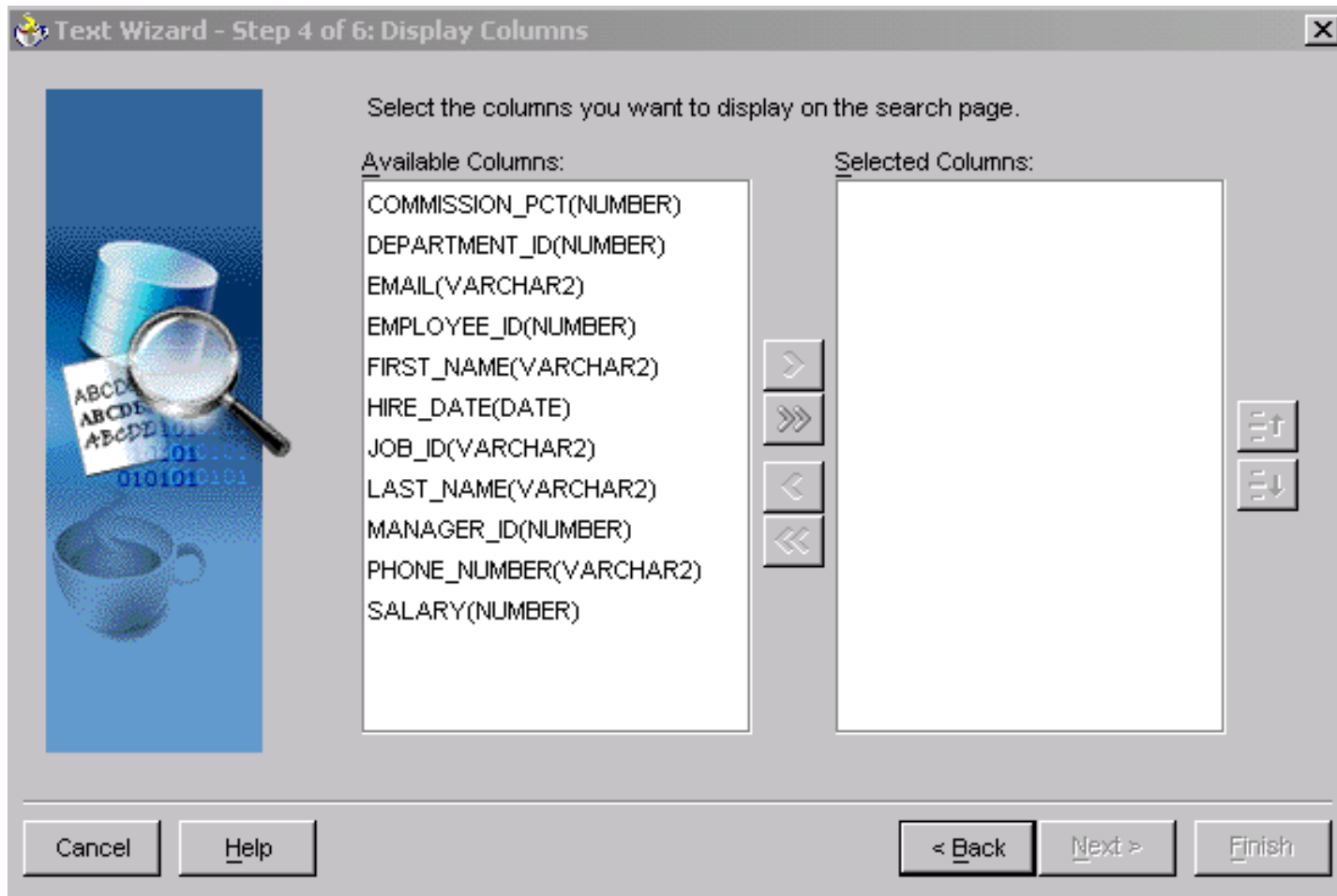
Table Name:
EMPLOYEES

Column Name:
EMAIL(VARCHAR2)

