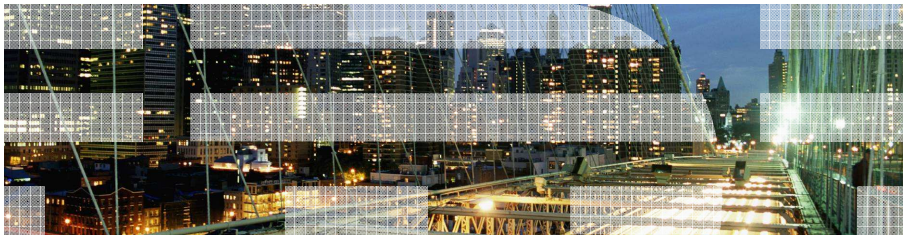
Nouveautés DB2 for i 7.1

COMMON Romandie - 03 mai 2011

C. GRIERE cgriere@fr.ibm.com



Information Management

© 2011 IBM Corporation

Plan de la session

- Développement Application
- Performance
- Résilience et disponibilité
- Interface et sécurité

© 2011 IBM Corporation

Développement Application

© 2011 IBM Corporation

Accès aux Result Set

- Accès aux Result Set en :
  - RPG ILE
  - Cobol ILE
  - Routines SQL :
    - Procédures
    - Fonctions
    - Déclencheurs
- 4 nouvelles instructions :
  - ASSOCIATE LOCATORS
  - ALLOCATE CURSOR
  - Les instructions DESCRIBE PROCEDURE et DESCRIBE CURSOR peuvent être utilisées pour déterminer dynamiquement le nombre et le contenu des Result Set

© 2011 IBM Corporation

Lecture Result Set en RPG ILE - Exemple

IBM Power Systems

```
...
D RSL1          S                   SQLTYPE(RESULT_SET_LOCATOR)
D RSL2          S                   SQLTYPE(RESULT_SET_LOCATOR)
C/EXEC SQL CALL proc (:parm1, :parm2, ...)
C/END-EXEC
...
C/EXEC SQL ASSOCIATE LOCATORS (:rsl1,:rsl2) WITH PROCEDURE proc
C/END-EXEC
C/EXEC SQL ALLOCATE c1 CURSOR FOR RESULT SET :rsl1
C/END-EXEC
C/EXEC SQL ALLOCATE c2 CURSOR FOR RESULT SET :rsl2
C/END-EXEC
...
C/EXEC SQL FETCH c1 INTO A1, A2 ;
C/END-EXEC
...
C/EXEC SQL FETCH c2 INTO B1, B2 ;
C/END-EXEC
```


© 2011 IBM Corporation

Lecture Result Set dans procédure SQL - Exemple

IBM Power Systems

```
...

DECLARE rsl1 RESULT_SET_LOCATOR VARYING ;

CALL Proc(nocli) ;

ASSOCIATE LOCATORS (rsl1) WITH PROCEDURE proc ;

ALLOCATE c1 CURSOR FOR RESULT SET rsl1 ;

myloop: LOOP
   FETCH c1 INTO nomcli, solde ;
   IF row_not_found=1 THEN
    LEAVE fetch_loop ;
   END IF;
   ...
END LOOP ;
CLOSE c1 ;
```


© 2011 IBM Corporation

Clause FIELDPROC – Codage & Décodage

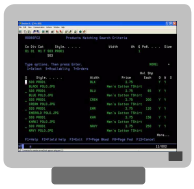
IBM Power Systems

NO_CARTE

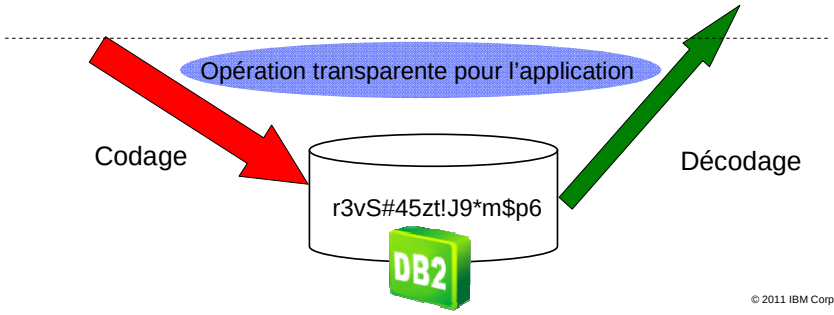


1111 2222 3333 4444

Accès autorisé



1111 2222 3333 4444



© 2011 IBM Corporation

FIELDPROC - Syntaxe

IBM Power Systems

- Créer une table CLIENT dans laquelle la colonne **NO_CARTE** est associée à un programme de codage/décodage :

```
CREATE TABLE  CLIENT (NOM VARCHAR(50), ADRESSE
VARCHAR(100),
        NO_CARTE CHAR(16) FIELDPROC DATA.CD1 )
```

- Altérer la colonne **SALAIRE** de la table EMPLOYE pour l'associer à un programme de codage/décodage :

```
ALTER TABLE  EMPLOYE
        ALTER COLUMN SALAIRE SET  FIELDPROC DATA.CD2
```


© 2011 IBM Corporation

IBM Power Systems

FIELDPROC - Implémentation

- Totale liberté pour modifier de façon **réversible** et **déterministe** le contenu de la colonne via un programme
  - **Cryptage** (solutions tierce partie : Linoma Software [Crypto Complete], Patrick Townsend)
  - Compression
  - ..
- Opérationnel pour toutes les interfaces :
  - SQL
  - IO natives
  - CL (DSPPFM, CPYF, RGZPFM)
- Le programme doit être de type *PGM ILE et ne pas contenir de SQL
- Il est appelé dans 3 cas :
  - Création/modification de la colonne
  - Ecriture
  - Lecture
- L'affectation d'un FIELDPROC se fait via SQL (create table/ alter table)
- Les index sur les colonnes avec FIELDPROC sont encodés
- Attention : un CHGPF perd les FIELDPROC