Cancel Help < Back Next > Finish

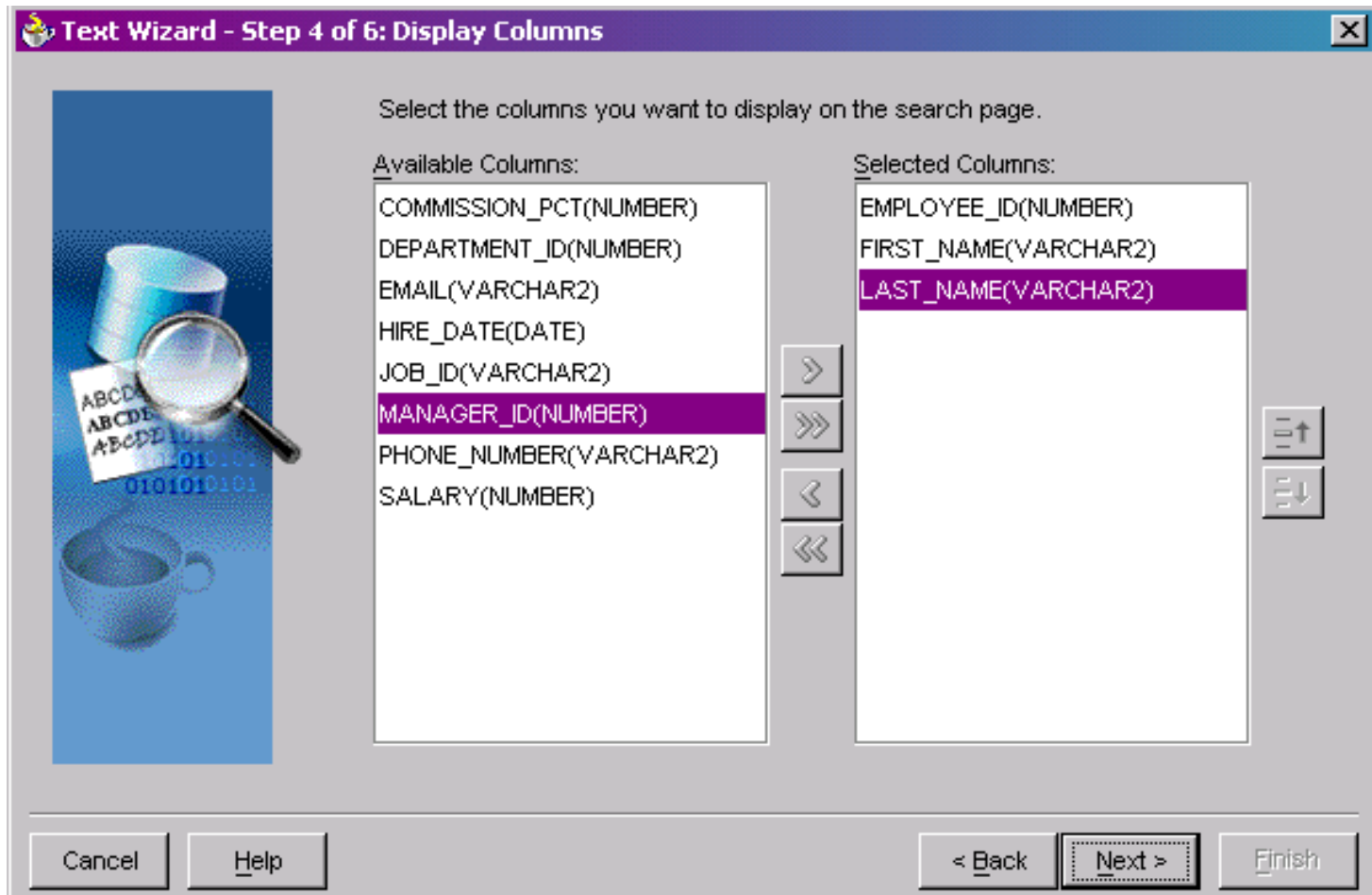
Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.pdf



Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.pdf



Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.pdf

Text Wizard - Step 5 of 6: Customize Search Page

Please enter the number of items you want to display per search result page, and select the Types of the Text Document Service you want to apply to your search application.

Number of items per page :

20

Text Document Services :

- ☒ Markup
- ☒ Highlight
- ☒ Theme
- ☒ Gist

Cancel Help < Back Next > Finish

Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_05.ppt

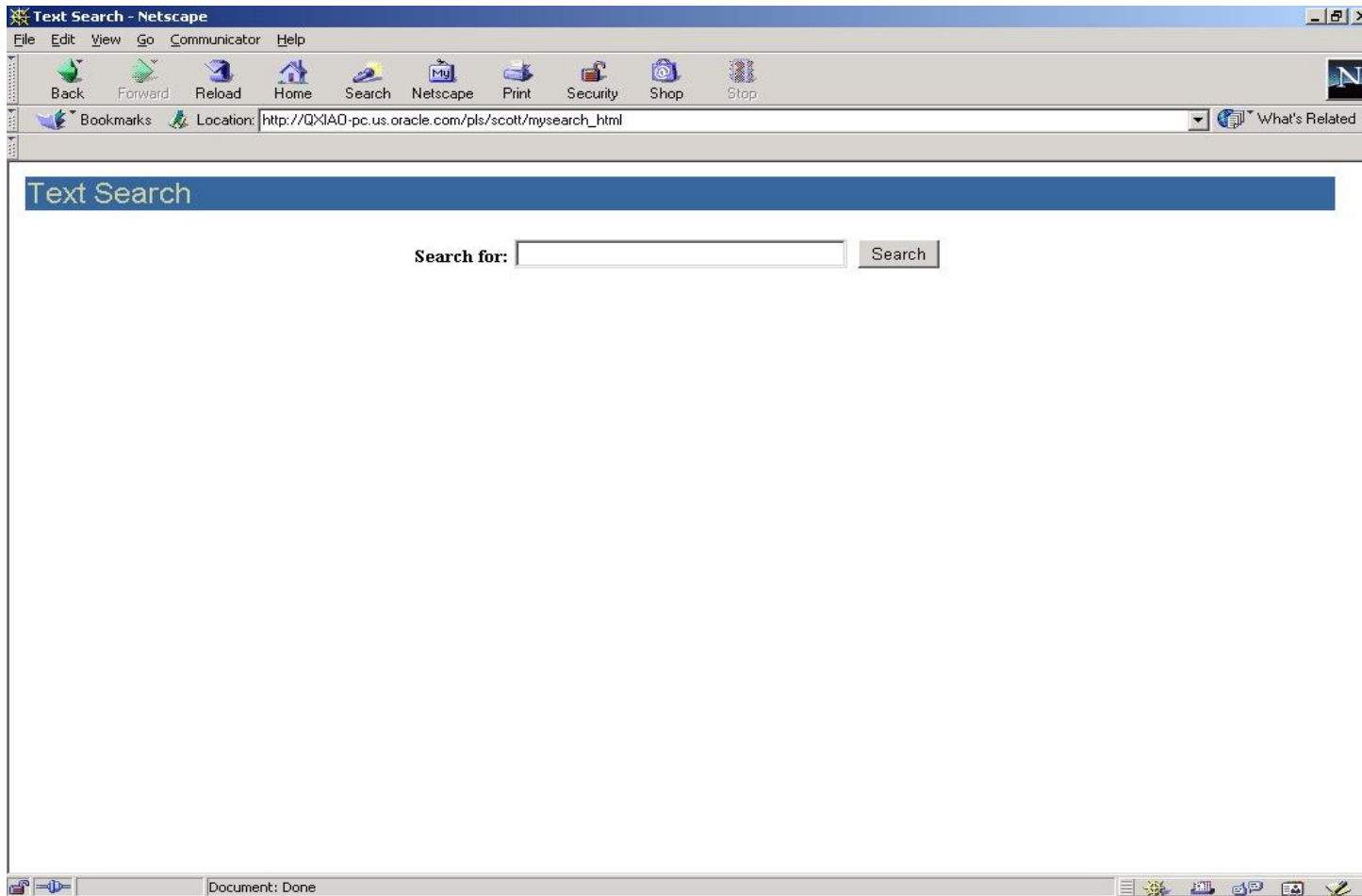
```
D:\oracle\ora9i\dev9i\dev\mywork\Workspace1\Project1\mysearch_createIndex.sql
1 begin
2   ctx_ddl.drop_preference('lexer_pref');
3 end;
4 /
5 begin
6   ctx_ddl.create_preference('lexer_pref','BASIC_LEXER');
7 end;
8 /
9 drop index EMPLOYEES_idx;
10 create index EMPLOYEES_idx on EMPLOYEES(LAST_NAME)
11 indextype is ctxsys.context parameters
12 ('FILTER ctxsys.null_filter LEXER lexer_pref SECTION GROUP CTXSYS.HTML_SECTION_GROUP');
13
```

```
D:\oracle\ora9i\dev9i\dev\mywork\Workspace1\Project1\mysearch_synchronizeIndex.sql
1 begin
2
3   ctx_ddl.sync_index('EMPLOYEES_idx', '2M');
4 end;
5 /
6
```