© 2011 IBM Corporation

IBM Power Systems

Précompilateurs et compilateurs

- Nouveautés précompilateur et compilateur ILE
  - Paramètre de comportement en cas de verrouillage d'enregistrement
  - Paramètre clé d'encryptage du source (DBGGENKEY)
- Nouveautés compilateur ILE Cobol
  - Nouveau type : COMP-5 (COMP-4 avec valeur maxi de la représentation interne)
- Nouveautés compilateur ILE RPG
  - Support mot-clé ALIAS au niveau des DS externes

```
CREATE TABLE CLIENTS (NUMERO_DU_CLIENT   INTEGER,
                      NOM_DU_CLIENT      CHAR(30),
                      ADRESSE_DU_CLIENT  VARCHAR(80))
```

```
d DS1          E DS      EXTNAME(CLIENTS) QUALIFIED ALIAS
* Les zones de la DS seront :
*   NUMERO_DU_CLIENT
*   NOM_DU_CLIENT
*   ADRESSE_DU_CLIENT
```


© 2011 IBM Corporation

IBM Power Systems

Variables globales SQL

IBM

- Permet un partage simple de valeur entre instructions/objets (déclencheurs, vues, ….) SQL au sein d'un travail/connexion
  - Exemple 1 : Passage de valeur :

```
CREATE or REPLACE VARIABLE departement_var INTEGER DEFAULT
(SELECT departement FROM employe WHERE no_employe = :user) ;

SELECT departement_var FROM qsys2.qsqptabl ;
SELECT etage  FROM location WHERE dept = departement_var ;

CREATE VIEW employe_vue AS (
 SELECT prenom, nom, telephone FROM employe WHERE departement =
departement_var) ;
 …
VALUES 43 INTO departement_var ;
```

Etat SQL : 21000 Code fournisseur : -811 Message : [SQL0811] Result of SELECT more than one row.

© 2011 IBM Corporation

IBM Power Systems

Variables globales SQL

IBM

  - Exemple 2 : Déclencheur optionnel :

```
CREATE VARIABLE mode_batch CHAR(1) ;

 CREATE TRIGGER depenses_dec AFTER INSERT ON depenses
    REFERENCING NEW AS n  FOR EACH ROW
 WHEN (mode_batch='N')
  BEGIN
     DECLARE nom_var CHAR(30) ;
     SET nom_var = (SELECT nom FROM employe WHERE no=n.no) ;

     IF n.total < 10000 THEN
       INSERT INTO audit_voyages
         VALUES(n.no, nom_var, n.departement, n.total, n.date_fin) ;
     ELSE
       SIGNAL SQLSTATE '38001' SET MESSAGE_TEXT='Maximum
dépassé' ;
     END IF ;
  END
 …
VALUES 'Y' INTO mode_batch ;
```


© 2011 IBM Corporation

IBM Power Systems

Support des tableaux SQL

- Permet d'échanger des tableaux de données
- Limité au type de donnée simple et à une dimension
- Le type ARRAY peut être utilisé comme paramètres de routine SQL ou de variable locale
- Pas d'assignation entre un tableau "host language" et un tableau SQL
- Interfaces qui supportent le type tableau SQL comme paramètre
  - JDBC
  - Routines SQL
- Exemples :

```
CREATE TYPE caractere_tableau AS CHAR(3)  ARRAY[10] ;
CREATE TYPE entier_tableau      AS INTEGER ARRAY[5] ;
```


© 2011 IBM Corporation

IBM Power Systems

Support des tableaux en SQL- Exemple

```
-- Retourne le tableau des quantités associées au tableau des articles
CREATE OR REPLACE PROCEDURE liste_articles
    (IN articles  caractere_tableau, OUT quantites  entier_tableau)
DYNAMIC RESULT SETS 1
LANGUAGE SQL
BEGIN
   DECLARE c1 CURSOR FOR SELECT t1.num as "NO", quantite, type
   FROM articles_table a, UNNEST(articles) AS t1(num)  WHERE t1.num = a.num ;
   IF CARDINALITY( articles ) > 5 THEN
      SIGNAL SQLSTATE '38003'
      SET MESSAGE_TEXT='Trop d"articles' ;
  END IF ;
  SET quantites = (SELECT ARRAY_AGG(quantite)
                   FROM articles_table a, UNNEST(articles) AS t2(num)
                   WHERE t2.num = a.num) ;
  OPEN   c1 ;
END ;
 ...
SET mes_articles = ARRAY['W12','S55','M22'] ;
CALL liste_articles (mes_articles, mes_quantites) ;
```

Sortie

NO	QUANTITE	TYPE
W12	25	KSR
S55	124	KSR
M22	125	MNG

Tableau quantités
[1] = 25
[2] = 124
[3] = 125

© 2011 IBM Corporation

Procédures stockées

IBM Power Systems

- Expressions dans les appels :

```
CALL maproc ( 1, UPPER(nom_societe), taux_remise*100 )
```


© 2011 IBM Corporation

Support XML dans DB2 for i

IBM Power Systems

- Remplace le produit DB2 XML Extender
  - Stockage de fichier XML dans une colonne de type **XML**
  - Déstockage d'une colonne XML dans un fichier
  - Décomposition de documents XML dans des colonnes DB2
  - Génération de documents XML à partir de colonnes via des fonctions SQL de publication
- IBM OmniFind Text Search Server fournit des recherches rapides dans les documents XML stockés
  - Produit gratuit (5733-OMF) disponible à partir de la 6.1
  - Recherche sur un élément ou un attribut XML : /book/title[. contains("winning")]
  - Interface XQuery pas fournie

© 2011 IBM Corporation

IBM Power Systems

Type de donnée XML

IBM

- Nouveau type de donnée : XML
- Taille : jusque 2 Go
- CCSID par défaut : UTF-8
- Utilisation : colonne, paramètre ou variable hôte

  – Colonne :

```
CREATE TABLE reservations (resID INTEGER GENERATED
ALWAYS AS IDENTITY, ReservationDoc XML )
```

  – Variable hôte :

  Exemple de déclaration en ILE RPG :

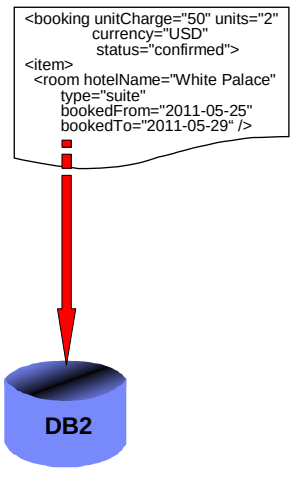
```
D MYXMLCLOB      S SQLTYPE(XML_CLOB:3000)
D MYXMLDBCLOB    S SQLTYPE(XML_DBCLOB:400)
D MYXMLBLOB      S SQLTYPE(XML_BLOB:780)
```