Line 1 Column 1 Insert Windows: CRLF

Démo du Wizard Texte

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.opt



Text – Recherche Simple

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.ppt

Search Oracle - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites History Print Copy Paste

Address http://benri.oracle.com/globalsearch/mysearch_ott.jsp?query=k-means&content=all Go Links

Google Search Web 18 blocked AutoFill Options

Basic Search [Advanced Search](#) [Register](#) [Login](#) [Help](#) [Feedback](#) [Submit URL](#) [My Bookshelf](#) [Knowledge Base](#) [Query Log](#) [Server Stats](#)

Search Oracle

All Web pages E-mail Files Online

Search for: k-means Search

Search results summary
(of top 100 hits)

Information sources:

- E-mails (2)
- Files Online (40)
- Web pages (58)

Categories:

- [oracle products > Oracle Development Tools > Oracle iLearning > Content](#)(22)
- [oracle products > Oracle Database > Oracle Database Certified](#)

Search returns about 343 matches

[Hide Summary](#)

[Oracle9i and Microsoft SQL Server 2000](#)
Confidential Page 1 07/01/02 Oracle9i Data Mining: Overview and Comparison with Microsoft SQL Server 2000 Oracle Data Mining Technologies Group Contents 1 O r a c l e 9 i Data Mining Strategy.....
Category: [oracle 10g knowledge base > oracle products > Oracle Applications > CRM > Oracle Marketing for Communications > Datamining](#)
http://datamining.us.oracle.com/DM_vs_SQL_Server_v9.pdf 95253 Bytes 2002-07-01
[Cached](#) [Highlight](#) [Theme](#) [Gist](#) [Rank](#) [This Document](#) [Links](#)

[Oracle9i Data Mining Administrator's Guide](#)
#Library Copyright Skip Headers Oracle9i Data Mining Administrator's Guide Release 2 (9.2) Part Number A95959-01 Go To Documentation Library Library Go To Product List Product Oracle is a registered trademark,
Category: [oracle 10g knowledge base > oracle products > Oracle Database > Oracle9iAS Migration Kit for ASP > Docs](#)
<http://dmt.us.oracle.com/docs/odm/920/adminODM920/html/toc.htm> 44677 Bytes 2002-03-06
[Cached](#) [Highlight](#) [Theme](#) [Gist](#) [Rank](#) [This Document](#) [Links](#)

<http://st-doc.us.oracle.com/8.0/815/server.815/a67790.pdf>
Oracle8i Reference Release 8.1.5 February, 1999 Part No. A67790-01 Oracle 8i Reference, Release 8.1.5 Part No. A67790-01 Copyright 1996, 1999, Oracle Corporation. All rights reserved. Primary Author:
Category: [oracle 10g knowledge base > oracle products > Oracle Database > Oracle Database Certified Configuration > DCC](#)

Done Local intranet

Text – Recherche Avancée

www.dbag.org/pub/docs/sig/intermedia/2003-11/DOAG2003-TextNewFeatures_DE.ppt


Oracle Text Advanced Search - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search History Print Mail News

Address http://benri.oracle.com/globalsearch/adsearch_otn.jsp Go Links

Google Search Web 18 blocked AutoFill Options

 [Basic Search](#) [Advanced Search](#) [Register](#) [Login](#) [Help](#) [Feedback](#) [Submit URL](#) [My Bookshelf](#) [Knowledge Base](#) [Query Log](#) [Server Stats](#)