© 2011 IBM Corporation

IBM Power Systems

Type de donnée XML

IBM



- Stockage avec validation basée sur un schéma XML:

```
INSERT INTO reservations (ReservationDoc)
 VALUES(XMLVALIDATE(
        XMLPARSE(DOCUMENT
GET_XML_FILE('/dir/resa1s.xml'))
        ACCORDING TO XMLSCHEMA
        ID macol.resschema) )
```

  – Le parsing est implicite ou explicite
  – La validation est optionnelle

- Déstockage
  – Les variables de type 'XML File Reference' permettent une exportation simple dans l'IFS.

```
D MY_XMLFILE   S      SQLTYPE(XML_CLOB_FILE)
/free
   MY_XMLFILE_NAME= '/resa1d.xml';
   MY_XMLFILE_NL = 10;
   MY_XMLFILE_FO = SQFCRT;
   exec sql SELECT ReservationDoc INTO
:MY_XMLFILE FROM reservations WHERE resID=1;
/end-free
```


© 2011 IBM Corporation

IBM Power Systems

Utilitaires XML 7.1

IBM

Fonction intégrées :	
GET_XML_FILE	Retourne le contenu d'un document IFS ou un membre source sous forme d'un LOB locator
XMLVALIDATE	Valide un type XML avec un schéma XML; Retourne un type XML.
XMLPARSE	Parse des données caractère ou LOB ; Retourne un type XML.
XMLSERIALIZE	Convertit un type XML ; Retourne une chaine LOB dans le type demandé
XSLTRANSFORM	Convertit des données XML en un autre XML, HTML en utilisant le processeur XSLT
Procédures stockées dans SYSPROC :	
XSR_REGISTER	Ajoute un schéma XML dans le repository du schéma XML (XSR) pour validation et décomposition
XSR_ADDSCHEMADOC	Fusionne un schéma XML avec un schéma XML
XSR_COMPLETE	Termine l'ajout d'un schéma XML dans le repository XSR
XSR_REMOVE	Enlève un schéma XML
XDBDECOMPXML	Décompose un document XML dans des tables DB2

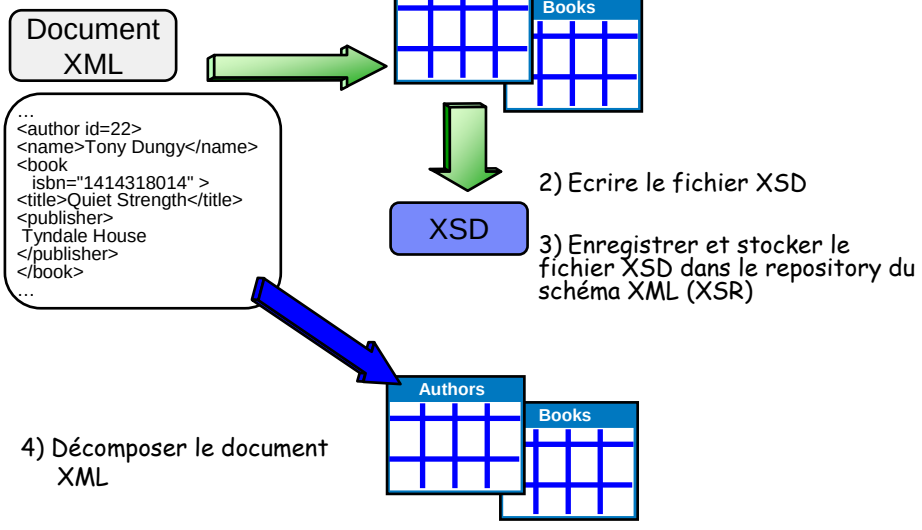
© 2011 IBM Corporation

IBM Power Systems

Décomposition de document XML

IBM

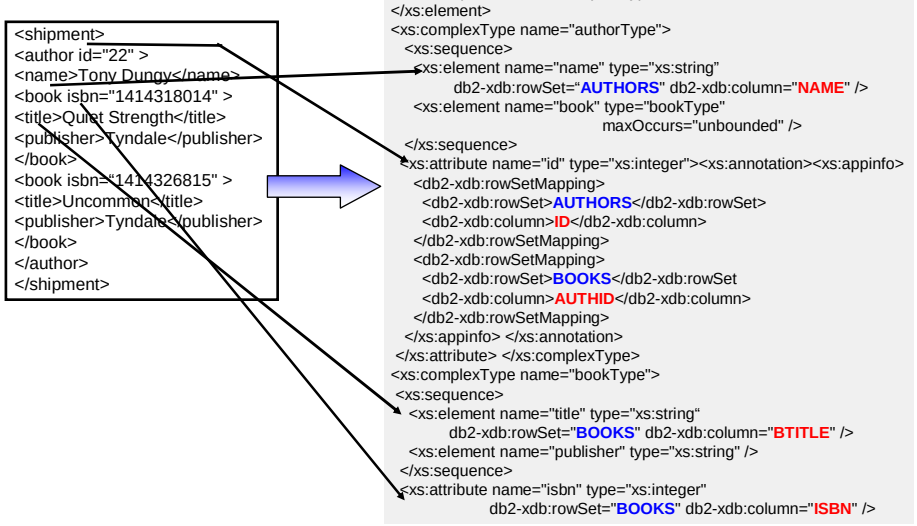
1) Faire correspondre les éléments /attributs XML et les colonnes / tables DB2



© 2011 IBM Corporation

Exemple de décomposition

IBM Power Systems



© 2011 IBM Corporation

Exemple de décomposition

IBM Power Systems

- Etapes de la décomposition :
  1. Création du fichier XSD

  2. Enregistrement du fichier XSD dans le repository du schéma XML

```
CALL SYSPROC.XSR_REGISTER ('MYLIB', 'BOOKSCHEM', null,
                           GET_XML_FILE('/dir/authbooks.xsd'), null)
CALL SYSPROC.XSR_COMPLETE('MYLIB','BOOKSCHEM',null,1)
```

  3. Décomposition du document XML

```
CALL SYSPROC.XDBDECOMPXML
  ('MYLIB','BOOKSCHEM', GET_XML_FILE('/mydir/ship1.xml'),null)
```

Instructions générées par la décomposition

```
INSERT INTO authors VALUES(22, 'Tony Dungy')
INSERT INTO books
   VALUES(22, 'Quiet Strength', 1414318014),
             (22, 'Uncommon', 1414326815)
```


© 2011 IBM Corporation

Publication – XMLELEMENT & XMLATTRIBUTE

IBM Power Systems

- Générer un document XML pour des employés célébrant leur 25ième anniversaire :

```
SELECT
   XMLSERIALIZE(
  XMLELEMENT(NAME "Employe", XMLATTRIBUTES(e.noemp as "no"),
  XMLELEMENT(NAME "Nom", e.prenom ||' '|| e.nom),
   XMLELEMENT (NAME "Telephone",e.telephone),
   XMLELEMENT (NAME "Departement", d.nomdpt))  AS CLOB(100) ) as
  "XMLResult"

FROM employe e, departement d
WHERE e.dpt = d.dpt AND
  YEAR(CURRENT DATE) –
  YEAR(date_embauche) = 25
```

Valeur de XMLResult :

```
<Employe no="000010">
  <Nom>Jean MARTIN</Nom>
  <Telephone>0601020304</Telephone>
  <Departement>Paie</Departement>
</Employe>
----------------------------
<Employe no="000050">
  <Nom>Jacques DUPOND</Nom>
  <Telephone>0611223344</Telephone>
  <Departement>Entretien</Departement>
</Employe>
```

23

Exemple de publication XML - XMLFOREST

IBM Power Systems

- Générer le même document XML en utilisant la fonction XMLFOREST pour simplifier la requête :

```
SELECT XMLSERIALIZE(
   XMLELEMENT(NAME "Employe", XMLATTRIBUTES(e.empno as "no"),
   XMLFOREST(e.firstname ||' '|| e.lastname as "Nom",
                    e.telephone as "Telephone",
                   d.nomdpt as "Departement") )  AS CLOB(100) ) as "XMLResult"

FROM employee e, department d
WHERE e.dpt = d.dpt AND
  YEAR(CURRENT DATE) –
     YEAR(date_embauche) = 25
```

Sortie de XMLResult :

```
<Employe no="000010">
  <Nom>Jean MARTIN</Nom>
  <Telephone>0601020304</Telephone>
  <Departement>Paie</Departement>
</Employe>
----------------------------
<Employe no="000050">
  <Nom>Jacques DUPOND</Nom>
  <Telephone>0611223344</Telephone>
  <Departement>Entretien</Departement>
</Employe>
```

24

IBM Power Systems

Exemple de publication XML - XMLAGG

IBM

- Générer le comptage et le document XML des pièces pour un type de pièces

```
SELECT COUNT(*) AS PartCnt,
XMLSERIALIZE(
XMLELEMENT(NAME "Parts", XMLATTRIBUTES(parttype AS "type"),
                XMLAGG(XMLELEMENT(NAME "pid", partid)  ORDER BY  partid))
 AS CLOB(130)) AS PartList
FROM parts WHERE parttype IN ('C01', 'E21')
GROUP BY parttype
```

PartCnt	PartList
2	<Parts type="C01"><pid>000130</pid><pid>200140</pid> </Parts>
3	<Parts type="E21"><pid>000320</pid><pid>100330</pid> <pid>200340</pid></Parts>

© 2011 IBM Corporation

IBM Power Systems

Exemple de publication XML - XMLGROUP

IBM

- Générer le comptage et une valeur XML des pièces d'un type donné :

```
SELECT COUNT(*) AS PartCnt,
 XMLGROUP(parttype AS "type", partid AS "pid"
                ORDER BY parttype, partid
                OPTION ROW "Parts" ROOT "PartList") AS partlist
FROM parts
WHERE parttype IN ('C01', 'E21')
GROUP BY parttype
```

PartCnt	PartList
2	<PartList><Parts><type>C01</type><pid>000130</pid> </Parts><Parts><type>C01</type><pid>200140</pid> </Parts></PartList>
3	<PartList><Parts><type>E21</type><pid>000320</pid> </Parts><Parts><type>E21</type><pid>100330</pid> </Parts><Parts><type>E21</type><pid>200340</pid> </Parts></PartList>

© 2011 IBM Corporation

IBM Power Systems

Fonctions de publication XML

IBM

XMLATTRIBUTES	Returns XML sequence that contains an attribute node for each non-null argument
XMLCOMMENT	Returns XML value with a single comment node from a string
XMLCONCAT	Returns XML value that represents a forest of XML elements generated by concatenating a variable number of arguments
XMLDOCUMENT	Returns XML value with a single document node and zero or more nodes as its children
XMLELEMENT	Returns XML value that represents an XML element
XMLFOREST	Returns XML value that represents a forest (sequence) of XML elements that all share a specific pattern
XMLPI	Returns XML value with a single processing instruction node
XMLNAMESPACES	Returns the declaration of one or more XML namespaces
XMLROW	Returns XML value with a single document node containing one top-level element node
XMLTEXT	Returns XML value with single text node that contains value of argument
XMLAGG	Returns an XML sequence that contains an item for each non-value in set of XML values
XMLGROUP	Returns XML value with a single document node containing one top-level element node from a group of rows

© 2011 IBM Corporation

IBM Power Systems

Instruction MERGE

IBM

- Permet à une application d'utiliser une seule instruction SQL pour mettre à jour, supprimer ou insérer des lignes dans une table cible à partir d'une table source
- Traitement typique
  - INSERT dans la table cible lorsque la (les) ligne(s) n'existe(nt) pas dans cette table
  - UPDATE des colonnes de la (des) ligne(s) de la table cible lorsqu'elle(s) existe(nt)
- Traitement conditionnel

© 2011 IBM Corporation

Instruction MERGE - Animation

IBM Power Systems

Détail Facture (source)

ORDID	ORDREG	...	ORDAMT
A11	W	...	10.00
A12	W	...	5.00
A13	E	...	30.00
A14	W	...	30.00
A15	E	...	20.00
...	...	...	...