Oracle Text Advanced Search

Search for:

Result Number	Number of hit list displayed per page	<input type="text" value="10 results"/>
File Format	<input type="text" value="Only"/> return results of the file format	<input type="text" value="any format"/>
Date	Return web pages updated in the	<input type="text" value="anytime"/>
Sort By	<input type="text" value="Descending"/> return results sorted by	<input type="text" value="score"/>
Occurrences	Return results where my terms occur	<input type="text" value="anywhere in the page"/>
Domain	<input type="text" value="Only"/> return results from the site or domain	<input type="text" value="e.g. technet.oracle.com"/>

Done Local intranet

Oracle Recherche Ultra

www.doag.org/pub/docs/sig/intermedia/2003-11/DWG2003_TextNewFeatures_DE.pdf

- Out-of-the-Box search engine
 - Fait avec Oracle Text
- Pour vos intranet/extranet
 - Web, Databases, Files, Mail Servers, Repositories
- Développé avec interface Web et les API Java pour l'interface utilisateur



Ultra Search

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.ppt



[\[Advanced Search\]](#) [\[Help\]](#) [\[Submit URL\]](#)

Search For

Submit

[Jeff McMahon Resigns as Enron President, COO](#)

JEFF McMAHON RESIGNS AS ENRON PRESIDENT, COO FOR IMMEDIATE RELEASE:

Friday, April 19, 2002 HOUSTON ; Enron Corp.

Score: **77** Author: Stefan Buchta <Stefan.Buchta@oracle.com> Last modified: 2002-05-16
02:04:45.0 Page size: 2280 From: Email Server

[Enron Stock - Buy One Now As A Collectible](#)

Enron Corp is quite possible to be the largest bankruptcy in the history of the U.S. This is a sure to be collectible.

Score: **71** Author: Stefan Buchta <Stefan.Buchta@oracle.com> Last modified: 2002-05-16
02:05:25.0 Page size: 832 From: Email Server

<http://files.oraclecorp.com/content/AllPublic/Users/Users-S/Stefan.Buchta-Public/enron.pdf>

Enron Annual Report 2000 Enron Annual Report 2000 Enron manages efficient, flexible networks to reliably deliver

Score: **33** Last modified: 2002-05-16 22:10:08.0 Page size: 792732 From: Files Online Public



Université du Québec

École de technologie supérieure

Département de génie logiciel et des TI

Ultra Search Adv. Search

www.doag.org/pub/docs/sig/intermedia/2003-11/DOAG2003_TextNewFeatures_DE.ppt

Welcome to the **ORACLE** Intranet Search Engine



[Basic Search](#) [Submit URL](#) [Feedback](#) [Help](#)

Search for

Performance

Search

Narrowed by

Title

Oracle9i Server

And

Author

Buchta

And

Description

And

Subject

Add more attributes

From

☒ Email Archives ☐ Local Files ☐ Mixed Sources ☒ Server Technologies
☐ arunagr ☐ dictgrp ☐ portaldatgrp ☐ sarahGRP

Language

AMERICAN



Université du Québec

École de technologie supérieure

Département de génie logiciel et des TI

Oracle Text : présentation des résultats

- Suite à une requête
 - Possibilité de conversion du format d'origine en texte, HTML et XML
 - Possibilité de présenter le document avec les termes de la requête surlignés
- Indépendamment des requêtes
 - Possibilité de déterminer automatiquement le/les thèmes du document
 - Possibilité de déterminer automatiquement un passage (*gist*) représentant le mieux le document

Oracle Text : présentation des résultats

- cf. "Oracle Text Reference", "Oracle Text Application Developer's Guide"
 - <http://www.lc.leidenuniv.nl/awcourse/oracle/text.920/a96517/toc.htm>

Travaux personnels et labo

- Faire les exercices du Chapitre 10
- Commencez à lire la documentation Oracle Text proposée pour ce cours
- <http://www.lc.leidenuniv.nl/awcourse/oracle/text.920/a96517/toc.htm>
- Continuez votre premier labo
 - SL3
 - Commencer à planifier l'intégration avec le labo 2 !