Insertion

MERGE

Maj

Récapitulatif CA Région (cible)

REG	REG_TOT
W	45.00
E	30.00

© 2011 IBM Corporation

Instruction MERGE – Exemple 1

IBM Power Systems

- Fusionne les lignes de la table TRANSACTIONS dans la table COMPTES, met à jour le solde de la table COMPTES et ajoute les nouveaux comptes et leur solde

```
MERGE INTO comptes AS c
    USING (SELECT id, SUM(montant) as montant FROM transactions GROUP BY id) AS t
     ON c.id = t.id
    WHEN MATCHED THEN UPDATE SET solde = c.solde + t.montant
    WHEN NOT MATCHED THEN INSERT (id, solde) VALUES (t.id, t.montant) ;
```

```
create table db2i710.comptes  (id decimal (10, 0), solde dec( 10, 2)) ;
```

select * from db2i710.comptes...

ID	SOLDE
1000	1500.00
2000	2500.00

```
create table db2i710.transactions (id decimal (10, 0), montant dec( 10, 2)) ;
```

select * from db2i710.transactions...

ID	MONTANT
2000	10.00
3000	3500.00
2000	10.00

Après le MERGE :

select * from db2i710.comptes...

ID	SOLDE
1000	1500.00
2000	2520.00
3000	3500.00

© 2011 IBM Corporation

Instruction MERGE – Exemple 2


```
MERGE INTO archive ar
USING (SELECT actID, actDesc, actDate, actLastChg
         FROM actAtlGrp) acON (ar.activityID = ac.actID) AND
   ar.activityGroup =     'A'

WHEN MATCHED and ac.actDate < CURRENT DATE THEN DELETE

WHEN MATCHED and ar.LastChg < ac.actLastChg
          THEN
             UPDATE SET(activityDesc,activityDate,LastChg) =
                                 (ac.actDesc,ac.actDate,DEFAULT)
WHEN NOT MATCHED AND ac.actDate >= CURRENT DATE
          THEN
            INSERT (activityGroup, activityID, activityDesc, activityDate)
            VALUES ('A', ac.actID, ac.actDesc, ac.actDate)
          ELSE IGNORE
```

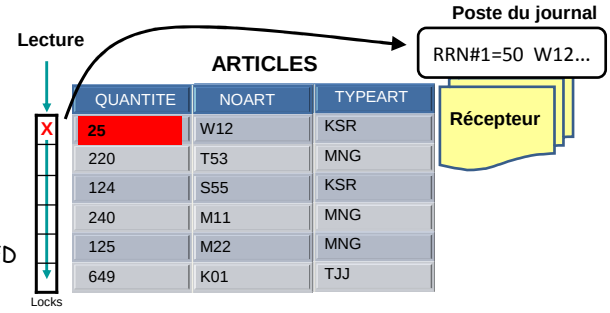

© 2011 IBM Corporation

Comportement en cas de verrouillage

- Plusieurs niveaux :
  - Niveau Système : SQL_CONCURRENT_ACCESS_RESOLUTION QAQQINI option
  - Niveau connexion : ADO.NET, JDBC, ODBC, OLE DB, SQL CLI et JDBC natif
  - Niveau programme : paramètre : CONACC(*DFT/*CURCMT/*WAIT) des précompilateurs
  - Niveau instruction : clause :
    - USE CURRENTLY COMMITTED
    - WAIT FOR OUTCOME
    - SKIP LOCKED DATA (ajouté en 6.1)

Travail #1 (mini *CHG/*UR) :
UPDATE articles
SET quantite = 25
WHERE noart='W12'

Travail #2 (si *CS/*RC)
SELECT quantite
FROM articles
WHERE noart = 'W12'
USE CURRENTLY COMMITTED



© 2011 IBM Corporation

Fonctions SQL

- Concernant MQ :
  - Fonctions scalaires :
    MQSEND, MQREAD, MQREADCLOB, MQRECEIVE, MQRECEIVECLOB
  - Fonctions table :
    MQREADALL, MQREADALLCLOB, MQRECEIVEALL, MQRECEIVEALLCLOB

- Concernant l'intégration BLOB et CLOB :
  - GET_BLOB_FROM_FILE
  - GET_CLOB_FROM_FILE, GET_DBCLOB_FROM_FILE

- Manipulation de bits :
  - BITAND, BITOR, BITXOR
  - BITNOT, BITANDNOT



© 2011 IBM Corporation

Simplification de l'accès distant

- Nom d'Alias/table/vue en trois parties
  - Exemple 1
    - Sur SYSTEMA :
      CREATE ALIAS macol.table1 FOR SYSTEMB.macol.table1

      Crée un ALIAS table1 dans macol de SYSTEMA qui pointe vers table1 de macol du système SYSTEMB

    - Sur SYSTEMA :
      SELECT c1, c2 FROM macol.table1
      ou
      SELECT c1, c2 FROM SYSTEMB.macol.table1
      Accède à la table1 du schéma macol du système SYSTEMB

    - Equivalent à un fichier DDM
    - A savoir :
      - » Une instruction SQL ne peut référencer que des objets d'un seul système
      - » Le nom de l'alias local doit être identique au nom de la table distante
      - » Un alias local peut référencer un alias sur un système distant

Elimine le CONNECT ou le SET CONNECTION.

© 2011 IBM Corporation

Simplification de l'accès distant

IBM Power Systems

- Exemple 2
  - Sur SYSTEMA :

```
CREATE ALIAS SYSTEMB.macol.table1
             FOR macol.table1
```

Crée un ALIAS macol.table1 sur SYSTEMB qui pointe vers macol.table1 du SYSTEMA

- Création automatique des packages SQL (*SQLPKG) sur le système distant

© 2011 IBM Corporation

Interfaces standard de l'industrie

IBM Power Systems

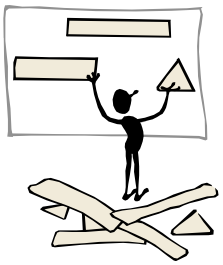
- ADO.NET
  - Propriété de comportement en cas de verrouillage
  - Support de Visual Studio 2008
  - Intégration Aide en ligne avec Visual Studio
  - Support middleware pour Delete, Merge et Update multi-lignes
- ODBC
  - Propriété de comportement en cas de verrouillage
  - Support middleware pour Delete, Merge et Update multi-lignes
- OLE DB
  - Propriété de comportement en cas de verrouillage
- SQL CLI
  - Support du type TINYINT
  - Attribut de connexion : SQL_ATTR_CONCURRENT_ACCESS_RESOLUTION
  - Support middleware pour Delete, Merge et Update multi-lignes
  - Variable d'environnement QIBM_SRVRMODE_SBS pour les travaux QSQSRVR (PTFs pour 5.4 et 6.1)
- JDBC
  - Support du type ARRAY pour les paramètres des routines SQL
  - Propriété de comportement en cas de verrouillage
  - Améliorations driver JDBC natif :
    - Propriété "servermode subsystem" pour contrôler le sous-système pour les travaux QSQSRVR
    - Compatibilité des métadata entre Toolbox JDBC et les autres drivers de l'industrie

© 2011 IBM Corporation

Améliorations diverses

- Option REPLACE pour les instructions CREATE
  - Elimine les instructions DROP
  - Concerne : Alias, Fonction, Procédure, Séquence, Trigger, Variable et Vue
  - Conserve les privilèges

  CREATE OR REPLACE ALIAS monAlias FOR schema.tab1
- Améliorations ALTER TABLE
  - Clause ADD BEFORE column
  - Support colonne IDENTITY pour des colonnes existantes
  - Sauvegarde des informations statistiques
  - Amélioration de performance pour tables partitionnées
- Améliorations tables partitionnées
  - Constrainte intégrités référentielles
  - Support colonne identity
- Support de compression SQL pour table, vue et index
- Nom de schéma sur 128 caractères
- Procédure QSYS2.CANCEL_SQL (PTF 6.1 : SI36319)
  CANCEL_SQL('197968/QUSER/QZDASOINIT')



© 2011 IBM Corporation

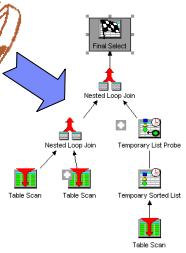
Performance

© 2011 IBM Corporation

Fonction AQP de SQE

- AQP = Adaptive Query Processing
  - Global Statistics Cache (GSC)
    - Contient les nombres de lignes réels
  - AQP Request Support
    - Après l'exécution d'une instruction il alimente GSC si divergence entre nb lignes estimé et réal
  - AQP Handler
    - Si exécution > 2 secondes et aucun enregistrement
    - Détection divergence entre nb ligne estimé et réalisé
    - Peut activer des modifications de plan lorsque la requête est active
    - Modification du plan et réexécution sont totalement transparentes à l'application





- Modifications de plan concernent :
  - Ordre de jonction
  - Mise en oeuvre d'un LPG





© 2011 IBM Corporation

Indexation et optimisation SQE

- L'optimiseur SQE n'ignore plus les index réduits et les fichiers logiques avec Select/Omit
  CREATE INDEX cust_ix1 ON customers(cust_id) WHERE activeCust = 'Y'

- Encoded Vector Index (EVI) avec colonnes d'aggrégation (SUM, AVG, COUNT, VARIANCE)

  CREATE ENCODED VECTOR INDEX evi1 ON ventes(region) **INCLUDE** (SUM(national) , count(*))

  SELECT region, count(*) , SUM(national) FROM ventes GROUP BY region

  CREATE ENCODED VECTOR INDEX evi2 ON ventes(region) **INCLUDE** (SUM(national + export))

  SELECT region, SUM(national+export) FROM ventes GROUP by region

Si les tables changent les EVIs correspondants sont maintenus

© 2011 IBM Corporation

Améliorations SQE

- Support des fichiers logiques dans la clause SELECT … FROM
- Chargement des objets en mémoire si utilisation par SQE :
  - CHGPF … KEEPINMEM(*YES), CHGLF … KEEPINMEM(*YES)
    - Pas pris en compte si CQE ou I/O natives



© 2011 IBM Corporation

Amélioration support SSD (Solid State Disks)

- Les unités SSD peuvent améliorer les performances :
  - Accès aléatoire
  - Beaucoup de lectures, peu d'écriture
- Les interfaces DB2 permettent d'indiquer une préférence pour les unités SSD pour les tables, index , fichiers physiques et fichiers logiques
  - SQL: Clause : UNIT SSD (255 pour 5.45)
    - CREATE/ALTER TABLE **5.45 – Groupe DB2 SF99504 #23**
    - CREATE INDEX **6.10 – Groupe DB2 SF99601 #10**
  - CL : Paramètre : UNIT(*SSD) (255 pour 5.45)
    - CRTPF, CRTLF et CRTSRCPF
    - CHGPF, CHGLF et CHGSRCPF
- Nouveautés 7.1 : ALTER et CHGPF/LF effectuent les déplacements de données de façon asynchrone
- Nouveautés 7.1 : Nouveaux compteurs des accès séquentiels/aléatoires pour les tables et les index (SYSTABLESTAT et SYSINDEXSTAT)



Associated Bank

En déplaçant des tables vers des unités SSD le batch de fin de mois a été réduit de 40% ! *

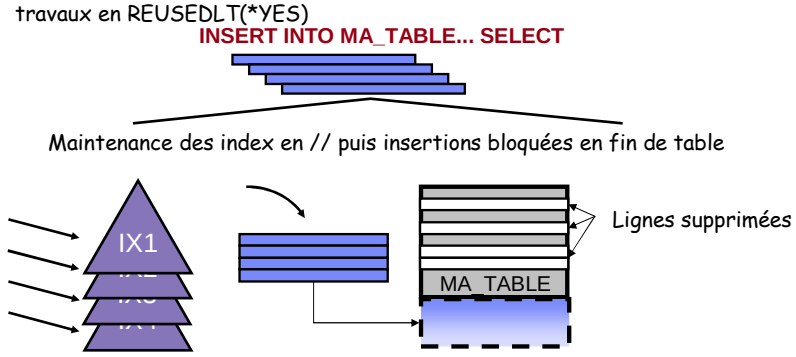
Nb d' unités SAS	Nb d' unités SSD	Temps batch
72	0	4:22
72	8	2:43
60	4	2:48

*http://www.ibmsystemsmagpowersystemsibmidigital.com/nxtbooks/ibmsystemsmag/ibmsystems_power_200909/index.php#/16

© 2011 IBM Corporation

OVRDBF REUSEDLT(*NO)

- Modification temporaire du paramètre REUSEDLT en *NO pour accélérer la performance des INSERT/WRITE
  - Permet à DB2 for i de faire du blocage de lignes (niveau DB2 ou/et niveau application)
  - Permet à DB2 for i SMP de faire de la maintenance d'index en //
  - Préserve la faculté de la table de faire du "concurrent write" dans les autres travaux en REUSEDLT(*YES)



Pour support en 5.4 et 6.1 (voir IBM Lab Services)
(http://ibm.com/systems/services/labservices/ -> "Contact Now" link)

© 2011 IBM Corporation

Résilience et disponibilité

© 2011 IBM Corporation

Résilience et disponibilité

- Moteur DB2 Engine
  - IASP
    - Commit/rollback sur tables de *SYSBAS et tables d'un IASP
    - Support for CICS transactions
  - Activation des contraintes sans vérification for HA Switchover
    - CHGPFCST … CHECK(*NO) pour 7.1
    - PTF SI32634 pour 5.4, Groupe DB2 level 6 pour 6.1
- Journalisation index SQL
  - Uniquement la page de 4 Ko qui contient la clé modifiée au lieu de la totalité de la page logique 64 Ko
- Règles de journalisation par nom d'objet générique pour les commandes STRJRNLIB et CHGJRNOBJ
- Attribut de Remote Journaling pour les commandes STRJRNLIB et CHGJRNOBJ

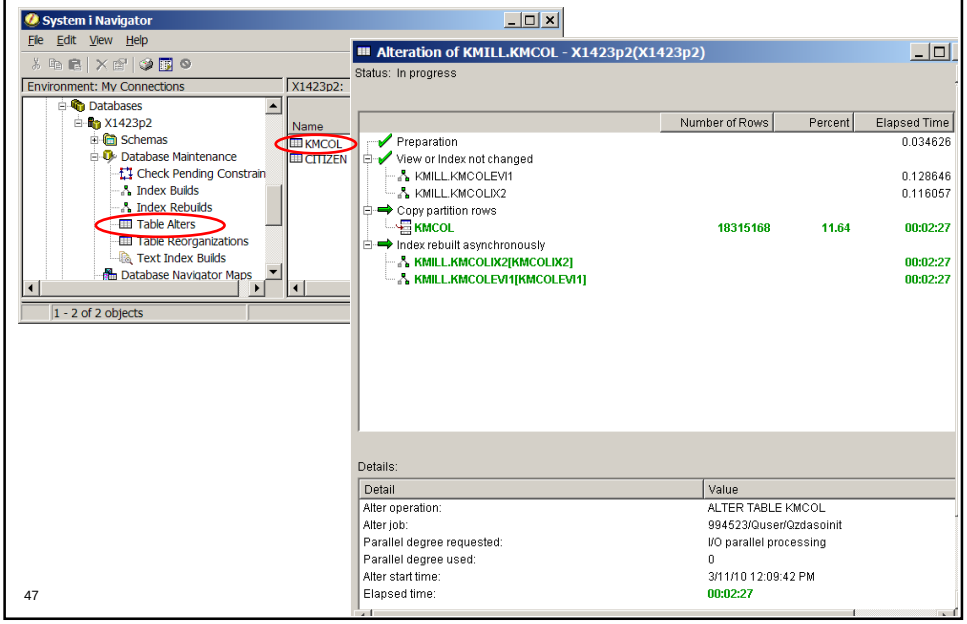
© 2011 IBM Corporation

Interface et sécurité

© 2011 IBM Corporation

Indicateur d'avancement – Exemple Alter Table

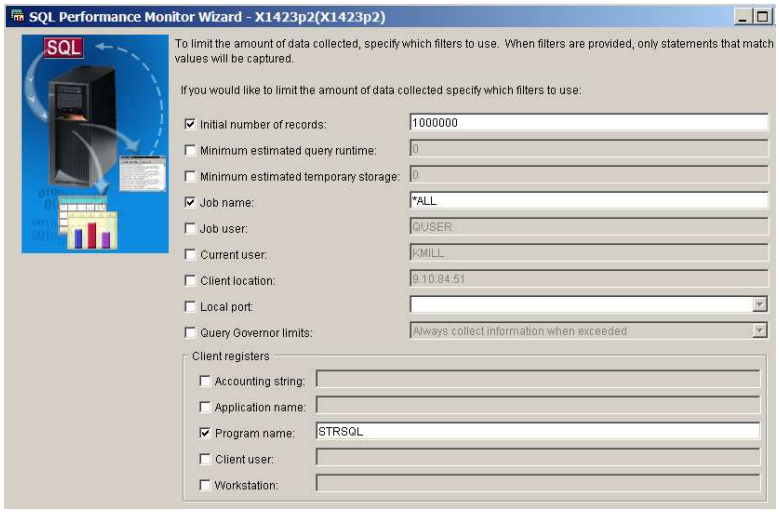
IBM Power Systems



47

Moniteur SQL – Filtres registres client

IBM Power Systems



Nouveau

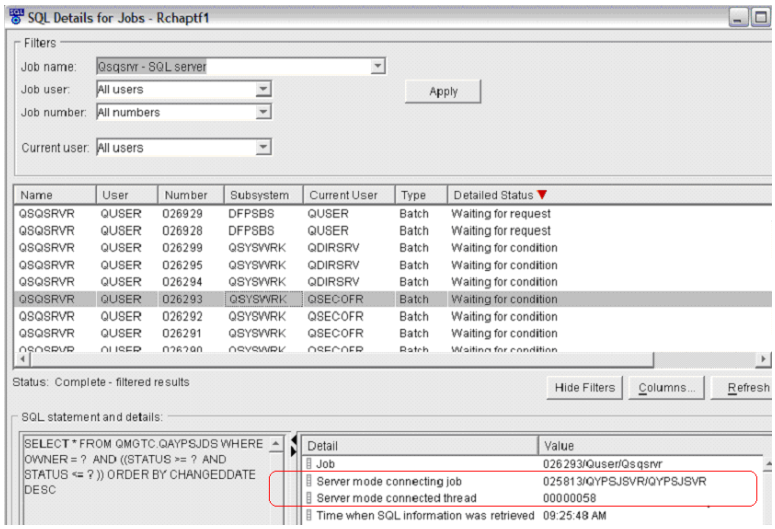
CL interface: STRDBMON OUTFILE(OUTMON1) JOB(*ALL) FTRCLTPGM(STRSQL)

48

© 2011 IBM Corporation

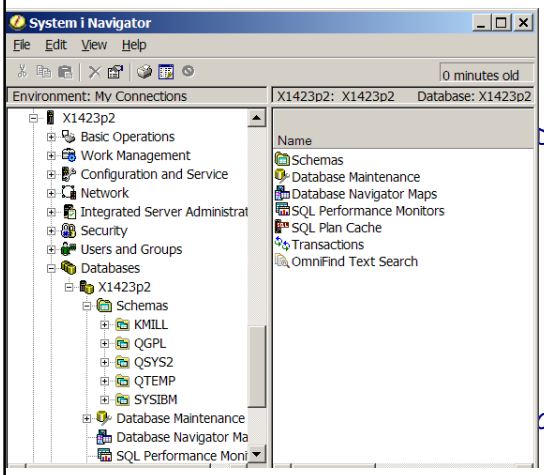
Détails SQL des travaux

- Information sur le travail à l'origine du travail QSQSRVR (PTF pour 5.4 et 6.1)



© 2011 IBM Corporation

System i Navigator - Divers



Les outils performance DB2 for i

- Améliorations Index Advisor
- Moniteur Database
  - Filtre sur erreur
  - Variable de remplacement

Database Management

- Support des index texte de OmniFind Text Search Server
- Generate SQL – Privilege & CCSID
- Gestion des listes d'objets :
  - Performance
  - Filtres liste d'objets
  - Sauvegarde contenu des listes

Centre de santé

- Suivi des erreurs SQL0901

© 2011 IBM Corporation

Sécurité - Fonctions

- *JOBCTL (Job Control Authority)
  - Ce qui fonctionnait en 6.1 continue à fonctionner

- QIBM_DB_SYSMON – Fonction Information BD
  - Permet à l'utilisateur de voir des détails niveau BD
  - Exemples :
    - Visualisation des informations SQL des travaux (API QUSRJOBI)
    - Visualisation des propriétés du cache de plan

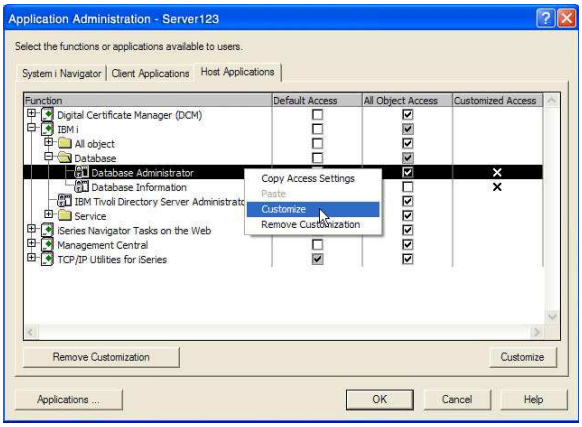
- QIBM_DB_SQLADM – Fonction Administrateur BD
  - QIBM_DB_SQLADM = QIBM_DB_SYSMON +
    - Utiliser Visual Explain
    - Changer le niveau de parallélisme de l'option DB2 SMP
    - Analyser des instructions du cache de plan
    - Analyser un moniteur de performance ou une image instantanée

© 2011 IBM Corporation

Sécurité - Fonctions

Commande 5250 :
CHGFCNUSG FCNID(QIBM_DB_SQLADM) USER(utilisateur) USAGE(*ALLOWED)

CHGFCNUSG FCNID(QIBM_DB_SYSMON) USER(utilisateur) USAGE(*ALLOWED)



© 2011 IBM Corporation

Sécurité - Données sensibles

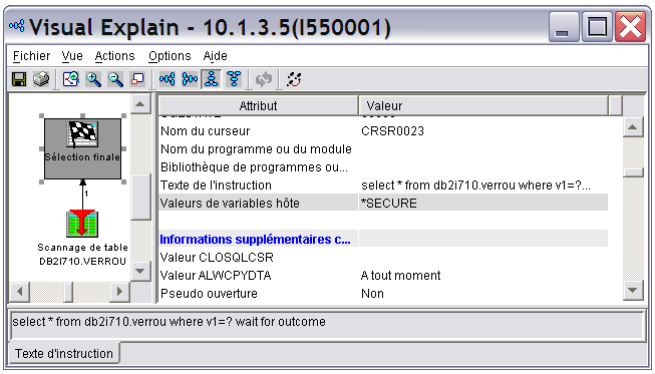
- Eviter la visualisation des données sensibles dans les outils de performance DB2 : moniteur de performance SQL et Cache de plan (PTF pour 5.4 et 6.1)
  - Seul QSECOFR pourra voir ces données sensibles
  - *SECURE sera présenté aux autres utilisateurs
  - Nécessité d'enregistrer les colonnes sensibles via une procédure stockée

- Procédure stockée SET_COLUMN_ATTRIBUTE
  - Paramètres :
    - Table_Schema - Nom du schéma
    - Table_Name - Nom de la table
    - Column_Name - Nom court de la colonne
    - Attribute - Attribut de sécurisation
      - » SECURE NO
      - » SECURE YES
  - Exemple :

```
CALL SYSPROC.SET_COLUMN_ATTRIBUTE
        ('DB2I710 'VERROU', 'V1', 'SECURE YES');
```


© 2011 IBM Corporation

Sécurité - Données sensibles



© 2011 IBM Corporation

Documentation et Informations

- Sites Web DB2 for i
  - Home Page: **ibm.com/systems/i/db2**
  - DeveloperWorks Zone: ibm.com/developerworks/db2/products/db2i5OS
  - Porting Zone: ibm.com/partnerworld/i/db2porting
- Newsgroups et Forums
  - USENET: comp.sys.ibm.as400.misc, comp.databases.ibm-db2
  - DeveloperWorks: https://www.ibm.com/developerworks/forums/forum.jspa?forumID=292
  - System i Network DB2 Forum: http://forums.systeminetwork.com/isnetforums/
- Education – Cours & Online
  - ibm.com/systemi/db2/gettingstarted.html
  - ibm.com/partnerworld/wps/training/i5os/courses
- Publications DB2 for i :
  - White Papers: ibm.com/partnerworld/wps/whitepaper/i5os
  - Online Manuals: ibm.com/systems/i/db2/books.html
  - DB2 for i Redbooks (http://ibm.com/redbooks)
    - Getting Started with DB2 Web Query for System i (SG24-7214)
    - OnDemand SQL Performance Analysis … in V5R4 (SG24-7326)
    - Preparing for and Tuning the SQL Query Engine on DB2 for i5/OS (SG24-6598)
    - Modernizing iSeries Application Data Access (SG24-6393)

© 2011 IBM Corporation